2 Debugging Command Reference

2.1 About This Document

Intended Audience

This document is intended for network engineers responsible for CloudEngine 8800, 7800, 6800, and 5800 series switches management and maintenance. You should be familiar with basic Ethernet knowledge and have extensive network management experience. In addition, you must have a good command of the CloudEngine 8800, 7800, 6800, and 5800 series switches product and master the implementation principles of each feature.

Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
NOTICE	Indicates a potentially hazardous situation which, if not avoided, could result in equipment damage, data loss, performance deterioration, or unanticipated results. NOTICE is used to address practices not related to personal injury.
📖 NOTE	Supplements the important information in the main text. NOTE is used to address information not related to personal injury, equipment damage, and environment deterioration.

Command Conventions

The command conventions that may be found in this document are defined as follows.

Convention	Description
Boldface	The keywords of a command line are in **boldface**.
Italic	Command arguments are in *italics*.
[ ]	Items (keywords or arguments) in brackets [ ] are optional.
{ x | y | ... }	Optional items are grouped in braces and separated by vertical bars. One item is selected.
[x | y | ...]	Optional items are grouped in brackets and separated by vertical bars. One item is selected or no item is selected.
{ x | y | ... }*	Optional items are grouped in braces and separated by vertical bars. A minimum of one item or a maximum of all items can be selected.
[x | y | ...]*	Optional items are grouped in brackets and separated by vertical bars. Several items or no item can be selected.
&<1-n>	The parameter before the & sign can be repeated 1 to n times.

Convention	Description
#	A line starting with the # sign is comments.

Interface Number Conventions

Interface numbers used in this manual are examples. In device configuration, use the existing interface numbers on devices.

Security Conventions

- Password setting
  - When configuring a password, the cipher text is recommended. To ensure device security, change the password periodically.
  - When you configure a password in plain text that starts and ends with %^%#......%^%# (the password can be decrypted by the device), the password is displayed in the same manner as the configured one in the configuration file. Do not use this setting. After the system master key is set using the **set master-key** command, do not start and end the key with **%@%#** because the string starting and ending with **%@%#** is considered as a valid cipher-text key.
  - When you configure a password in cipher text, different features cannot use the same cipher-text password. For example, the cipher-text password set for the AAA feature cannot be used for other features.
  - After the system software is downgraded and the switch restarts with the configuration of the higher version, AAA, VTY, serial interface login, and SNMP user passwords become invalid. As a result, users fail to log in to the switch using the passwords and the switch is disconnected from the network management system.

    To address this problem, take the following measures:

    i. If no password is configured for the console port, log in to the device through the console port, and reconfigure AAA and password for users such as VTY and SNMP users. For security purposes, the console port password is recommended.

    ii. If a password is configured for login through the console port, the password becomes invalid after the downgrade and you cannot log in to the switch through the console port. In the case of downgrade to a version later than V200R005C10, contact Huawei technical support engineers for assistance. If the version is downgraded to V200R005C10 or an earlier version, perform the following steps to resolve the issue:

       1) Connect to the console port.
       2) Power cycle the device. During the startup, enter **Ctrl+B** according to the prompt to enter the BIOS menu.
       3) Select **7.Modify console password** to delete and change the console port password.
       4) Restart the device, log in to the device through the console port, and reconfigure the password for AAA, VTY, or SNMP user.

- Encryption algorithm

  Currently, the device uses the following encryption algorithms: DES, 3DES, AES, DSA, RSA, DH, ECDH, HMAC, SHA1, SHA2, PBKDF2, scrypt, and MD5. The encryption algorithm depends on the applicable scenario. Use the recommended encryption algorithm; otherwise, security defense requirements may be not met.

  – For the symmetrical encryption algorithm, use AES with the key of 256 bits or more.

  – When you need to use an asymmetric cryptography, RSA (2048-bit or longer key) is recommended. In addition, use different key pairs for encryption and signature.

  – For the digital signature, RSA (2048-bit or longer key) or DSA (2048-bit or longer key) is recommended.

  – For key negotiation, DH (2048-bit or longer key) or ECDH (256-bit or longer key) is recommended.

  – For the hash algorithm, use SHA with the key of 256 bits or more.

  – For the HMAC algorithm, use HMAC-SHA2.

  – DES, 3DES, RSA and AES are reversible encryption algorithm. If protocols are used for interconnection, the locally stored password must be reversible.

  – SHA1, SHA2, and MD5 are irreversible encryption algorithm. When configuring a password for local administrator, it is recommended that you use the SHA2 irreversible encryption algorithm.

  – To prevent brute force cracking of the user password, the iteration algorithm is added to the password on the basis of salts. The iteration algorithm uses PBKDF2 or scrypt key export algorithm.

  – The ECB mode has a poor capability of defending against plaintext playback attacks, so ECB is not recommended for password encryption.

  – In SSH2.0, the symmetric cryptography using the CBC mode may undergo the plaintext-recovery attack to cause a data leak. Therefore, the CBC mode is not recommended for SSH2.0.

- Data

  Some data (such as MAC or IP addresses of terminals) may be obtained or used during operation or fault location of your purchased products, services, features, so you have an obligation to make privacy policies and take measures according to the applicable law of the country to protect data.

- The terms mirrored port, port mirroring, traffic mirroring, and mirroring in this manual are mentioned only to describe the product's function of communication error or failure detection, and do not involve collection or processing of any personal information or communication data of users.

Declaration

- This manual is only a reference for you to configure your devices. The contents in the manual, such as command line syntax, and command outputs, are based on the device conditions in the lab. The manual provides instructions for general scenarios, but do not cover all usage scenarios of all product models. The contents in the manual may be different from your actual device situations due to the differences in software versions, models, and configuration files. The manual will not list every possible difference. You should configure your devices according to actual situations.

- The specifications provided in this manual are tested in lab environment (for example, the tested device has been configured with a certain type of cards or only one protocol is run on the device). Results may differ from the listed specifications when you attempt to obtain the maximum values with multiple functions enabled on the device.
- In this document, public IP addresses may be used in feature introduction and configuration examples and are for reference only unless otherwise specified.

2.2 Usage of Debugging Commands

Overview

Debugging information is the traced information about internal running states of equipment and can be output to terminals. Debugging commands are an important tool used by the network administrator or system maintenance engineers to maintain equipment and locate faults.

- During routine system maintenance, the network administrator or system maintenance engineers can run the **ping** and **tracert** commands to check the network connectivity.
- During routine system debugging, the network administrator or system maintenance engineers can run **debugging** commands to output debugging information and locate faults in the system based on debugging information. Huawei data communications equipment provides a complete debugging command set to facilitate equipment maintenance.

Basic Principles

Huawei data communications equipment provides diversified debugging functions. For a majority of protocols and functions supported by the equipment, the system provides related debugging information to help users diagnose and locate faults.

- Format of debugging information

  Debugging information consists of the following fields:

  Timestamp Sysname Module/Level/Digest: Content

  For example:

  ```
  Dec 22 2012 18:22:54.230 HUAWEI %%01LDM/6/LDM_PKT(d):CID=0x80782743;LDM use APP VLAN
  priority=6
  ```

  Fields of debugging information are described as follows:

  – Timestamp

    This field records the time that debugging information is generated so that users can view and locate system events.

  – Sysname

    This field indicates the name of the system. The network administrator or system maintenance engineers can run the **sysname** command to modify the system name.

- Module

  This field indicates the name of the functional module that generates debugging information.

  For example, PPP indicates that the module processes the Point-to-Point Protocol (PPP).

- Level

  Debugging information is of eight levels from level 0 to level 7. The level of debugging information generated by each module has been already determined at the development phase.

- Digest

  This field is a phrase that outlines debugging information.

- Content

  This field is the detailed description about debugging information. The digest must be separated from the content by a colon (:). If a lot of information needs to be displayed, the information is displayed in multiple lines.

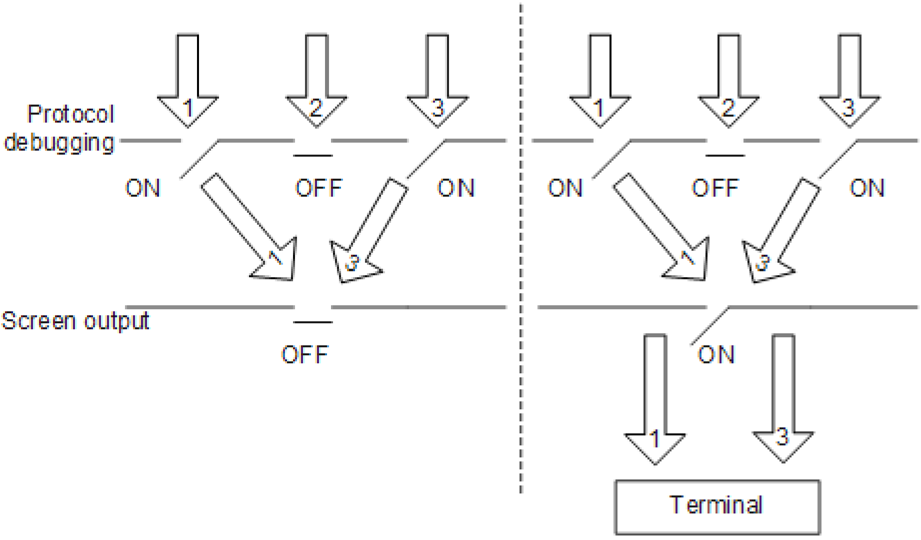
- System debugging

  Two types of debugging are used to control the output of debugging information:

  - The protocol debugging determines whether debugging information about a protocol is generated.
  - The screen output determines whether debugging information is displayed on the screen of a specified user.

  As shown in **Figure 2-1**, the system provides debugging information for modules 1 to 3. You must enable both the protocol debugging and the screen output to display debugging information on the terminal.

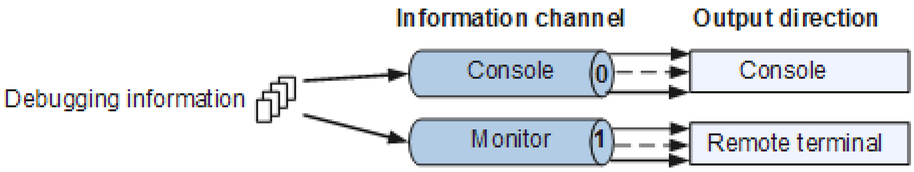
Figure 2-1 Relationship between debugging switches of the system



- Information output direction

  Debugging information is managed by the information center. Based on the association between the information channel and the output direction, the information center can output debugging information to different directions, as shown in **Figure 2-2**.

Figure 2-2 Output directions of debugging information



- The console is a terminal that is directly connected to the equipment through the Console port.
- The monitor is a terminal that can remotely log in to the equipment by using protocols, such as Secure Shell (SSH) and Telnet.

Outputting Debugging Information

NOTICE

- Debugging affects system performance. Enable debugging only when debugging information is required for fault location.

  When you shut down the console or tear down the SSH or Telnet remote connection, the system automatically disables all debugging to ensure that the system is not affected in the case debugging is left enabled.
- Be cautious when using the debugging command that contains the keyword "all". You are advised not to use the **debugging all** command. After debugging is complete, run the **undo debugging all** command to disable debugging immediately.

1. To enable the debugging of a specified module, run the **debugging** *module* command.

   You can view the module name by entering the question mark (?) after the **debugging** command, for example, **debugging ?**.
2. To enable the screen output of debugging information.
   - Run the **terminal debugging** command to display debugging information on a console.

     By default, when debugging information is output to a console, **terminal monitor** is in the enabled state. Therefore, you do not need to run the **terminal monitor** command.
   - Output debugging information to a monitor.
     1. Run the **terminal debugging** command to display debugging information on a terminal.
     2. Run the **terminal monitor** command to display debugging information on the monitor.

     By default, when debugging information is output to a monitor, **terminal monitor** is in the disabled state. Therefore, you need to run the **terminal monitor** command.

3. Run the **undo debugging** command to disable debugging.

   If all debugging tasks are complete, you can directly run the **undo debugging all** command to disable all debugging.

NOTE

In this document, commands used for debugging specified modules or protocols are provided as examples, and commands used for outputting debugging information to terminals are not provided.

2.3 Basic Configurations Debugging Commands

2.3.1 FTP Debugging Command

2.3.1.1 debugging

Function

The **debugging** command enables the debugging function of the FTP client.

The **undo debugging** command disables the debugging function of the FTP client.

By default, the debugging function of the FTP client is disabled.

Format

debugging

undo debugging

Parameters

None

Views

FTP client view

Default Level

3: Management level

Usage Guidelines

You can run the **debugging** command to check the session information sent by the FTP client to the FTP server.

Example

# Enable the debugging function of the FTP client.

```
<HUAWEI> ftp 1.1.1.1
Trying 1.1.1.1 ...
Press CTRL + K to abort
Connected to 1.1.1.1.
220 FTP service ready.
User(1.1.1.1:(none)):ftp
331 Password required for ftp.
Enter password:
230 User logged in.
[ftp] debugging
```

2.3.1.2 debugging ftp client

Function

Using the **debugging ftp client** command, you can enable the debugging of FTP client.

Using the **undo debugging ftp client** command, you can disable the debugging of FTP client.

By default, the debugging of FTP client is disabled.

Format

debugging ftp client { **all** | **cfsm** | **comm** | **dfsm** | **message** | **session** | **timer** } { **error** | **info** | **warning** }

undo debugging ftp client { **all** | **cfsm** | **comm** | **dfsm** | **message** | **session** | **timer** }

Parameters

Parameter	Description	Value
all	Enables the debugging of all the modules of the FTP client.	-
cfsm	Enables the debugging of the control state machine module of the FTP client.	-
comm	Enables the debugging of the communication adaptation module of the FTP client.	-
dfsm	Enables the debugging of the data state machine module of the FTP client.	-
message	Enables the message interaction debugging of the FTP client.	-
session	Enables the debugging of the session module of the FTP client.	-
timer	Enables the debugging of the timer module of the FTP client.	-
error	Enables the error debugging of the FTP client.	-
info	Enables the information debugging of the FTP client.	-
warning	Enables the warning debugging of the FTP client.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ftp-client	execute

Usage Guidelines

When FTP client is connecting to server, you can run this command to print the debug prints of FTP client to know the connection progress and rapidly locate faults based on the obtained information.

Example

# Enable the error debug information of FTP client.

```
<HUAWEI> debugging ftp client all error
```

2.3.1.3 debugging ftp server

Function

Using the **debugging ftp server** command, you can enable the debugging of FTP server.

Using the **undo debugging ftp server** command, you can disable the debugging of FTP server.

By default, the debugging of FTP server is disabled.

Format

debugging ftp server { **aaa** | **all** | **comm** | **message** | **socket** | **tfs** }

undo debugging ftp server { **aaa** | **all** | **comm** | **message** | **socket** | **tfs** }

Parameters

Parameter	Description	Value
aaa	Enables debugging of the FTP server's AAA information.	-
all	Enables debugging of all the FTP server information.	-
comm	Enables debugging of the FTP server's public module.	-

Parameter	Description	Value
message	Enables debugging of the FTP server's component information.	-
socket	Enables debugging of the FTP server's socket information.	-
tfs	Enables debugging of the FTP server's TFS information.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ftp-server	read

Usage Guidelines

When FTP server module becomes faulty, you can run this command to print the debug prints of FTP server to know the connection progress and rapidly locate faults based on the obtained information.

Example

# Enable debugging of the FTP server's AAA information.

```
<HUAWEI> debugging ftp server aaa
```

2.3.2 TFTP Debugging Command

2.3.2.1 debugging tftp client

Function

Using the **debugging tftp client** command, you can enable the debugging of TFTP client.

Using the **undo debugging ftp client** command, you can disable the debugging of TFTP client.

By default, the debugging of TFTP client is disabled.

Format

debugging tftp client

undo debugging tftp client

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
tftp-client	execute

Usage Guidelines

When TFTP client is connecting to server, you can run this command to print the debug prints of TFTP client to know the connection progress and rapidly locate faults based on the obtained information.

Example

# Enable debug prints of TFTP client information.

```
<HUAWEI> debugging tftp client
```

2.3.3 Telnet Debugging Command

2.3.3.1 debugging telnet

Function

Using the **debugging telnet** command, you can enable the debugging function of Telnet connection.

Using the **undo debugging telnet** command, you can disable the debugging function of Telnet connection.

By default, the debugging of Telnet connection is disabled.

Format

debugging telnet

undo debugging telnet

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
telnet-client	execute

Usage Guidelines

When a Telnet client connection becomes faulty, the network administrator cannot perform local management using Telnet client on the remote device. You can run this command to start the debugging information on the Telnet client connection and rapidly locate faults based on the obtained information.

Example

# Enable Telnet client debugging information.

```
<HUAWEI> debugging telnet
```

2.3.3.2 debugging telnet server fsm

Function

The **debugging telnet server fsm** command enables the debugging function of Telnet server FSM.

The **undo debugging telnet server fsm** command disables the debugging function of Telnet server FSM.

By default, the debugging of Telnet server FSM is disabled.

Format

debugging telnet server fsm

undo debugging telnet server fsm

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
telnet-server	read

Usage Guidelines

When a Telnet client is connecting to server, you can run this command to output FSM debug prints in the terminal and rapidly locate faults based on the obtained information.

Example

# Enable debugging of Telnet server FSM information.

```
<HUAWEI> debugging telnet server fsm
Feb 28 2012 06:55:51.321 HUAWEI %%01TELNETS/7/
TELNETS_SEND_WILLECHO(d):CID=0x80c8272b;TELNETS: Negotiation data sent for WILL ECHO

Feb 28 2012 06:55:51.321 HUAWEI %%01TELNETS/7/TELNETS_SEND_NOGA(d):CID=0x80c8272b;TELNETS:
Negotiation data sent for WILL NOGA

Feb 28 2012 06:55:51.321 HUAWEI %%01TELNETS/7/
TELNETS_SEND_DOTERMTYPE(d):CID=0x80c8272b;TELNETS: Negotiation data sent for DO TERMTYPE

Feb 28 2012 06:55:51.321 HUAWEI %%01TELNETS/7/
TELNETS_SEND_DONAWS(d):CID=0x80c8272b;TELNETS: Negotiation data sent for DO NAWS
```

2.3.3.3 debugging telnet server negotiate

Function

The **debugging telnet server negotiate** command prints the debug prints of Telnet server.

The **undo debugging telnet server negotiate** command stops the Telnet debug prints coming on the terminal.

By default, the debugging prints of Telnet server is disabled.

Format

debugging telnet server negotiate

undo debugging telnet server negotiate

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
telnet-server	read

Usage Guidelines

When a Telnet client is connecting to server, you can run this command to print the debug prints of Telnet server to know the connection progress and rapidly locate faults based on the obtained information.

Example

# Enable debug prints of Telnet server information.

```
<HUAWEI> debugging telnet server negotiate
Feb 28 2012 09:54:14.208 HUAWEI %%01TELNETS/7/
TELNETS_ACCEPT_BEGIN(d):CID=0x80c8272b;TELNETS: (0) VTY ACCEPT BEGIN

Feb 28 2012 09:54:14.208 HUAWEI %%01TELNETS/7/
TELNETS_MAXCONN_CHECK(d):CID=0x80c8272b;TELNETS: (3) USER NUMBER LESS THAN MAX
CONNECTION OK!

Feb 28 2012 09:54:14.208 HUAWEI %%01TELNETS/7/TELNETS_ACL_CHECK(d):CID=0x80c8272b;TELNETS:
(4) ACCESS-LIST PASSED. OK!

Feb 28 2012 09:54:14.208 HUAWEI %%01TELNETS/7/
TELNETS_SOCKET_ACCEPT(d):CID=0x80c8272b;TELNETS: (1) SOCKET ACCEPT OK!
```

2.3.4 SSH Debugging Command

2.3.4.1 debugging ssh client

Function

Using the **debugging ssh client** command, you can enable the debugging function of SSH client module.

Using the **undo debugging ssh client** command, you can disable the debugging function of SSH client module.

By default, the debugging of SSH client module is disabled.

Format

debugging ssh client { **all** | **event** | **message** | **packet** | **error** | **state-transition** }

undo debugging ssh client { **all** | **event** | **message** | **packet** | **error** | **state-transition** }

Parameters

Parameter	Description	Value
all	Enables all debugging functions of SSH client.	-
event	Displays debugging information about event of SSH client.	-
message	Displays debugging information about message of SSH client.	-
packet	Displays debugging information about packet of SSH client.	-
error	Displays debugging information about error of SSH client.	-
state-transition	Displays debugging information about state transition of SSH client.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ssh-client	execute

Usage Guidelines

When a SSH client module becomes faulty, the network administrator cannot perform local management using SSH client to start, modify, or delete configuration on the remote device. You can run this command to start the debugging information on the SSH client module and rapidly locate faults based on the obtained information.

Example

# Enable all SSH client module debugging information.

```
<HUAWEI> debugging ssh client all
```

2.3.4.2 debugging ssh server

Function

Using the **debugging ssh server** command, you can enable the debugging function of SSH server module.

Using the **undo debugging ssh server** command, you can disable the debugging function of SSH server module.

By default, the debugging of SSH server module is disabled.

Format

debugging ssh server { **all** | **event** | **message** | **packet** | **error** | **state-transition** } [**session** *session-id*]

undo debugging ssh server { **all** | **event** | **message** | **packet** | **error** | **state-transition** } [**session** *session-id*]

Parameters

Parameter	Description	Value
all	Enables all debugging functions of SSH server.	-
event	Displays debugging information about event of SSH server.	-
message	Displays debugging information about message of SSH server.	-
packet	Displays debugging information about packet of SSH server.	-
error	Displays debugging information about error of SSH server.	-
state-transition	Displays debugging information about state transition of SSH server.	-
session *session-id*	Specifies the session ID.	The value is an integer ranging from 1 to 1024.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ssh-server	execute

Usage Guidelines

When a SSH server module becomes faulty, the network administrator cannot perform local management using SSH server to start, modify, or delete configuration on the remote device. You can run this command to start the debugging information on the SSH server module and rapidly locate faults based on the obtained information.

Example

# Enable all SSH server module debugging information.

```
<HUAWEI> debugging ssh server all
```

2.3.5 HTTP Debugging Commands

2.3.5.1 debugging http client all

Function

The **debugging http client all** command enables HTTP client debugging.

The **undo debugging http client all** command disables HTTP client debugging.

By default, HTTP client debugging is disabled.

Format

debugging http client all

undo debugging http client all

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

If the HTTP module does not function properly, run the **debugging http client all** command to enable HTTP client debugging so that you can locate the fault based on debugging information.

Example

# Enable HTTP client debugging.

```
<HUAWEI> debugging http client all
Oct 24 2014 15:21:21.547 "HUAWEI" %%01HTTPC/7/NORMALDEBUGOUT(d):CID=0x82d70411;
[HTTPC] [15:21:21:546] :[CMD]----Recv from hPid[0x00cc000b]: [SMP CFGI(0)] [ACTION(2)]----

Oct 24 2014 15:21:21.547 "HUAWEI" %%01HTTPC/7/NORMALDEBUGOUT(d):CID=0x82d70411;
[HTTPC] [15:21:21:546] :[CMFM]: CreateRspMsg: sessID:59.

Oct 24 2014 15:21:21.547 "HUAWEI" %%01HTTPC/7/NORMALDEBUGOUT(d):CID=0x82d70411;
[HTTPC] [15:21:21:546] :HTTPC_CMFM_ProcCfgActionReq CMF_SndRspMsg.

Oct 24 2014 15:21:21.547 "HUAWEI" %%01HTTPC/7/NORMALDEBUGOUT(d):CID=0x82d70411;
[HTTPC] [15:21:21:546] :[CMFM]: FreeRspMsg: sessID:59, timeid:-1.
```

2.3.5.2 debugging http server all

Function

The **debugging http server all** command enables HTTP server debugging.

The **undo debugging http server all** command disables HTTP server debugging.

By default, HTTP server debugging is disabled.

Format

debugging http server all

undo debugging http server all

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

If the HTTP module does not function properly, run the **debugging http server all** command to enable HTTP server debugging. The command output helps locate the fault.

Example

# Enable HTTP server debugging.

```
<HUAWEI> debugging http server all
```

2.3.6 CFG Debugging Commands

2.3.6.1 debugging packet filter

Function

The **debugging packet filter** command sets the filter condition for tracing protocol packets.

The **undo debugging packet filter** command cancels the filter condition for tracing protocol packets.

By default, all protocol packets are displayed after the protocol packet tracking function is enabled.

Format

# Set the filter condition for tracing Layer 2 protocol packets.

debugging packet filter L2 { **application** | **link** }

undo debugging packet filter L2 { **application** | **link** }

# Set the filter condition for tracing Layer 3 protocol packets.

debugging packet filter L3 { **application** | **ipv4** | **ipv6** | **transport** }

undo debugging packet filter L3 { **application** | **ipv4** | **ipv6** | **transport** }

# Set the filter condition for tracing protocol packets on the inbound and outbound interfaces.

debugging packet filter { **egress** | **ingress** }

undo debugging packet filter { **egress** | **ingress** }

Parameters

Parameter	Description	Value
l2	Filters Layer 2 protocol packets.	-
application	Filters application layer protocol packets.	-
link	Filters link layer protocol packets.	-
l3	Filters Layer 3 protocol packets.	-
ipv4	Filters IPv4 protocol packets.	-
ipv6	Filters IPv6 protocol packets.	-

Parameter	Description	Value
transport	Filters transport layer protocol packets.	-
egress	Filters protocol packets on the outbound interface.	-
ingress	Filters protocol packets on the inbound interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

When the debugging information about layer 2 or layer 3 protocol packets needs to be output on a terminal, the **debugging packet filter** command sets the filter condition for tracing the specified protocol packets. The debugging information about the protocol packets that meets the filter condition is not output on the terminal.

Follow-up Procedure

After the filter condition for tracing protocol packets is set, run the **display debugging packet all** command to view the debugging information about the protocol packets on the terminal.

Example

# Filter protocol packets at the link layer.

```
<HUAWEI> debugging packet filter l2 link
```

# Filter protocol packets at the transport layer.

```
<HUAWEI> debugging packet filter l3 transport
```

2.3.6.2 debugging packet timeout

Function

The **debugging packet timeout** command sets the timeout interval for tracing protocol packets.

By default, the timeout interval for tracing protocol packets is 0, indicating that no timeout interval is set.

Format

debugging packet timeout *timeout-value*

Parameters

Parameter	Description	Value
timeout-value	Specifies the timeout interval for tracing protocol packets.	The value is an integer ranging from 0 to 3600, seconds.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
cfg	debug

Usage Guidelines

Usage Scenario

Debugging affects the system performance. A user may forget to disable debugging in time, which adversely affects the system performance. The **debugging packet timeout** command sets the timeout interval for tracing protocol packets. When the timeout interval for tracing protocol packets reaches the preset value, the debugging command for tracing all protocol packets become invalid automatically and the terminal stops outputting debugging information.

Follow-up Procedure

After the timeout interval for tracing protocol packets is set, run the **display debugging packet all** command to check the timeout interval for tracing protocol packets.

Example

# Set the timeout interval for tracing protocol packets to 10s.

```
<HUAWEI> debugging packet timeout 10
```

2.3.6.3 display debugging packet all

Function

The **display debugging packet all** command displays the tracing status of all protocol packets.

Format

display debugging packet all

Parameters

None

Views

All views

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
cfg	read

Usage Guidelines

After the following operation is performed, you can run the **display debugging packet all** command to view the tracing status of all protocol packets.

- Run the **debugging packet filter** command to set the filter condition for tracing protocol packets.
- Run the **debugging packet timeout** command to set the timeout interval for tracing protocol packets.

Example

# Display the tracing status of all protocol packets.

```
<HUAWEI> display debugging packet all
[Y]: The packet of specified type will display
L2:  Application [Y]  Link [Y]
L3:  Application [Y]  Transport [N]  Ipv4 [Y]  Ipv6 [Y]
Ingress/Egress:  Ingress [Y]  Egress [Y]

Debugging time left: 8(s)
```

Table 2-1 Description of the **display debugging packet all** command output

Item	Description
[Y]	A terminal outputs debugging information about the specified protocol packets.
[N]	A terminal does not output debugging information about the specified protocol packets.
L2	A terminal outputs debugging information about protocol packets at Layer 2.
Application	A terminal outputs debugging information about protocol packets at the application layer.

Item	Description
Link	A terminal outputs debugging information about protocol packets at the link layer.
L3	A terminal outputs debugging information about protocol packets at Layer 3.
Transport	A terminal outputs debugging information about protocol packets at the transport layer.
Ipv4	A terminal outputs debugging information about IPv4 protocol packets.
Ipv6	A terminal outputs debugging information about IPv6 protocol packets.
Ingress	A terminal outputs debugging information about protocol packets on the inbound interface.
Egress	A terminal outputs debugging information about protocol packets on the outbound interface.
Debugging time left	Whether a terminal sets the timeout interval for tracing protocol packets. ● After the **debugging packet timeout** command is run to set the timeout interval for tracing protocol packets, the specific interval is displayed here. ● If the timeout interval for tracing protocol packets is not set, the following information is displayed: No debugging timeout.

2.3.6.4 undo debugging packet all

Function

The **undo debugging packet all** command disables debugging of all protocol packets.

Format

undo debugging packet all

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
cfg	debug

Usage Guidelines

Debugging affects the system performance. After debugging is complete, run the **undo debugging packet all** command in time to disable debugging of all protocol packets.

Example

# Disable debugging of all protocol packets.

```
<HUAWEI> undo debugging packet all
```

2.3.6.5 debugging tty

Function

The **debugging tty** command enables TTY debugging.

The **undo debugging tty** command disables TTY debugging.

By default, TTY debugging is disabled.

Format

debugging tty { **message** | **all** } { **info** | **warning** | **error** }

undo debugging tty { **message** | **all** }

Parameters

Parameter	Description	Value
message	Enables TTY message debugging.	-
all	Enables all TTY debugging functions.	-
info	Enables TTY information debugging.	-
warning	Enables TTY warning debugging.	-
error	Enables TTY error debugging.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

If a TTY fault occurs, run the **debugging tty** command to enable TTY debugging so that you can locate the fault based on debugging information.

Example

# Enable the debugging of informational TTY messages.

```
<HUAWEI> debugging tty message info
```

# Enable the debugging of all TTY warnings.

```
<HUAWEI> debugging tty all warning
```

2.3.7 Upgrade Debugging Commands

2.3.7.1 debugging license

Function

The **debugging license** command enables the debugging of a license module.

The **undo debugging license** command disables the debugging of a license module.

By default, the debugging of a license module is disabled.

Format

debugging license { **all** | **error** | **event** | **lib** }

undo debugging license { **all** | **error** | **event** | **lib** }

Parameters

Parameter	Description	Value
all	Enables all the debugging.	-
error	Enables the debugging of errors.	-
event	Enables the debugging of events.	-
lib	Enables the debugging of the encryption base.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

You can enable all the debugging of a license module to quickly locate faults.

Example

# Enable the debugging of events on a license module.

```
<HUAWEI> debugging license event
```

2.4 Ethernet Switching Configuration Debugging Commands

2.4.1 LACP Debugging Commands

2.4.1.1 debugging lacp

Function

The **debugging lacp** command enables LACP debugging.

The **undo debugging lacp** command disables LACP debugging.

By default, LACP debugging is disabled.

Format

debugging lacp { **all** | **error** | **event** | **fsm** | **message** | **packet** [**eth-trunk** *trunk-id* [**interface** *interface-type interface-number*]] }

undo debugging lacp { **all** | **error** | **event** | **fsm** | **message** | **packet** [**eth-trunk** *trunk-id* [**interface** *interface-type interface-number*]] }

Parameters

Parameter	Description	Value
all	Enables all the LACP debugging.	-
error	Enables the debugging of LACP errors.	-

Parameter	Description	Value
event	Enables the debugging of LACP events.	-
fsm	Enables the debugging of LACP state machine.	-
message	Enables the debugging of LACP messages.	-
packet	Enables the debugging of LACPDUs.	-
eth-trunk *trunk-id*	Specifies the ID of an Eth-Trunk interface. If this parameter is not specified, the debugging information about all Eth-Trunk interfaces is displayed.	The value is an integer. You can enter a question mark (?) and select a value from the displayed value range.
interface *interface-type interface-number*	Enables the debugging of packets on a specified member interface of an Eth-Trunk. If multiple physical interfaces of the device are added to an Eth-Trunk interface in LACP mode, when you run the command to enable debugging on the Eth-Trunk interface, a lot of debugging information will be displayed, causing high CPU usage. In this situation, you can specify **interface** *interface-type interface-number* in the command to view debugging information on a specified Eth-Trunk member interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging lacp** command displays the debugging information about the LACP module. Different debugging information can be obtained by selecting different key words, which facilitates fault location and equipment maintenance.

Prerequisites

When you select **eth-trunk** *trunk-id* to obtain debugging information, the Eth-Trunk interface has been created.

Example

# Enable debugging of the LACP module and display debugging information of LACPDUs.

```
<HUAWEI> debugging lacp packet
2011-07-20 16:13:12 V8_B7_158 %%01LACP/7/LACP_DEBUG_RECEIVEPKT(d):CID=2;LACP receive protocol
packet.(IfIndex = 9)
      Actor info: sysPri = 32768, sysId = 0025-9eb1-abfa, portKey = 25649, portPri = 32768, portState =
10111100
      Partner info: sysPri = 32768, sysId = 0025-9eb2-4618, portKey = 25649, portPri = 32768, portState =
10111100
```

Table 2-2 Description of the debugging lacp command output

Item	Description
Actor info	Information about member interfaces of the local Eth-Trunk
sysPri	Priority of the LACP system
sysId	ID of the LACP system
portKey	Port key word
portPri	Port priority
portState	Port state
Partner info	Information about member interfaces of the peer Eth-Trunk

2.4.2 VLAN Debugging Commands

2.4.2.1 debugging vlan

Function

The **debugging vlan** command enables VLAN debugging.

The **undo debugging vlan** command disables VLAN debugging.

By default, VLAN debugging is disabled.

Format

debugging vlan { **error** | **event** | **message** }

undo debugging vlan { **all** | **error** | **event** | **message** }

Parameters

Parameter	Description	Value
all	Specifies all debugging functions.	-
error	Specifies the error debugging function.	-
event	Specifies the event debugging function.	-
message	Specifies the message debugging function.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable debugging for the VLAN module, run the **debugging vlan** command. The command output helps you troubleshoot faults and maintain devices.

Example

# Enables the VLAN error debugging function and display debugging information.

```
<HUAWEI> debugging vlan error
```

2.4.3 MSTP/VBST Debugging Commands

2.4.3.1 debugging stp all

Function

Using the **debugging stp all** command, you can enable all the MSTP debugging.

Using the **undo debugging stp all** command, you can disable all the MSTP debugging.

By default, the debugging of all the MSTP is disabled.

Format

debugging stp all

undo debugging stp all

Parameters

None.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging stp all** command enables all the MSTP debugging, facilitating fault location.

Example

# Enable all the MSTP debugging.

```
<HUAWEI> debugging stp all
Apr 14 2009 13:19:11.650.4 HUAWEI MSTP/8/MEXS:Instance 0's port7 enters PPM%SENDING_RSTP state.
```

# The status of port 7 of the Spanning Tree Protocol (STP) instance 0 turns SENDING_RSTP of the PPM state machines. In PPM%SENDING_RSTP, PPM indicates the type of the state machines, while SENDING_RSTP indicates the status of the state machines.

```
Apr 14 2009 13:19:11.650.4 HUAWEI MSTP/8/MEXS:Instance 0's port7 enters PIM%CURRENT state.
```

# The status of port 7 of the STP instance 0 turns CURRENT of the PPM state machines. In PPM%CURRENT, PPM indicates the type of the state machines, while CURRENT indicates the status of the state machines.

```
Apr 14 2009 13:19:11.650.4 HUAWEI MSTP/8/PKT:
Port7(10GE1/0/1)  Rcvd Packet(Length: 519)
ProtocolVersionID:03
BPDUType:02( RST BPDU )
Flags: 2c( DESIGNATED  Forwarding )
CIST Root Identifier: 0.000b-09c9-6bac
CIST External Path Cost: 0
CIST Bridge Identifier: 0.000b-09c9-6bac
CIST Port Identifier: 128.73
Message Age: 0
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 480
CIST Regional Root Identifier: 0.000b-09c9-6bac
CIST Internal Root Path Cost: 0
CIST Remaining Hops: 20
Instance: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13,
14, 15, 16
```

# 10GE1/0/1 receives an RST BPDU packet. The peer port of 10GE1/0/1, namely, the port that sends the packet, is the designated port and the port status is Forwarding.

```
Apr 14 2009 13:19:11.650.4 HUAWEI MSTP/8/MSG:
InstanceID: 1
```

```
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 32768.000b-09c9-6bac
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 32768.000b-09c9-6bac
MSTI Port Priority: 128.73
MSTI Remaining Hops: 20
```

# The preceding display describes information about MSTI 1. The peer port of 10GE1/0/1, namely, the port that sends the packet, is the designated port in MSTI 1 and the port status is Forwarding.

```
Apr 14 2009 13:19:11.650.4 Quidway MSTP/8/MSG:
InstanceID: 2
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 32768.000b-09c9-6bac
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 32768.000b-09c9-6bac
MSTI Port Priority: 128.73
MSTI Remaining Hops: 20
```

# The preceding display describes information about MSTI 2. The peer port of 10GE1/0/1, namely, the port that sends the packet, is the designated port in MSTI 2 and the port status is Forwarding.

```
Apr 14 2009 13:19:11.650.4 Quidway MSTP/8/MEXS:Instance 0's port1 enters PTX%PERIODIC state.
```

# The status of port 1 of STP instance 0 turns PERIODIC of the PTX state machines. In PTX%PERIODIC, PTX indicates the type of the state machines, while PERIODIC indicates the status of the state machines.

```
Apr 14 2009 13:19:11.650.4 Quidway MSTP/8/MEXS:Instance 0's port1 enters PTX%RSTP state.
```

# The status of port 1 of STP instance 0 turns RSTP of the PTX state machines. In PTX%RSTP, PTX indicates the type of the state machines, while RSTP indicates the status of the state machines.

```
Apr 14 2009 13:19:11.650.4 Quidway MSTP/8/PKT:
Port1(10GE1/0/2)  Send Packet(Length: 155)
ProtocolVersionID: 03
BPDUType: 02( RST BPDU )
Flags: 6c( DESIGNATED  Forwarding  Agreement )
CIST Root Identifier: 0.000b-09c9-6bac
CIST External Path Cost: 199999
CIST Bridge Identifier: 8192.00e0-fca4-9c2a
CIST Port Identifier: 128.259
Message Age: 1
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 116
CIST Regional Root Identifier: 8192.00e0-fca4-9c2a
CIST Internal Root Path Cost: 0
CIST Remaining Hops: 39
Instance: 0, 27, 35
```

# 10GE1/0/2 receives an MSTP BPDU packet. The peer port of 10GE1/0/2, namely, the port that sends the packet, is the designated port in the CIST region and the port status is Forwarding.

```
Apr 14 2009 13:19:11.650.4 Quidway MSTP/8/MSG:
InstanceID: 27
MstiFlags: 6c( DESIGNATED  Forwarding  Agreement )
MSTI Regional Root Identifier: 8192.00e0-fca4-9c2a
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 8192.00e0-fca4-9c2a
MSTI Port Priority: 128.259
MSTI Remaining Hops: 20
```

# The preceding display describes information about MSTI 27. The peer port of 10GE1/0/2, namely, the port that sends the packet, is the designated port in MSTI 27 and the port status is Forwarding.

```
Apr 14 2009 13:19:11.650.4 Quidway MSTP/8/MSG:
InstanceID: 35
MstiFlags: 6c( DESIGNATED  Forwarding  Agreement )
MSTI Regional Root Identifier: 8192.00e0-fca4-9c2a
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 8192.00e0-fca4-9c2a
MSTI Port Priority: 128.259
MSTI Remaining Hops: 20
```

# The preceding display describes information about MSTI 35. The peer port of 10GE1/0/2, namely, the port that sends the packet, is the designated port in MSTI 35 and the port status is Forwarding.

The fields of the preceding debugging information are described as follows.

Field	Description
ProtocolVersionID	Indicates the version of the protocol. The versions are as follows: ● 0x0: STP ● 0x2: RSTP ● 0x3: MSTP
BPDUType	Indicates the type of the BPDU packets. The types are as follows: ● 0x00: Configuration BPDU of STP ● 0x80: Topology Change Notification BPDU (TCN BPDU) of STP ● 0x02: Rapid Spanning Tree BPDU (RST BPDU) or Multiple Spanning Tree BPDU (MST BPDU)
Flags	Indicates the flag field of CIST. The flag field contains information about the port role and status.
CIST Root Identifier	Indicates the ID of the CIST root switch.
CIST External Path Cost	Indicates the cost of the CIST external path.
CIST Bridge Identifier	Indicates the ID of the CIST designated switch.
CIST Port Identifier	Indicates the ID of the port in the CIST.
Message Age	Indicates the age during which a BPDU packet can keep effective.
Max Age	Indicates the maximum age of the BPDU packet. After the maximum age, the link to the root switch is considered as faulty.
Hello Time	Indicates the timeout period of the Hello timer.

Field	Description
Forward Delay	Indicates the timeout period of the Forward Delay timer.
Version 1 Length	Indicates the length of the STP BPDU packet. Version 1 indicates STP. The value of this field is fixed 0.
Version 3 Length	Indicates the length of the MSTP BPDU packet. Version 3 indicates MSTP.
CIST Regional Root Identifier	Indicates the ID of the root switch in the CIST region.
CIST Internal Root Path Cost	Indicates the cost of the CIST internal path.
CIST Remaining Hops	Indicates the remaining hops of the BPDU packet in CIST.
InstanceID	Indicates the ID of the MSTI instance.
MstiFlags	Indicates the flag of the MSTI. This field contains information about the port role and status.
MSTI Regional Root Identifier	Indicates the ID of the root switch in the MSTI region.
MSTI Internal Root Path Cost	Indicates the cost of the MSTI internal path. The internal path is the path from the local port to the root switch of the MSTI region.
MSTI Bridge Priority	Indicates the priority of the designated switch in the MSTI region.
MSTI Port Priority	Indicates the priority of the designated port in the MSTI region.
MSTI Remaining Hops	Indicates the remaining hops of the BPDU packet in the MSTI region.

The following table describes the state machines.

No.	State Machine	Description
1	PIM	Port Information Machine
2	PPM	Port Protocol Machine
3	PRS	Port Role Select
4	PRT	Port Role Transition
5	PST	Port State Transition
6	PTX	Port Transmit

No.	State Machine	Description
7	TCM	Topology Change Machine

2.4.3.2 debugging stp event

Function

Using the **debugging stp event** command, you can enable the debugging of the STP events on a specified interface.

Using the **undo debugging stp event** command, you can disable the debugging of the STP events on a specified interface.

By default, the debugging of the STP events on a specified interface is disabled.

Format

debugging stp [**interface** *interface-type interface-number*] **event**

undo debugging stp [**interface** *interface-type interface-number*] **event**

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the interface type and number.	The interface type can be GE, 10GE, 40GE and Eth-Trunk.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging stp event** command enables the debugging function of the STP events on a specified interface, facilitating fault location.

If you do not specify an interface, the debugging of STP events is enabled or disabled on all the interfaces.

Example

# Enable the debugging of STP events on all the interfaces.

```
<HUAWEI> debugging stp event
Oct 10 14:49:06 2006 Quidway MSTP/8/PEVT:Port 6 occurs LINK DOWN event
```

# An event that the link turns Down occurs on Port 6. The possible cause is that the link is shut down or the cable is plug out.

```
Oct 10 14:49:06 2006 Quidway MSTP/8/PRS:Instance 0 Enters PRS Machine.
```

# Instance 0 turns into a certain state of the PRS state machines. The possible cause is that the network topology changes; thus, instance 0 needs to recalculate the topology.

```
Oct 10 14:49:06 2006 Quidway MSTP/8/PROLE:Instance 0's Port1 is selected as DESIGNATED role.
```

# Port 1 of instance 0 is selected as the designated port.

```
Oct 10 14:49:07 2006 Quidway MSTP/8/PROLE:Instance 0's Port7 is selected as ROOT role.
```

# Port 7 of instance 0 is selected as the root port.

```
Oct 10 14:49:07 2006 Quidway MSTP/8/PROLE:Instance 0's Port8 is selected as DESIGNATED role.
```

# Port 8 of instance 0 is selected as the designated port.

```
Oct 10 14:49:07 2006 Quidway MSTP/8/PROLE:Instance 0's Port12 is selected as DESIGNATED role.
```

# Port 12 of instance 0 is selected as the designated port.

```
Oct 10 14:49:07 2006 Quidway MSTP/8/MEXS:Instance 0's port1 enters PRT%ACTIVE_PORT state.
```

# The status of Port 1 of instance 0 turns ACTIVE_PORT of the PRT state machines. In PRT%ACTIVE_PORT, PRT indicates the type of the state machines, while ACTIVE_PORT indicates the status of the state machines.

```
Oct 10 14:49:07 2006 Quidway MSTP/8/MEXS:Instance 0's port7 enters PRT%ACTIVE_PORT state
```

# The status of Port 7 of instance 0 turns ACTIVE_PORT of the PRT state machines. In PRT%ACTIVE_PORT, PRT indicates the type of the state machines, while ACTIVE_PORT indicates the status of the state machines.

2.4.3.3 debugging stp instance event

Function

Using the **debugging stp instance event** command, you can enable event debugging of a specified STP instance.

Using the **undo debugging stp instance event** command, you can enable event debugging of a specified STP instance.

By default, the event debugging of a specified STP instance is disabled.

Format

debugging stp instance *instance-id* **event**

undo debugging stp instance *instance-id* **event**

Parameters

Parameter	Description	Value
instance-id	Specifies the ID of the Spanning Tree Protocol (STP) instance.	The value is an integer ranging from 0 to 4094, each process supports a maximum of 64 instances.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging stp instance event** command enables event debugging of a specified STP instance, facilitating fault location.

Example

# Enable event debugging of STP instance 0.

```
<HUAWEI> debugging stp instance 0 event
Oct 10 14:49:00 2006 HUAWEI MSTP/8/MEXS:Instance 0's port1 enters PTX%PERIODIC state.
Oct 10 14:49:01 2006 HUAWEI MSTP/8/MEXS:Instance 0's port1 enters PTX%RSTP state.
Oct 10 14:49:02 2006 HUAWEI MSTP/8/MEXS:Instance 0's port2 enters PTX%PERIODIC state.
Oct 10 14:49:03 2006 HUAWEI MSTP/8/MEXS:Instance 0's port2 enters PTX%RSTP state.
Oct 10 14:49:04 2006 HUAWEI MSTP/8/MEXS:Instance 0's port50 enters PTX%PERIODIC state.
Oct 10 14:49:05 2006 HUAWEI MSTP/8/MEXS:Instance 0's port50 enters PRT%ACTIVE_PORT state.
Oct 10 14:49:06 2006 HUAWEI MSTP/8/MEXS:port7 occurs SPEED CHANGE event
Oct 10 14:49:07 2006 HUAWEI MSTP/8/PRS:Instance 0 Enters PRS Machine.
```

# The preceding information shows that the port enters a certain state of the state machines. Take the following display as an example:

```
Oct 10 14:49:06 2006 HUAWEI MSTP /8/MEXS:Instance 0's port50 enters PIM%RECEIVED state.
```

This example indicates the status of Port 50 of instance 0 turns Received of the state machines. In PIM%RECEIVED, PIM indicates the type of the state machines, while RECEIVED indicates the status of the state machines.

The following table lists the state machines.

No.	State Machine	Description
1	PIM	Port Information Machine
2	PPM	Port Protocol Machine
3	PRS	Port Role Select
4	PRT	Port Role Transition
5	PST	Port State Transition

No.	State Machine	Description
6	PTX	Port Transmit
7	TCM	Topology Change Machine

2.4.3.4 debugging stp msti

Function

Using the **debugging stp msti** command, you can enable packet debugging of a specified MSTI.

Using the **undo debugging stp msti** command, you can disable packet debugging of a specified MSTI.

By default, the packet debugging of a specified MSTI is disabled.

Format

debugging stp msti { *instance-id1* [**to** *instance-id2*] } &<1-10>

undo debugging stp msti { *instance-id1* [**to** *instance-id2*] } &<1-10>

Parameters

Parameter	Description	Value
instance *instance-id*	Indicates the ID of the Multiple Spanning Tree Instance (MSTI).	The value is an integer ranging from 0 to 4094, each process supports a maximum of 64 instances.

Views

User view

Default Level

3: Management level

Usage Guidelines

When using this command, you can specify either a MSTI or a range of MSTIs.

Example

# Enable packet debugging of MSTI 1 and MSTI 2.

```
<HUAWEI> debugging stp msti 1 2
Oct 10 14:49:06 2006 Quidway MSTP/8/MSG:
InstanceID: 1
```

```
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 8192.00e0-fca4-9c2a
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 8192.00e0-fca4-9c2a
MSTI Port Priority: 16.268
MSTI Remaining Hops: 20
```

# The preceding display describes information about MSTI 1. The port that sends the packet is the designated port in MSTI 1 and the port status is Forwarding.

```
Oct 10 14:49:06 2006 Quidway MSTP/8/MSG:
InstanceID: 2
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 8192.00e0-fca4-9c2a
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 8192.00e0-fca4-9c2a
MSTI Port Priority: 32.268
MSTI Remaining Hops: 20
```

# The preceding display describes information about MSTI 2. The port that sends the packet is the designated port in MSTI 2 and the port status is Forwarding. The fields of the preceding debugging information are described as follows.

Field	Description
InstanceID	Indicates the ID of the MSTI.
MstiFlags	Indicates the flag of the MSTI. The flag field contains information about the port role and status.
MSTI Regional Root Identifier	Indicates the ID of the root switch in the MSTI region.
MSTI Internal Root Path Cost	Indicates the cost of the MSTI internal path. The internal path is the path from the local port to the root switch of the MSTI region.
MSTI Bridge Priority	Indicates the priority of the designated switch in the MSTI region.
MSTI Port Priority	Indicates the priority of the designated port in the MSTI region.
MSTI Remaining Hops	Indicates the remaining hops of the BPDU packet in the MSTI region.

2.4.3.5 debugging stp packet all

Function

Using the **debugging stp packet all** command, you can enable the debugging of BPDU packets on a specified interface.

Using the **undo debugging stp packet all** command, you can disable the debugging of BPDU packets on a specified interface.

By default, the debugging of BPDU packets on a specified interface is disabled.

Format

debugging stp [**interface** *interface-type interface-number*] **packet all**

undo debugging stp [**interface** *interface-type interface-number*] **packet all**

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the interface type and number.	The interface type can be 10GE, 40GE and Eth-Trunk.

Views

User view

Default Level

3: Management level

Usage Guidelines

If you run the **debugging stp all** command to enable the debugging of the MSTP module, you can run the **undo debugging stp all** command to disable the debugging of the MSTP module instead of running the **undo debugging stp packet all** command to disable the debugging of BPDU packets on a specified interface.

If you do not specify an interface, the debugging of BPDU packets on all the interfaces is enabled or disabled.

Example

# Enable the debugging of BPDU packets on all the interfaces.

```
<HUAWEI> debugging stp packet all
Oct 10 14:49:06 2006 Quidway MSTP/8/PKT:
Port6(10GE1/0/1)  Send Packet(Length: 1351)
ProtocolVersionID: 03
BPDUType: 02( RST BPDU )
Flags: 2c( DESIGNATED  Forwarding )
CIST Root Identifier: 0.000b-09c9-6bac
CIST External Path Cost: 199999
CIST Bridge Identifier: 8192.00e0-fca4-9c2a
CIST Port Identifier: 0.268
Message Age: 1
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 1312
CIST Regional Root Identifier: 8192.00e0-fca4-9c2a
CIST Internal Root Path Cost: 0
CIST Remaining Hops: 39
Instance: 0, 1, 2, 3,
Oct 10 14:49:06 2006 Quidway MSTP/8/MSG:
InstanceID: 1
```

```
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 8192.00e0-fca4-9c2a
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 8192.00e0-fca4-9c2a
MSTI Port Priority: 16.268
MSTI Remaining Hops: 20
.
Oct 10 14:49:06 2006 Quidway MSTP/8/MSG:
InstanceID: 2
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 8192.00e0-fca4-9c2a
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 8192.00e0-fca4-9c2a
MSTI Port Priority: 32.268
MSTI Remaining Hops: 20
```

# The preceding display shows the detailed contents of the BPDU packets sent by 10GE1/0/1. The display includes information about MSTI 1 and MSTI 2.

```
Oct 10 14:49:50 2006 Quidway MSTP/8/PKT:
Port7(10GE1/0/1)  Rcvd Packet(Length: 519)
ProtocolVersionID: 03
BPDUType: 02( RST BPDU )
Flags: 2c( DESIGNATED  Forwarding )
CIST Root Identifier: 0.000b-09c9-6bac
CIST External Path Cost: 0
CIST Bridge Identifier: 0.000b-09c9-6bac
CIST Port Identifier: 128.73
Message Age: 0
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 480
CIST Regional Root Identifier: 0.000b-09c9-6bac
CIST Internal Root Path Cost: 0
CIST Remaining Hops: 20
Instance: 0, 1, 2,
14, 15, 16
Oct 10 14:49:50 2006 Quidway MSTP/8/MSG:
InstanceID: 1
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 32768.000b-09c9-6bac
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 32768.000b-09c9-6bac
MSTI Port Priority: 128.73
MSTI Remaining Hops: 20
.
Oct 10 14:49:50 2006 Quidway MSTP/8/MSG:
InstanceID: 2
MstiFlags: 2c( DESIGNATED  Forwarding )
MSTI Regional Root Identifier: 32768.000b-09c9-6bac
MSTI Internal Root Path Cost: 0
MSTI Bridge Priority: 32768.000b-09c9-6bac
MSTI Port Priority: 128.73
MSTI Remaining Hops: 20
```

# The preceding display shows the detailed contents of the BPDU packets received by 10GE1/0/1. The display includes information about MSTI 1 and MSTI 2.

The following table describes the fields in the preceding debugging information:

Field	Description
ProtocolVersionID	Indicates the version of the protocol. The versions are as follows: ● 0x0: STP ● 0x2: RSTP ● 0x3: MSTP
BPDUType	Indicates the type of the BPDU packet. The types are as follows: ● 0x00: Configuration BPDU of STP ● 0x80: Topology Change Notification BPDU (TCN BPDU) of STP ● 0x02: Rapid Spanning-Tree BPDU (RST BPDU) or Multiple Spanning-Tree BPDU (MST BPDU)
Flags	Indicates the flag field of CIST. The flag field contains information about the port role and status.
CIST Root Identifier	Indicates the ID of the CIST root switch.
CIST External Path Cost	Indicates the cost of the CIST external path.
CIST Bridge Identifier	Indicates the ID of the designated switch in the CIST region.
CIST Port Identifier	Indicates the ID of the designated port in the CIST region.
Message Age	Indicates the age of the BPDU packet.
Max Age	Indicates the maximum age of the BPDU packet. After the maximum age, the link to the root switch is considered as faulty.
Hello Time	Indicates the timeout period of the Hello timer.
Forward Delay	Indicates the timeout period of the Forward Delay timer.
Version 1 Length	Indicates the length of the STP BPDU packet. Version 1 indicates STP. The value of this field is fixed 0.
Version 3 Length	Indicates the length of the MSTP BPDU packet. Version 3 indicates MSTP.
CIST Regional Root Identifier	Indicates the ID of the root switch in the CIST region.
CIST Internal Root Path Cost	Indicates the cost of the CIST internal path.
CIST Remaining Hops	Indicates the remaining hops of the BPDU packet in the CIST region.

Field	Description
InstanceID	Indicates the ID of the MSTI.
MstiFlags	Indicates the flag of the MSTI. The flag field contains information about the port role and status.
MSTI Regional Root Identifier	Indicates the ID of the root switch in the MSTI region.
MSTI Internal Root Path Cost	Indicates the cost of the MSTI internal path. The internal path is the path from the local port to the root switch of the MSTI region.
MSTI Bridge Priority	Indicates the priority of the designated switch in the MSTI region.
MSTI Port Priority	Indicates the priority of the designated switch in the MSTI region.
MSTI Remaining Hops	Indicates the remaining hops of the BPDU packet in the MSTI region.

2.4.3.6 debugging stp packet receive

Function

The **debugging stp packet receive** command enables debugging of BPDU packets received on the specified port.

The **undo debugging stp packet receive** command disables debugging of BPDU packets received on the specified port.

By Default, the debugging of BPDU packets received on the specified port is disabled.

Format

debugging stp [**interface** *interface-type interface-number*] **packet receive**

undo debugging stp [**interface** *interface-type interface-number*] **packet receive**

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the interface type and interface number.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging stp packet receive** command enables debugging of BPDU packets received on the specified port, facilitating fault location.

If no port is specified, the preceding commands enable or disable debugging of BPDU packets received on all ports.

Example

# Enable debugging of BPDU packets received on the specified port and display debugging information.

```
<HUAWEI> debugging stp interface 10GE 1/0/1 packet receive
Oct 10 14:49:10 2012 HUAWEI MSTP/8/PKT:
Port7(10GE1/0/1)  Rcvd Packet(Length: 519)
ProtocolVersionID: 03
BPDUType: 02( RST BPDU )
Flags: 2c( DESIGNATED  Forwarding )
CIST Root Identifier: 0.000b-09c9-6bac
CIST External Path Cost: 0
CIST Bridge Identifier: 0.000b-09c9-6bac
CIST Port Identifier: 128.73
Message Age: 0
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 480
CIST Regional Root Identifier: 0.000b-09c9-6bac
CIST Internal Root Path Cost: 0
CIST Remaining Hops: 20
Instance: 0, 1, 2,
```

Table 2-3 Description of the **debugging stp packet receive** command output

Field	Description
ProtocolVersionID	Protocol version. The values are as follows: ● 00: STP ● 02: RSTP ● 03: MSTP
BPDUType	Type of BPDU packets. The values are as follows: ● 00: Configuration BPDU of STP ● 80: Topology Change Notification BPDU (TCN BPDU) of STP ● 02: Rapid Spanning-Tree BPDU (RST BPDU) or Multiple Spanning-Tree BPDU (MST BPDU)
Flags	CIST flag, including information such as the role and status of this port

Field	Description
CIST Root Identifier	ID of the CIST root bridge
CIST External Path Cost	External path cost of the CIST
CIST Bridge Identifier	ID of the specified CIST switch
CIST Port Identifier	Specified port ID of this port in the CIST
Message Age	Indicates the age of the BPDU packet.
Max Age	Indicates the maximum age of the BPDU packet. Beyond the maximum age, the link of the root bridge is considered faulty.
Hello Time	Value of the hello timer
Forward Delay	Value of the forward delay timer
Version 1 Length	Length of version 1 BPDU, namely length of STP BPDU. The value is permanently **0**.
Version 3 Length	Length of version 3 BPDU, namely length of MSTP BPDU
CIST Regional Root Identifier	ID of the domain root switch of the CIST, namely ID of IST master
CIST Internal Root Path Cost	Internal path cost of CIST
CIST Remaining Hops	Remaining hops of BPDU packets in the CIST
Instance	ID of an MSTI instance

2.4.3.7 debugging stp packet send

Function

The **debugging stp packet send** command enables debugging of BPDU packets sent on the specified port.

The **undo debugging stp packet send** command disables debugging of BPDU packets sent on the specified port.

By Default, the debugging of BPDU packets sent on the specified port is disabled.

Format

debugging stp [**interface** *interface-type interface-number*] **packet send**

undo debugging stp [**interface** *interface-type interface-number*] **packet send**

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the interface type and interface number.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging stp packet send** command enables debugging of BPDU packets sent on the specified port, facilitating fault location.

If no port is specified, the preceding commands enable or disable debugging of BPDU packets sent on all ports.

Example

# Enable debugging of BPDU packets sent on the specified port and display debugging information.

```
<HUAWEI> debugging stp interface 10GE 1/0/1 packet send
Oct 10 14:49:06 2012 HUAWEI MSTP/8/PKT:
Port6(10GE1/0/1)  Send Packet(Length: 1351)
ProtocolVersionID: 03
BPDUType: 02( RST BPDU )
Flags: 2c( DESIGNATED  Forwarding )
CIST Root Identifier: 0.000b-09c9-6bac
CIST External Path Cost: 199999
CIST Bridge Identifier: 8192.00e0-fca4-9c2a
CIST Port Identifier: 0.268
Message Age: 1
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 1312
CIST Regional Root Identifier: 8192.00e0-fca4-9c2a
CIST Internal Root Path Cost: 0
CIST Remaining Hops: 39
Instance: 0, 1, 2, 3
```

Table 2-4 Description of the **debugging stp packet send** command output

Field	Description
ProtocolVersionID	Protocol version. The values are as follows: • 00: STP • 02: RSTP • 03: MSTP

Field	Description
BPDUType	Type of BPDU packets. The values are as follows: ● 00: Configuration BPDU of STP ● 80: Topology Change Notification BPDU (TCN BPDU) of STP ● 02: Rapid Spanning-Tree BPDU (RST BPDU) or Multiple Spanning-Tree BPDU (MST BPDU)
Flags	CIST flag, including information such as the role and status of this port
CIST Root Identifier	ID of the CIST root bridge
CIST External Path Cost	External path cost of the CIST
CIST Bridge Identifier	ID of the specified CIST switch
CIST Port Identifier	Specified port ID of this port in the CIST
Message Age	Indicates the age of the BPDU packet
Max Age	Indicates the maximum age of the BPDU packet. Beyond the maximum age, the link of the root bridge is considered faulty.
Hello Time	Value of the hello timer
Forward Delay	Value of the forward delay timer
Version 1 Length	Length of version 1 BPDU, namely length of STP BPDU. The value is permanently **0**.
Version 3 Length	Length of version 3 BPDU, namely length of MSTP BPDU
CIST Regional Root Identifier	ID of the domain root switch of the CIST, namely ID of IST master
CIST Internal Root Path Cost	Internal path cost of CIST
CIST Remaining Hops	Remaining hops of BPDU packets in the CIST
Instance	ID of an MSTI instance

2.4.3.8 debugging stp process

Function

The **debugging stp process** command enables debugging of the specified MSTP process.

The **undo debugging stp process** command disables debugging of the specified MSTP process.

By default, the debugging of the specified MSTP process is disabled.

Format

debugging stp process *process-id*

undo debugging stp process *process-id*

Parameters

Parameter	Description	Value
process *process-id*	Specifies the ID of an MSTP process. The value is an integer ranging from 1 to 256.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging stp process** command enables debugging of the specified MSTP process. This command must be used together with other debugging commands.

Example

# Enable debugging of MSTP process 1.

```
<HUAWEI> debugging stp process 1
```

2.4.3.9 debugging stp v-stp

Function

The **debugging stp v-stp** command enables V-STP debugging.

The **undo debugging stp v-stp** command disables V-STP debugging.

By default, V-STP debugging is disabled.

Format

debugging stp v-stp packet

undo debugging stp v-stp packet

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

To locate a V-STP problem, run the **debugging stp v-stp** command to enable V-STP debugging. After the problem is resolved, run the **undo debugging stp v-stp** command to disable V-STP debugging.

Example

# Enable V-STP debugging.

```
<HUAWEI> debugging stp v-stp packet
May 18 2015 13:58:22.052 HUAWEI %%01MSTP/7/MSTP_DEBUG_VSTP_PACKET(d):CID=0x80542722;
Port7 Send V-STP Packet(Length: 85)
ProtocolVersionID                : 00
BPDUType                         : ff
Flags                            : 0c
System MAC                       : 38af-7611-1200
Root Port m-lag Identifier       : Invalid
CIST Root Identifier             : 32768.38af-7611-1200
CIST External Path Cost          : 0
CIST Regional Root Identifier    : 32768.38af-7611-1200
CIST Internal Root Path Cost     : 0
CIST Bridge Identifier           : 32768.38af-7611-1200
CIST Port Identifier             : 0.0
Receive Port Identifier          : 1
Message Age                      : 0
Max Age                          : 20
Hello Time                       : 2
Forward Delay                    : 15
CIST Remaining Hops              : 20
```

Table 2-5 Description of the **debugging stp v-stp packet** command output

Item	Description
ProtocolVersionID	Protocol version ID (00 indicating STP)
BPDUType	BPDU type: ● 00: STP's configuration BPDU ● 80: STP's topology change notification (TCN) BPDU ● 02: Rapid Spanning Tree (RST) BPDU or Multiple Spanning Tree (MST) BPDU ● ff: V-STP proprietary protocol BPDU

Item	Description
Flags	CIST flag, including the local interface's role and status information
System MAC	System MAC address
Root Port m-lag Identifier	Root interface's M-LAG ID ● Invalid: The remote root interface is not an M-LAG interface.
CIST Root Identifier	CIST root bridge ID
CIST External Path Cost	CIST's external path cost
CIST Regional Root Identifier	ID of CIST's regional root switch or CIST master's ID
CIST Internal Root Path Cost	CIST's internal path cost
CIST Bridge Identifier	CIST's designated bridge ID
CIST Port Identifier	Designated ID of the local interface in the CIST
Receive Port Identifier	ID of the receive interface
Message Age	Time to live (TTL) of BPDUs
Max Age	Maximum TTL of BPDUs (If the maximum lifetime elapses, the link to the root bridge fails.)
Hello Time	Hello timer value
Forward Delay	Forward Delay timer value
CIST Remaining Hops	Remaining hops of BPDUs in CIST

2.4.3.10 debugging vbst

Function

The **debugging vbst** command enables debugging of the VBST module.

The **undo debugging vbst** command disables debugging of the VBST module.

By default, all debugging of the VBST module is disabled.

NOTE

CE5880EI, CE6863, CE6863K, CE6881E, CE6820, CE6881, CE6881K, and CE6880EI do not support this command.

Format

debugging vbst { **packet** { **receive** | **send** } | **ifm** | **info** | **error** | **event** | **instance** | **vlan** } [**vlan-id** *vlan-id*] [**port-id** *port-id*]

undo debugging vbst { **packet** { **receive** | **send** } | **ifm** | **info** | **error** | **event** | **instance** | **vlan** }

Parameters

Parameter	Description	Value
packet	Data packet Information	-
receive	Inbound packet	-
send	Outbound packet	-
ifm	Interface Management Information	-
info	Information	-
error	Error information	-
event	Event information	-
instance	Instance information	-
vlan	VLAN information	-
vlan-id *vlan-id*	Specifies the ID of the VLAN whose information will be displayed	The value is an integer ranging from 1 to 4094.
port-id *port-id*	Specifies the ID of the port whose information will be displayed	The value is an integer ranging from 1 to 2304.

Views

Diagnostic view

Default Level

3: Management level

Usage Guidelines

When a fault occurs on the network, run the **debugging vbst** command to enable debugging of the VBST module to locate the fault.

Example

# Enable the debugging of events in the VBST module.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging vbst event
Nov 20 2015 17:22:32.764 Huawei %%01VBST/7/VBST_DEBUG(d):CID=0x8305042f; [VBST]: PRT-
>ACTIVE_PORT.(vrId=0, vlanId=1, portId=66)


Nov 20 2015 17:22:32.764 Huawei %%01VBST/7/VBST_DEBUG(d):CID=0x8305042f; [VBST]: PRT:ALL
IfNodeSynced.(vrId=0, vlanId=1, ifnode
id=257)
```

Table 2-6 Description of the **debugging vbst event** command output

Item	Description
CID	Subgroup ID
PRT	State machines ACTIVE_PORT: the status of interface is active. ALL IfNodeSynced: all the interfaces are Synced.
vrId	VS number
valnId	VLAN ID
portId	Port ID

2.4.4 GVRP Debugging Commands

2.4.4.1 debugging gvrp

Function

The **debugging gvrp** command enables GVRP debugging.

The **undo debugging gvrp** command disables GVRP debugging.

By default, GVRP debugging is disabled.

Format

debugging gvrp state [**interface** *interface-type interface-number* [**vlan** *vlan-id*]]

undo debugging gvrp state [**interface** *interface-type interface-number* [**vlan** *vlan-id*]]

debugging gvrp packet { **receive** | **transmit** } [**interface** *interface-type interface-number*]

undo debugging gvrp packet { **receive** | **transmit** } [**interface** *interface-type interface-number*]

debugging gvrp all

undo debugging gvrp all

debugging gvrp error

undo debugging gvrp error

debugging gvrp info

undo debugging gvrp info

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
vlan *vlan-id*	Specifies a VLAN ID.	-
receive	Specifies received GVRP PDUs.	-
transmit	Specifies sent GVRP PDUs.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable GVRP state debugging, run the **debugging gvrp state** command.

To enable GVRP PDU debugging, run the **debugging gvrp packet** command.

To enable all GVRP debugging functions, run the **debugging gvrp all** command.

To enable GVRP error debugging, run the **debugging gvrp error** command.

To enable GVRP message debugging, run the **debugging gvrp info** command.

Example

# Enable all GVRP debugging functions.

```
<HUAWEI> debugging gvrp all
```

2.4.5 ERPS (G.8032) Debugging Commands

2.4.5.1 debugging erps

Function

The **debugging erps** command enables the debugging of the ERPS module.

The **undo debugging erps** command disables the debugging of the ERPS module.

By default, the debugging of the ERPS module is disabled.

Format

debugging erps { **all** | **error** | **message** } [**interface** *interface-type interface-number* | **ring** *ring-id*]

undo debugging erps { **all** | **error** | **message** } [**interface** *interface-type interface-number* | **ring** *ring-id*]

debugging erps fsm [**ring** *ring-id*]

undo debugging erps fsm [**ring** *ring-id*]

debugging erps pdu [**receive** | **transmit**] [**interface** *interface-type interface-number* | **ring** *ring-id*]

undo debugging erps pdu [**receive** | **transmit**] [**interface** *interface-type interface-number* | **ring** *ring-id*]

Parameters

Parameter	Description	Value
all	Indicates all debugging functions.	-
error	Indicates the debugging of errors.	-
message	Indicates the debugging of messages.	-
interface *interface-type interface-number*	Indicates the debugging of a specified interface.	-
ring *ring-id*	Indicates the debugging of a specified ERPS ring.	The value is an integer ranging from 1 to 255.
fsm	Indicates the debugging of the state machine.	-
pdu	Indicates the debugging of R-APS PDUs.	-
receive	Indicates the debugging of R-APS PDUs received.	-

Parameter	Description	Value
transmit	Indicates the debugging of R-APS PDUs sent.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To check the debugging information on the ERPS module, run the **debugging erps** command. The debugging information facilitates fault locating and device maintenance.

Example

# Enable the debugging of the ERPS module.

```
<HUAWEI> debugging erps pdu
Nov 13 2013 19:24:19.885 HUAWEI %%01ERPS/7/
ERPS_DEBUG_RINGANDINTF_PDU_SEND(d):CID=0x80bd273f; ring 1, interface 10ge 1/0/1 Send ERPS packet
successfully, type = NRRB.
```

2.5 00EIP Service Debugging Commands

2.5.1 IPv4 Debugging Commands

2.5.1.1 debugging ip packet

Function

The **debugging ip packet** command enables the debugging of IP packets.

The **undo debugging ip packet** command disables the debugging of IP packets.

By default, the debugging of IP packets is disabled.

Format

debugging ip { **packet** [**error**] [**min-length** *min-length*] [**max-length** *max-length*] [**source** *src-ip*] [**destination** *dst-ip*] [**interface** *interface-type interface-number*] [**verbose** [*verbose-length*]] [**number** *packet-number*] | **icmp** [**verbose** [*verbose-length*]] }

undo debugging ip { **packet** | **icmp** }

Parameters

Parameter	Description	Value
packet	Indicates IP packets.	-
error	Displays information about IP error packets.	-
min-length *min-length*	Specifies the minimum length of IP packets.	The value is an integer ranging from 1 to 65535.
max-length *max-length*	Specifies the maximum length of IP packets.	The value is an integer ranging from 1 to 65535.
source *src-ip*	Specifies the source IP address of a specified IP packet.	The value is in dotted decimal notation.
destination *dst-ip*	Specifies the destination address of a specified IP packet.	The value is in dotted decimal notation.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
number *packet-number*	Specifies the number of debugged IP packets.	The value is an integer ranging from 1 to 65535. The default value is 10.
icmp	Indicates ICMP packets.	-
verbose *verbose-length*	Specifies the length of detailed information about IP packets. The hexadecimal detailed information about IP packets are printed based on length. **NOTE** If the actual packet length is greater than the specified length, the specified length is displayed. If the actual packet length is smaller than the specified length, the actual packet length is displayed.	The value is an integer ranging from 1 to 64, in bytes. The default value is 20.

Views

User view

Default Level

3: Management level

Usage Guidelines

To locate problems that occur during IP packet processing, you can run the **debugging ip packet** command to query debugging information about IP packets.

Example

# Enable the debugging of IP packets, and the following debugging information is displayed.

```
<HUAWEI> debugging ip packet
2011-08-12 07:16:41 HUAWEI %%01PP4/7/ip_packet(d):CID=6694683;Receiving, ifIndex = 4, vrf = 0, eventId
= 0, flag = 0x200, version = 4, headlen = 5, tos = 0, pktlen = 0x4e, pktid = 0xe1b7, offset = 0x0, ttl = 128,
protocol = 17, checksum = 0xca, s = 172.16.255.254, d = 172.16.255.255

2011-08-12 07:16:41 HUAWEI %%01PP4/7/ip_packet(d):CID=6694683;Discarding, Reason = Drop packet for
ICMP security(1073), ifIndex = 4, vrf = 0, eventId = 0, flag = 0x100200, version = 4, headlen = 5, tos = 0,
pktlen = 0x4e00, pktid = 0xe1b7, offset = 0x0, ttl = 128, protocol = 17, checksum = 0xca, s = 172.16.255.254,
d = 172.16.255.255

2011-08-12 07:16:42 HUAWEI %%01PP4/7/ip_packet(d):CID=6694683;Receiving, ifIndex = 4, vrf = 0, eventId
= 0, flag = 0x200, version = 4, headlen = 5, tos = 0, pktlen = 0x4e, pktid = 0xe1b8, offset = 0x0, ttl = 128,
protocol = 17, checksum = 0xc9, s = 172.16.255.254, d = 172.16.255.255
```

# Display detailed information about the IP packet with packet length of 64.

```
<HUAWEI> debugging ip packet verbose 64
2011-08-12 07:17:25 HUAWEI %%01PP4/7/ip_packet(d):CID=6694683;Receiving, ifIndex = 4, vrf = 0, eventId
= 0, flag = 0x200, version = 4, headlen = 5, tos = 0, pktlen = 0x4e, pktid = 0xe1fb, offset = 0x0, ttl = 128,
protocol = 17, checksum = 0x86, s = 172.16.255.254, d = 172.16.255.255
Memory (IPv4 Pkt):
      4500 004e e1fb 0000 8011 0086 ac0f fffe
      ac0f ffff 0089 0089 003a 586c c865 0110
      0001 0000 0000 0000 2045 4444 4144 4443
      4e44 4244 4144 4344 4a43 4e46 4445 4d45
      5046 4544 4244 4241 4100 0020 0001
```

Table 2-7 Description of the debugging ip packet command output

Item	Description
ip_packet(d)	IP packets
CID	CID of the current output information
Receiving	Received packets
Discarding	Discarded packets
Sending	Sent packets
ifIndex	Interface index
vrf	VPN ID
eventId	Event ID
flag	Flag
version	Version number

Item	Description
headlen	Packet header length
tos	Type of Service (ToS)
pktlen	Packet length
pktid	Packet identification
offset	Fragment offset value
ttl	Time To Live (TTL)
protocol	Protocol number
checksum	Checksum
s	Source IP address
d	Destination IP address
Memory (IPv4 Pkt)	IPv4 packet displayed in hexadecimal notation

2.5.1.2 debugging rawip

Function

The **debugging rawip** command enables debugging of RAWIP packets and outputs debugging information.

The **undo debugging rawip** command disables debugging of RAWIP packets.

By default, the debugging of RAWIP packets is disabled.

Format

debugging rawip packet [**src-ip** *ipv4-address*] [**dest-ip** *ipv4-address*] [**protocol** *protocol-number*] [**verbose** *packet-length*] [**socket-id** *socket-id*]

undo debugging rawip packet [**src-ip** *ipv4-address*] [**dest-ip** *ipv4-address*] [**protocol** *protocol-number*] [**verbose** *packet-length*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Outputs a packet.	-
src-ip *ipv4-address*	Specifies the source address so that packets with the same source address are filtered.	The value is in dotted decimal notation.

Parameter	Description	Value
dest-ip *ipv4-address*	Specifies the destination address so that packets with the same destination address are filtered.	The value is in dotted decimal notation.
protocol *protocol-number*	Specifies the protocol number so that packets with the same protocol number are filtered.	The value is an integer ranging from 0 to 255.
verbose	Displays the detailed information about a packet.	-
packet-length	Displays the length of the detailed information about a packet.	The value is an integer ranging from 0 to 64.
socket-id *socket-id*	Specifies the Socket ID.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging rawip** command enables debugging of the RAWIP packets sent and received by the host so that you can view sending and receiving of ping packets or diagnose interworking between OSPF connections. The debugging information helps locate faults.

Example

# Enable debugging of RAWIP packets.

```
<HUAWEI> debugging rawip packet
<HUAWEI> terminal debugging
```

# Detect that the host sends a RAWIP packet.

```
Aug  4 2011 11:55:30 HUAWEI %%01SOCKET/7/debug_rawip_packet(d):CID=0x806527
49;
870: Output: pid = 6629141, socketid = 36, protocol = 58, ifindex = 2,
src = ::1, dst = ::1, datelen = 64
```

# Detect that the host receives an IPv6 RAWIP packet.

```
 Aug  4 2011 09:06:12 HUAWEI %%01SOCKET/7/debug_rawip_packet(d):CID=0x806527
49;
220: Input: pid = 6629141, socketid = 35, protocol = 1, ifindex = 0,
src = 127.0.0.1, dst = 127.0.0.1, datelen = 84
```

Table 2-8 Description of the **debugging rawip** command output

Item	Description
pid	Path ID
socketid	Socket ID
protocol	Protocol number
ifindex	Index of an interface
src	Source IP address
dst	Destination IP address
datelen	Data length

2.5.1.3 debugging rawlink

Function

The **debugging rawlink** command enables debugging of RAWLINK packets and outputs debugging information.

The **undo debugging rawlink** command disables debugging of RAWLINK packets.

By default, the debugging of RAWLINK packets is disabled.

Format

debugging rawlink packet [**src-mac** *mac-address*] [**dest-mac** *mac-address*] [**verbose** *packet-length*] [**socket-id** *socket-id*]

undo debugging rawlink packet [**src-mac** *mac-address*] [**dest-mac** *mac-address*] [**verbose** *packet-length*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Debugs a packet.	-
src-mac *mac-address*	Specifies the source physical address so that packets with the same source physical address are filtered.	-
dest-mac *mac-address*	Specifies the destination physical address so that packets with the same destination physical address are filtered.	-

Parameter	Description	Value
verbose	Displays detailed information about the specified packet.	-
packet-length	Displays the length of the detailed information about a packet.	The value is an integer ranging from 0 to 64.
socket-id *socket-id*	Specifies the Socket ID so that packets with the same Socket ID are filtered.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging rawlink** command enables debugging of RAWLINK packets and outputs debugging information about the RAWLINK packets sent and received by the host. For example, to diagnose interworking between IS-IS connections, run the **debugging rawlink** command.

Example

# Enable debugging of RAWLINK packets.

```
<HUAWEI> debugging rawlink packet
<HUAWEI> terminal debugging
```

# Detect that the host sends a RAWLINK packet.

```
2037-11-24 22:03:37 RT5 %%01SOCKET/7/debug_rawlink_packet(d):CID=2154112826
;
820: Output: pid = 6629167, socketid = 21748, ifindex = 375,
SRC MAC[0 0 0 0 0 0].DST MAC[9 0 2b 0 0 5].
```

# Detect that the host receives a RAWLINK packet.

```
2037-11-24 22:03:37 RT5 %%01SOCKET/7/debug_rawlink_packet(d):CID=2154112826
;
280: Input: pid = 6629167, socketid = 21725, ifindex = 352,
SRC MAC[28 6e d4 51 4e a].DST MAC[9 0 2b 0 0 5].
```

Table 2-9 Description of the **debugging rawlink** command output

Item	Description
pid	Path ID

Item	Description
socketid	Socket ID
ifindex	Index of an interface
SRC MAC	Source MAC address
DST MAC	Destination MAC address

2.5.1.4 debugging tcp

Function

The **debugging tcp** command enables debugging of TCP packets and outputs debugging information.

The **undo debugging tcp** command disables debugging of TCP packets.

By default, the debugging of TCP packets is disabled.

Format

debugging tcp packet [**src-ip** *ipv4-address*] [**src-port** *port-number*] [**dest-ip** *ipv4-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

undo debugging tcp packet [**src-ip** *ipv4-address*] [**src-port** *port-number*] [**dest-ip** *ipv4-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Debugs a packet.	-
src-ip *ipv4-address*	Displays packets of the specified source IP address.	The value is in dotted decimal notation.
src-port *port-number*	Displays packets of the specified source TCP port number.	The value is an integer ranging from 0 to 65535.
dest-ip *ipv4-address*	Displays packets of the specified destination IP address.	The value is in dotted decimal notation.
dest-port *port-number*	Displays packets of the specified destination TCP port number.	The value is an integer ranging from 0 to 65535.

Parameter	Description	Value
socket-id *socket-id*	Displays the data length of a TCP packet.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging tcp** command enables debugging of TCP packets and outputs debugging information about the TCP packets sent and received by the host. For example, to detect interworking between BGP connections and protocol interworking between MPLS LDP neighbors, run the **debugging tcp** command.

Example

# Enable debugging of TCP packets.

```
<HUAWEI> debugging tcp packet
<HUAWEI> terminal debugging
```

# Detect that the host sends a TCP packet.

```
2037-11-24 22:07:03 RT5 %%01SOCKET/7/debug_tcp_packe(d):CID=2154103793;
TCP debug packet information:
450: Output: pid = 6620139, socketid = 7925,
(src = 10.2.217.5:23, dst = 10.2.1.13:3356, seq = 1198729414, ack = 2793145850
, datalen = 42, optlen = 20 ,
flag = ACK PUSH , window = 65364, ttl = 64, tos = 192)
```

# Detect that the host receives a TCP packet.

```
2037-11-24 22:07:03 RT5 %%01SOCKET/7/debug_tcp_packe(d):CID=2154103793;
TCP debug packet information:
450: Input: pid = 6620139, socketid = 7925,
(src = 10.2.1.13:3356, dst = 10.2.217.5:23, seq = 2793145848, ack = 1198729414
, datalen = 42, optlen = 20 ,
flag = ACK PUSH , window = 64571, ttl = 127, tos = 0)
```

Table 2-10 Description of the **debugging tcp** command output

Item	Description
pid	Path ID
socketid	Socket ID
src	Source IP address

Item	Description
dst	Destination IP address
seq	Sequence numbers of ARP entries
ack	Acknowledgment number
datalen	Data length
optlen	Option length
flag	Flag bit
window	Size of sliding window
ttl	Time to live
tos	Priority

2.5.1.5 debugging udp

Function

The **debugging udp** command enables debugging of UDP packets and outputs debugging information.

The **undo debugging udp** command disables debugging of UDP packets.

By default, the debugging of UDP packets is disabled.

Format

debugging udp packet [**src-ip** *ipv4-address*] [**src-port** *port-number*] [**dest-ip** *ipv4-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

undo debugging udp packet [**src-ip** *ipv4-address*] [**src-port** *port-number*] [**dest-ip** *ipv4-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Debugs a packet.	-
src-ip *ipv4-address*	Displays packets of the specified source IP address.	The value is in dotted decimal notation.
src-port *port-number*	Displays packets of the specified source UDP port number.	The value is an integer ranging from 0 to 65535.

Parameter	Description	Value
dest-ip *ipv4-address*	Displays packets of the specified destination IP address.	The value is in dotted decimal notation.
dest-port *port-number*	Displays packets of the specified destination UDP port number.	The value is an integer ranging from 0 to 65535.
socket-id *socket-id*	Displays the data length of a UDP packet.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging udp** command enables debugging of UDP packets sent and received by the host and outputs debugging information so that you can locate problems. For example, to detect sending and receiving of BFD packets or Trace packets, run the **debugging udp** command.

Example

# Enable debugging of UDP packets.

```
<HUAWEI> debugging udp packet
<HUAWEI> terminal debugging
```

# Detect that the host sends a UDP packet.

```
2037-11-24 22:12:46 RT5 %%01SOCKET/7/debug_udp_packet(d):CID=2154103793;
UDP debug packet information:
650: Output: pid = 6620139,socketid = 7209, ifindex = 25,
(Time = 75, src = 10.5.5.5:40000, dst = 10.0.0.2:1, datalen = 76)
```

# Detect that the host receives a UDP packet.

```
2037-11-24 22:12:46 RT5 %%01SOCKET/7/debug_udp_packet(d):CID=2154103793;
UDP debug packet information:
760: Input: pid = 6620139,socketid = 6804, ifindex = 1176,
(Time = 193, src = 10.0.54.1:56206, dst = 10.0.54.2:4784, datalen = 52)
```

Table 2-11 Description of the **debugging udp** command output

Item	Description
pid	Path ID

Item	Description
socketid	Socket ID
ifindex	Index of an interface
Time	Time when the host sent or received UDP packets
src	Source IP address
dst	Destination IP address
datelen	Data length

2.5.1.6 debugging tcp event

Function

The **debugging tcp event** command enables debugging of TCP events and outputs debugging information.

The **undo debugging tcp event** command disables debugging of TCP events.

By default, the debugging of TCP events is disabled.

Format

debugging tcp event [**local-ip** *local-ip*] [**local-port** *local-port*] [**remote-ip** *remote-ip*] [**remote-port** *remote-port*] [**socket-id** *socket-id*]

undo debugging tcp event [**local-ip** *local-ip*] [**local-port** *local-port*] [**remote-ip** *remote-ip*] [**remote-port** *remote-port*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
local-ip *local-ip*	Specifies the IPv4 address of the local end.	The value is in dotted decimal notation.
local-port *local-port*	Specifies the TCP port number of the local end.	The value is an integer ranging from 0 to 65535.
remote-ip *remote-ip*	Specifies the IPv4 address of the remote end.	The value is in dotted decimal notation.
remote-port *remote-port*	Specifies the TCP port number of the remote end.	The value is an integer ranging from 0 to 65535.
socket-id *socket-id*	Specifies the Socket ID.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging tcp event** command enables debugging of TCP events and outputs debugging information so that you can detect the status of a TCP packet when the TCP packet is abnormal. For example, to detect change of TCP status change during establishment of a three-way handshake or closing of a four-way handshake, run the **display tcp event** command.

Example

# Enable debugging of TCP events.

```
<HUAWEI> debugging tcp event
<HUAWEI> terminal debugging
```

# Detect that the host initiates a three-way handshake and the TCP status changes from CLOSED to SYN_SENT.

```
Aug 25 2011 09:44:49 HUAWEI %%01SOCKET/7/debug_tcp_event(d):CID=0x80652795;
TCP state changed from [CLOSED] to [SYN_SENT]
(Time = 2011-8-25:9:44:49:684 ,Task name = XXXX ,Pid = 0x6503F6, Socket ID = 1)
```

# Detect that the three-way handshake initiated by the host is successful and the TCP status changes from SYN_SENT to ESTABLISHED.

```
Aug 25 2011 09:44:49 HUAWEI %%01SOCKET/7/debug_tcp_event(d):CID=0x80652795;
TCP state changed from [SYN_SENT] to [ESTABLISHED]
(Time = 2011-8-25:9:44:49:884 ,Task name = XXXX ,Pid = 0x6503F6, Socket ID = 1)
```

Table 2-12 Description of the **debugging tcp event** command output

Item	Description
Time	Time when the TCP status was changed
Task name	Task name
Pid	Component PID
Socket ID	Socket instance ID

2.5.2 ARP Debugging Commands

2.5.2.1 debugging arp

Function

The **debugging arp** command enables ARP debugging and displays the debugging information.

The **undo debugging arp** command disables ARP debugging.

By default, all ARP debugging functions are disabled.

Format

debugging arp { **process** | **error** | **event** } [**interface** *interface-type interface-number* | **vlan** *vlan-id* | **bridge-domain** *bd-id*]

undo debugging arp { **process** | **error** | **event** } [**interface** *interface-type interface-number* | **vlan** *vlan-id* | **bridge-domain** *bd-id*]

debugging arp process ip *ip-address*

undo debugging arp process ip *ip-address*

Parameters

Parameter	Description	Value
process	Enables the process debugging.	-
error	Enables the error debugging.	-
event	Enables the event debugging.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
vlan *vlan-id*	Specifies the ID of a VLAN.	The value is an integer ranging from 1 to 4094.
bridge-domain *bd-id*	Specifies the ID of a bridge domain. **NOTE** The CE5810EI, CE5850EI, CE5850HI, CE5855EI, CE6810LI, CE6810EI, CE6820, and CE6850EI do not support this parameter.	The value is an integer ranging from 1 to 16777215.
ip *ip-address*	Specifies the IP address.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Usage Guidelines

Debugging information includes ARP errors, events, and information about packet receiving, sending, and processing. When an ARP fault occurs, run the **debugging arp** command to output the debugging information. The output information helps you quickly locate the fault.

Example

# Enable ARP error debugging.

```
<HUAWEI>debugging arp error
Apr 25 2013 09:35:01.462 HUAWEI %%01ARP/7/ARP_DBG_Error(d):CID=0x807703fc;The ARP Error : Discard
the packet because ifnet down. ifindex = 0x3
```

# Enable ARP process debugging.

```
<HUAWEI>debugging arp process ip 1.1.1.1
```

2.5.2.2 debugging arp packet

Function

The **debugging arp packet** command enables debugging of Address Resolution Protocol (ARP) packets to view the debugging information, and locate and analyze faults.

The **undo debugging arp packet** command disables debugging of ARP packets.

By default, the debugging function of Address Resolution Protocol (ARP) packets is disabled.

Format

debugging arp packet [**interface** *interface-type interface-number* [**ip** *ip-address*] | **vlan** *vlan-id* | **bridge-domain** *bd-id*]

undo debugging arp packet [**interface** *interface-type interface-number* [**ip** *ip-address*] | **vlan** *vlan-id* | **bridge-domain** *bd-id*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the interface type and interface number.	-

Parameter	Description	Value
ip *ip-address*	Specifies the IP address of an interface.	The value is in dotted decimal notation.
vlan *vlan-id*	Specifies the ID of a VLAN.	The value is an integer ranging from 1 to 4094.
bridge-domain *bd-id*	Specifies the ID of a bridge domain. **NOTE** The CE5810EI, CE5850EI, CE5850HI, CE5855EI, CE6810LI, CE6810EI, CE6820, and CE6850EI do not support this command.	The value is an integer ranging from 1 to 16777215.

Views

User view

Default Level

3: Management level

Usage Guidelines

Debugging affects system performance. Therefore, run the **undo debugging all** command when debugging is complete. In addition, when the CPU usage is close to 100%, debugging of ARP packets may cause board reset. Therefore, exercise caution when using this command.

Example

# Enable debugging of ARP packets, and display the debugging information on the terminal.

```
<HUAWEI> debugging arp packet
```

# Send an ARP response packet.

```
Apr 25 2013 09:35:01.462 HUAWEI %%01ARP/7/ARP_DBG_PACKET_SND(d):CID=0x80770413;Send an ARP Packet, operation : 2, send_eth_addr : dcd2-fc20-c295, sender_ip_addr : 10.10.10.1, target_eth_addr : 00e0-fc94-e46a, target_ip_addr : 10.10.10.2
```

# Receive an ARP request packet.

```
Apr 25 2013 09:35:01.462 HUAWEI %%01ARP/7/ARP_DBG_PACKET_RCV(d):CID=0x80770413;Receive an ARP Packet, operation : 1, send_eth_addr : 00e0-fc94-e46a, sender_ip_addr : 10.10.10.2, target_eth_addr : 0000-0000-0000, target_ip_addr : 10.10.10.1
```

Table 2-13 Description of the **debugging arp packet** command output

Field	Description
Receive an ARP Packet	An ARP packet is received.
Send an ARP Packet	An ARP packet is sent.
operation	Packet type. The types of ARP packets are as follows: ● 1: ARP request packet ● 2: ARP response packet
sender_eth_addr	Ethernet address of the sender, that is, the Media Access Control (MAC) address of the sender.
sender_ip_addr	IP address of the sender.
target_eth_addr	Ethernet address of the recipient, that is, the MAC address of the recipient.
target_ip_addr	IP address of the recipient.

2.5.2.3 debugging arp proxy

Function

The **debugging arp proxy** command, you can enable proxy ARP debugging.

The **undo debugging arp proxy** command, you can disable proxy ARP debugging.

By default, the proxy ARP debugging function is disabled.

Format

debugging arp proxy [**intra-vlan** | **inter-vlan**] [**interface** *interface-type interface-number*]

undo debugging arp proxy [**intra-vlan** | **inter-vlan**] [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
intra-vlan	Enables proxy ARP debugging within a VLAN.	-
inter-vlan	Enables proxy ARP debugging between VLANs.	-
interface *interface-type interface-number*	Enables proxy ARP debugging on the specified interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

Using the **debugging arp proxy** command, you can enable proxy ARP debugging. The command will enable all kinds of proxy ARP debugging on all interfaces when no arguments specified.

Using the **undo debugging arp proxy** command, you can disable proxy ARP debugging. The command will disable all kinds of proxy ARP debugging on all interfaces when no arguments specified.

Precautions

To display debugging information on the terminal, you can run the **terminal monitor** and **terminal debugging** commands. For details, refer to the chapter "Information Center Configuration" in the *Configuration Guide* - *System Management*.

You can specify the interface and proxy ARP type in the command to reduce the number of the output of debugging information and improve information usability as well as the accuracy and efficiency of fault location.

Example

# Enable proxy ARP debugging between VLANs.

```
<HUAWEI>debugging arp proxy inter-vlan
```

# Enable proxy ARP debugging.

```
<HUAWEI>debugging arp proxy
```

2.5.3 DHCP Debugging Command

NOTE

The CE6810LI does not support commands that are relevant to DHCPv6 Relay.

2.5.3.1 debugging dhcp relay

Function

Using the **debugging dhcp relay** command, you can enable the debugging flag of DHCP relay module.

Using the **undo debugging dhcp relay** command, you can disable the debugging flag of DHCP relay module.

By default, the debugging of dhcp relay module is disabled.

Format

debugging dhcp relay { **all** | **error** | **event** | **info** | **packet** [**client mac** *mac-address*] }

undo debugging dhcp relay { **all** | **error** | **event** | **info** | **packet**}

Parameters

Parameter	Description	Value
all	Enables or disables debugging of all DHCP relay functions.	-
error	Indicates that the debugging includes DHCP internal error information only.	-
event	Indicates that the debugging includes event information only.	-
info	Indicates that the debugging includes info only.	-
packet	Indicates that the debugging includes packet only.	-
client mac *mac-address*	Indicates that the packet debugging includes client mac address.	The value is in the H-H-H format.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

When a DHCP relay module becomes faulty, the network administrator cannot perform local management using DHCP relay to start, modify, or delete configuration on the remote device. You can run this command to start the debugging information on the DHCP relay module and rapidly locate faults based on the obtained information.

Example

# Enable all DHCP relay module debugging.

```
<HUAWEI> debugging dhcp relay all
```

2.5.3.2 debugging dhcp server

Function

The **debugging dhcp server** command enables debugging of the DHCP server module.

The **undo debugging dhcp server** command disables debugging of the DHCP server module.

By default, the debugging of the DHCP server module is disabled.

Format

debugging dhcp server { **all** | **error** | **event** | **info** | **packet** | **timer** }

undo debugging dhcp server { **all** | **error** | **event** | **info** | **packet** | **timer** }

Parameters

Parameter	Description	Value
all	Enables or disables all debugging of the DHCP server module.	-
error	Enables or disables error debugging of the DHCP server module.	-
event	Enables or disables event debugging of the DHCP server module.	-
info	Enables or disables message debugging of the DHCP server module.	-
packet	Enables or disables packet debugging of the DHCP server module.	-
timer	Enables or disables timer debugging of the DHCP server module.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To locate a fault on the DHCP server module, run the **debugging dhcp server** command to view debugging information of the DHCP server.

Example

# Enable all debugging of the DHCP server.

```
<HUAWEI> debugging dhcp server all
```

2.5.3.3 debugging dhcpv6 relay

Function

The **debugging dhcpv6 relay** command enables the debugging flag of DHCPv6 Relay component.

The **undo debugging dhcpv6 relay** command disables the debugging flag of DHCPv6 Relay component.

By default, the debugging of DHCPv6 Relay component is disabled.

Format

debugging dhcpv6 relay { **all** | **packet** | **error** | **event** }

undo debugging dhcpv6 relay { **all** | **packet** | **error** | **event** }

Parameters

Parameter	Description	Value
all	Enables or disables debugging of all DHCPv6 Relay component functions.	-
packet	Indicates debugging of packet functions.	-
error	Indicates DHCP internal error information.	-
event	Indicates debugging of event functions.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

When a DHCPv6 Relay module becomes faulty, You can run this command to start the debugging of system message functions on the DHCPv6 Relay module and rapidly locate faults based on the obtained information.

Example

# Enable all DHCPv6 Relay component debugging.

```
<HUAWEI> debugging dhcpv6 relay all
```

2.5.3.4 debugging dhcpv6 server

Function

The **debugging dhcpv6 server** command enables the debugging of a DHCPv6 server.

The **undo debugging dhcpv6 server** command disables the debugging of a DHCPv6 server.

By default, the debugging of a DHCPv6 server is disabled.

Format

debugging dhcpv6 server { **all** | **error** | **event** | **info** | **packet** | **timer** }

undo debugging dhcpv6 server { **all** | **error** | **event** | **info** | **packet** | **timer** }

Parameters

Parameter	Description	Value
all	Enables or disables the debugging of all information.	-
error	Enables or disables the debugging of errors.	-
event	Enables or disables the debugging of events.	-
info	Enables or disables the debugging of messages.	-
packet	Enables or disables the debugging of packets.	-
timer	Enables or disables the debugging of timers.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

When a DHCPv6 server becomes faulty, you can run the **debugging dhcpv6 server** command to enable the debugging of the DHCPv6 server. The debugging information helps locate the fault.

Example

# Enable the debugging of all information.

```
<HUAWEI> debugging dhcpv6 server all
```

2.5.4 DNS Debugging Commands

2.5.4.1 debugging dns event

Function

The **debugging dns event** command enables the debugging of messages sent and received by the DNS module.

The **undo debugging dns event** command disables the debugging of messages sent and received by the DNS module.

By default, the debugging of messages sent and received by the DNS module is disabled.

Format

debugging dns event

undo debugging dns event

Parameters

None

Views

User views

Default Level

3: Management level

Usage Guidelines

Usage Scenario

If you want to view information about messages sent and received by the DNS module, run the **debugging dns event** command.

Example

# Display information about messages sent and received by the DNS module.

```
<HUAWEI> debugging dns event
May 25 2012 07:24:38.621 HUAWEI %%01DNS/7/dns_ipv4_event(d):CID=0x807503ff;DNS event :Smp
Query TransNo: 0, TestFlag:0,  HostName:google
May 25 2012 07:24:38.621 HUAWEI %%01DNS/7/dns_ipv4_event(d):CID=0x807503ff;DNS
event :Host:google, Query Server:1.1.1.1, Domain:huawei
```

2.5.4.2 debugging dns ipv4 packet

Function

The **debugging dns ipv4 packet** command enables the debugging of IPv4 packets sent and received by the DNS module.

The **undo debugging dns ipv4 packet** command disables the debugging of IPv4 packets sent and received by the DNS module.

By default, the debugging of packets sent and received by the DNS module is disabled.

Format

debugging dns ipv4 packet

undo debugging dns ipv4 packet

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

If you want to view information about IPv4 packets sent or received by the DNS module, run the **debugging dns ipv4 packet** command to enable the debugging function.

Example

# Display information about IPv4 packets sent or received by the DNS module.

```
<HUAWEI> debugging dns ipv4 packet
May 25 2012 07:26:15.761 HUAWEI %%01DNS/7/dns_ipv4_packet(d):CID=0x807503ff;DNS PKT :PKT Type:
DNS Send pkt; PKT: 05000100 00010000 00000000 06676F6F 676C6506 68756177 65690000 010001.
```

2.5.4.3 debugging dns ipv6 packet

Function

The **debugging dns ipv6 packet** command enables debugging of IPv6 packets on the DNS module.

The **undo debugging dns ipv6 packet** command disables debugging of IPv6 packets on the DNS module.

By default, debugging of IPv6 packets on the DNS module is disabled.

Format

debugging dns ipv6 packet

undo debugging dns ipv6 packet

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

To check information about IPv6 packets on the DNS module, run the **debugging dns ipv6 packet** command to enable debugging of the IPv6 packets.

Example

# Enable debugging of IPv6 packets on the DNS module.

```
<HUAWEI> debugging dns ipv6 packet
Aug  1 2014 16:11:44.028 HUAWEI %%01DNS/7/dns_ipv6_packet(d):CID=0x80750490;DNS IPv6 PKT: PKT
Type(Send IPv6 pkt); PKT Content: 1A000100 00010000 00000000 07687561 77656933 03636F6D
00001C00 01.

Aug  1 2014 16:11:44.036 HUAWEI %%01DNS/7/dns_ipv6_packet(d):CID=0x80750490;DNS IPv6 PKT: PKT
Type(Receive IPv6 pkt); PKT Content: 1A008580 00010001 00000000 07687561 77656933 03636F6D
00001C00 01C00C00 1C000100 01518000 10201400 00000000 00000000 00000000 01.
```

Table 2-14 Description of the **debugging dns ipv6 packet** command output

Item	Description
PKT Type	DNS packet type
PKT Content	DNS packet content

2.5.5 IPv6 Debugging Commands

2.5.5.1 debugging rawip ipv6

Function

The **debugging rawip ipv6** command enables debugging of IPv6 RAWIP packets.

The **undo debugging rawip ipv6** command disables debugging of IPv6 RAWIP packets.

By default, the debugging of IPv6 RAWIP packets is disabled.

Format

debugging rawip ipv6 packet [**src-ip** *ipv6-address*] [**dest-ip** *ipv6-address*] [**protocol** *protocol-number*] [**verbose** *packet-length*] [**socket-id** *socket-id*]

undo debugging rawip ipv6 packet [**src-ip** *ipv6-address*] [**dest-ip** *ipv6-address*] [**protocol** *protocol-number*] [**verbose** *packet-length*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Outputs a packet.	-
src-ip *ipv6-address*	Specifies the source address so that packets with the same source address are filtered.	The value is a 32-digit hexadecimal number, in the X:X:X:X:X:X:X:X format.
dest-ip *ipv6-address*	Specifies the destination address so that packets with the same destination address are filtered.	The value is a 32-digit hexadecimal number, in the X:X:X:X:X:X:X:X format.
protocol *protocol-number*	Specifies the protocol number so that packets with the same protocol number are filtered.	The value is an integer ranging from 0 to 255.
verbose	Displays the detailed information about a packet.	-
packet-length	Displays the length of the detailed information about a packet.	The value is an integer ranging from 0 to 64.
socket-id *socket-id*	Specifies the Socket ID.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging rawip ipv6** command enables debugging of IPv6 RAWIP packets sent and received by the host. For example, to check sending and receiving of IPv6 ping packets or diagnose interworking between OSPFv3 connections, run the **debugging rawip ipv6** command.

Example

# Enable debugging of IPv6 RAWIP packets.

```
<HUAWEI> debugging rawip ipv6 packet
<HUAWEI> terminal debugging
```

# Detect that the host sends an IPv6 RAWIP packet.

```
Aug  4 2011 11:55:30 HUAWEI %%01SOCKET/7/debug_rawip_packet(d):CID=0x80652749;
870: Output: pid = 6629141, socketid = 36, protocol = 58, ifindex = 2,src = ::1, dst = ::1, datelen = 64
```

# Detect that the host receives an IPv6 RAWIP packets.

```
Aug  4 2011 11:55:30 HUAWEI %%01SOCKET/7/debug_rawip_packet(d):CID=0x80652749;
870: Input: pid = 6629141, socketid = 36, protocol = 58, ifindex = 2,src = ::1, dst = ::1, datelen = 64
```

2.5.5.2 debugging tcp ipv6

Function

The **debugging tcp ipv6** command enables debugging of IPv6 TCP packets and outputs debugging information.

The **undo debugging tcp ipv6** command disables debugging of IPv6 TCP packets.

By default, the debugging of IPv6 TCP packets is disabled.

Format

debugging tcp ipv6 packet [**src-ip** *ipv6-address*] [**src-port** *port-number*] [**dest-ip** *ipv6-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

undo debugging tcp ipv6 packet [**src-ip** *ipv6-address*] [**src-port** *port-number*] [**dest-ip** *ipv6-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Debugs a packet.	-
src-ip *ipv6-address*	Displays packets of the specified local IPv6 address.	The value is a 32-digit hexadecimal number, in the X:X:X:X:X:X:X:X format.
src-port *port-number*	Displays packets of the specified local TCP port number.	The value is an integer ranging from 0 to 65535.
dest-ip *ipv6-address*	Displays packets of the specified destination IPv6 address.	The value is a 32-digit hexadecimal number, in the X:X:X:X:X:X:X:X format.

Parameter	Description	Value
dest-port *port-number*	Displays packets of the specified destination TCP port number.	The value is an integer ranging from 0 to 65535.
socket-id *socket-id*	Displays the data length of a TCP packet.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging tcp ipv6** command enables debugging of IPv6 TCP packets sent and received by the host and outputs debugging information. For example, to detect interworking between BGP 4+ connections, run the **debugging tcp ipv6** command.

Example

# Enable debugging of TCP IPv6 packets.

```
<HUAWEI> debugging tcp ipv6 packet
<HUAWEI> terminal debugging
```

# Detect that the host sends a TCP IPv6 packet.

```
2037-11-24 22:10:15 RT5 %%01SOCKET/7/debug_tcp_packe(d):CID=2154103793;
TCP6 debug packet information:
420: Output: pid = 6620139, socketid = 7943,
(src = ::1 : 23, dst = ::1 : 23973,
seq = 2523083814, ack = 3434375926, datalen = 29, optlen = 20 ,
flag = ACK PUSH , window = 65526, ttl = 64, tos = 0)
```

# Detect that the host receives a TCP IPv6 packet.

```
2037-11-24 22:10:15 RT5 %%01SOCKET/7/debug_tcp_packe(d):CID=2154103793;
TCP6 debug packet information:
230: Input: pid = 6620139, socketid = 6794,
(src = ::1 : 23973, dst = ::1 : 23,
seq = 3434375917, ack = 2523083814, datalen = 29, optlen = 20 ,
flag = ACK PUSH , window = 65535, ttl = 64, tos = 0)
```

2.5.5.3 debugging udp ipv6

Function

The **debugging udp ipv6** command enables debugging of UDP IPv6 packets and outputs debugging information.

The **undo debugging udp ipv6** command disables debugging of UDP IPv6 packets.

By default, the debugging of UDP IPv6 packets is disabled.

Format

debugging udp ipv6 packet [**src-ip** *ipv6-address*] [**src-port** *port-number*] [**dest-ip** *ipv6-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

undo debugging udp ipv6 packet [**src-ip** *ipv6-address*] [**src-port** *port-number*] [**dest-ip** *ipv6-address*] [**dest-port** *port-number*] [**socket-id** *socket-id*]

Parameters

Parameter	Description	Value
packet	Debugs a packet.	-
src-ip *ipv6-address*	Displays packets of the specified local IPv6 address.	The value is a 32-digit hexadecimal number, in the X:X:X:X:X:X:X:X format.
src-port *port-number*	Displays packets of the specified local UDP IPv6 port number.	The value is an integer ranging from 0 to 65535.
dest-ip *ipv6-address*	Displays packets of the specified destination IPv6 address.	The value is a 32-digit hexadecimal number, in the X:X:X:X:X:X:X:X format.
dest-port *port-number*	Displays packets of the specified destination UDP IPv6 port number.	The value is an integer ranging from 0 to 65535.
socket-id *socket-id*	Displays the data length of a UDP packet.	The value is an integer ranging from 0 to 2147418111.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging udp ipv6** command enables debugging of IPv6 UDP packets sent and received by the host and outputs debugging information. For example, to detect interworking between BGP 4+ connections, run the **debugging udp ipv6** command.

Example

# Enable debugging of UDP IPv6 packets.

```
<HUAWEI> debugging udp ipv6 packet
<HUAWEI> terminal debugging
```

# Detect that the host sends a UDP IPv6 packet.

```
Aug  4 2011 12:14:18 HUAWEI %%01SOCKET/7/debug_udp_packet(d):CID=0x80652793:
UDP6 debug packet information:
380: Output: pid = 6629209,socketid = 17, ifindex = 2,
(Time = 1337310519,src = ::1 : 300, dst = ::1 : 300,
datalen = 208)
```

# Detect that the host receives a UDP IPv6 packet.

```
Aug  4 2011 12:14:18 HUAWEI %%01SOCKET/7/debug_udp_packet(d):CID=0x80652793:
UDP6 debug packet information:
380: Input: pid = 6629209,socketid = 17, ifindex = 2,
(Time = 1337310520,src = ::1 : 300, dst = ::1 : 300,
datalen = 208)
```

2.5.5.4 debugging ipv6 nd

Function

The **debugging ipv6 nd** command debugs IPv6 packets and the ND state machine, and displays debugging information.

The **undo debugging ipv6 nd** command disables debugging of IPv6 packets and the ND state machine.

By default, the debugging of IPv6 packets and the ND state machine is disabled.

Format

debugging ipv6 nd [**source** *src-ip*] [**destination** *dst-ip*] [**interface** *interface-type interface-number*] [**verbose** [*verbose-length*]] [**number** *print-number*]

undo debugging ipv6 nd

Parameters

Parameter	Description	Value
source *src-ip*	Specifies the source IPv6 address.	The value is a 128-bit hexadecimal number, in the X:X:X:X:X:X:X:X format.
destination *dst-ip*	Specifies the destination IPv6 address.	The value is a 128-bit hexadecimal number, in the X:X:X:X:X:X:X:X format.
interface *interface-type*	Specifies the interface type.	-

Parameter	Description	Value
interface-number	Specifies the interface number.	-
verbose *verbose-length*	Specifies the length of the detailed packet information.	The value is an integer ranging from 1 to 64, bytes.By default, the value is 64 bytes.
number *print-number*	Specifies the number of the packet information.	The value is an integer ranging from 1 to 4294967295, bytes. By default, the value is **10**.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging ipv6 nd** command displays IPv6 ND packets and the information about the ND state machine for locating system exceptions.

Precautions

- When the packet length exceeds the length specified by *verbose-length*, the information about the length specified by *verbose-length* is printed.
- When the packet is shorter than *verbose-length*, the information about the actual length is printed.

Example

# Enable debugging on IPv6 ND packets and the information about the ND state machine.

```
<HUAWEI> debugging ipv6 nd interface 10GE 1/0/1
<HUAWEI> terminal debugging
Info: Current terminal debugging is on.
<HUAWEI> ping ipv6 -c 1 2001:db8::2
PING 2001:db8::2 : 56  data bytes, press CTRL_C to break
Mar 30 2012 08:37:20.623 HUAWEI %%01ND/7/packet(d):CID=0x80730411;On The Interface 10GE1/0/1,
Received NDMISS: 2001:db8::2

Mar 30 2012 08:37:20.623 HUAWEI %%01ND/7/packet(d):CID=0x80730411;On The Interface 10GE1/0/1,
Adding NB Entry: 2001:db8::2  NB State : INCOMPLETE

Mar 30 2012 08:37:20.623 HUAWEI %%01ND/7/packet(d):CID=0x80730411;On The Interface 10GE1/0/1,
Sending NS to 2001:db8::FF00:2, IP6(Version = 6, TrafficClass = 192, FlowLabel = 0, PayloadLength = 32,
HopLimit = 255, NextHeader = 58, Src = 2001:db8::1, Dst = 2001:db8::FF00:2), ICMP6(Type = 135(NS), Code
= 0, Checksum = 0x8E98, Reserved = 0, TargetAddr = 2001:db8::2, Type = 1, Length = 1, SrcLLAddr = 36CD-
B111-0303)
```

```
Mar 30 2012 08:37:20.713 HUAWEI %%01ND/7/packet(d):CID=0x80730411;On The Interface 10GE1/0/1, 
Sending NS to 2001:db8::FF00:2, IP6(Version = 6, TrafficClass = 192, FlowLabel = 0, PayloadLength = 32, 
HopLimit = 255, NextHeader = 58, Src = 2001:db8::1, Dst = 2001:db8::FF00:2), ICMP6(Type = 135(NS), Code 
= 0, Checksum = 0x8E98, Reserved = 0, TargetAddr = 2001:db8::2, Type = 1, Length = 1, SrcLLAddr = 36CD-
B111-0303)

    Request time out

  ---2001:db8::2 ping statistics---
    1 packet(s) transmitted
    0 packet(s) received
    100.00% packet loss
    round-trip min/avg/max=0/0/0 ms
```

Table 2-15 Description of the **debugging ipv6 nd** command output

Item	Description
CID	CID of the component to which the output information belongs
Received NDMISS	Received NDMISS event
Adding NB Entry	Adding NB entry
Sending NS to	Sending NS packets to the peer
Received NA from	Receiving NA packets from the peer
NB State	NB entry status: ● INCOMPLETE: false entry ● REACHABLE: reachable entry ● STALE: out-of-date entry ● DELAY: delayed entry ● PROBE: probe entry
Received NS from	Receiving NS packets from the peer
Sending NA to	Sending NA packets to the peer
Version	Version number
TrafficClass	Type of communication traffic
FlowLabel	Flow label
PayloadLength	Valid payload length
HopLimit	Hop limit
NextHeader	Next-hop packet header
Src	Source IPv6 address
Dst	Destination IPv6 address
ICMP6	ICMP6 protocol

Type	ICMP packet type
Code	ICMP packet code
Checksum	ICMP packet checksum
Reserved	Reserved word in the ICMP packet
TargetAddr	Destination IP address in the ICMP packet
Length	ICMP packet length
SrcLLAddr	Source link-local IPv6 address

2.5.5.5 debugging ipv6 packet

Function

The **debugging ipv6 packet** command debugs IPv6 packets and displays debugging information.

The **undo debugging ipv6 packet** command disables debugging of IPv6 packets.

By default, the debugging of IPv6 packets is disabled.

Format

debugging ipv6 packet [**error**] [**min-length** *min-length*] [**max-length** *max-length*] [**source** *src-ip*] [**destination** *dst-ip*] [**interface** *interface-type interface-number*] [**verbose** [*verbose-length*]] [**number** *print-number*]

undo debugging ipv6 packet [**error**]

Parameters

Parameter	Description	Value
error	Displays the specified IPv6 error packets.	-
min-length *min-length*	Specifies the minimum packet length.	The value is an integer ranging from 1 to 65535.
max-length *max-length*	Specifies the maximum packet length.	The value is an integer ranging from 1 to 65535.
source *src-ip*	Specifies the source IPv6 address.	The address is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.

Parameter	Description	Value
destination *dst-ip*	Specifies the destination IPv6 address.	The address is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
interface *interface-type*	Specifies the interface type.	-
interface-number	Specifies the interface number.	-
verbose *verbose-length*	Specifies the length of the detailed packet information.	The value is an integer ranging from 1 to 64, bytes. The default value is **64**.
number *print-number*	Specifies the number of the packet information.	The value is an integer ranging from 1 to 4294967295, bytes. By default, the value is **10**.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging ipv6 packet** command displays IPv6 packets for locating system exceptions.

Example

```
<HUAWEI> debugging ipv6 packet
2011-04-18 17:40:47 HUAWEI %%01PP6/7/packet(d):CID=2154964763;Receiving packet. (IfIndex = 5, Version = 6,
TrafficClass = 0, FlowLabel = 0, PayloadLength = 64, HopLimit = 64, Src = 12::1, Dst = 12::1, NextHeader = ICMPv6,
Type = 128, Code = 0, Chechsum = 0x795F, EchoRequest, Identifier = 1280, SequenceNumber = 256)


2011-04-18 17:40:47 HUAWEI %%01PP6/7/packet(d):CID=2154964763;Sending packet. (IfIndex = 5, Version = 6,
TrafficClass = 0, FlowLabel = 0, PayloadLength = 64, HopLimit = 64, Src = 12::1, Dst = 12::1, NextHeader = ICMPv6,
Type = 129, Code = 0, Chechsum = 0x785F, EchoReply, Identifier = 1280, SequenceNumber = 256)


<HUAWEI> debugging ipv6 packet verbose 50
2011-04-18 17:41:11 HUAWEI %%01PP6/7/packet(d):CID=2154964763;Receiving packet. (IfIndex = 5, Version = 6,
TrafficClass = 0, FlowLabel = 0, PayloadLength = 64, HopLimit = 64, Src = 12::1, Dst = 12::1, NextHeader = ICMPv6,
Type = 128, Code = 0, Chechsum = 0x793F, EchoRequest, Identifier = 1312, SequenceNumber = 256)
Memory (IPv6 Pkt):
        6000 0000 0040 3a40 0012 0000 0000 0000
        0000 0000 0000 0001 0012 0000 0000 0000
        0000 0000 0000 0001 8000 793f 0520 0100
        0000


2011-04-18 17:41:11 HUAWEI %%01PP6/7/packet(d):CID=2154964763;Sending packet. (IfIndex = 5, Version
= 6,
TrafficClass = 0, FlowLabel = 0, PayloadLength = 64, HopLimit = 64, Src = 12::1, Dst = 12::1, NextHeader =
ICMPv6,
Type = 129, Code = 0, Chechsum = 0x783F, EchoReply, Identifier = 1312, SequenceNumber = 256)
Memory (IPv6 Pkt):
        6000 0000 0040 3a40 0012 0000 0000 0000
        0000 0000 0000 0001 0012 0000 0000 0000
        0000 0000 0000 0001 8100 783f 0520 0100
        0000
```

Table 2-16 Description of the **debugging ipv6 packet** command output

Item	Description
CID	CID of the component that outputs the current information
Receiving packet	Receiving packet
Sending packet	Sending packet
Discarding packet	The packet is Discarding
IfIndex	Index of the interface over which packets are transmitted
Version	Version number
TrafficClass	Traffic class
FlowLabel	Flow label
PayloadLength	Payload length
HopLimit	Hop limit
Src	Source address
Dst	Destination address
NextHeader	Next packet header
Type	Types of ICMP packets
Code	Code of ICMP packets
Checksum	Checksum of ICMP packets
EchoRequest	EchoRequest packet
EchoReply	EchoReply packet

Identifier	The Identifier of ICMP packet
SequenceNumber	The sequence number of ICMP packet
Memory (IPv6 Pkt)	IPv6 packet displayed in hexadecimal notation

2.5.5.6 debugging ipv6 tunnel packet

Function

The **debugging ipv6 tunnel packet** command enables the debugging of ingress and egress 6over4 tunnels, and displays debugging information.

The **undo debugging ipv6 tunnel packet** command disable the debuging of ingress and egress 6over4 tunnels, and displays debugging information.

By default, the debugging of ingress and egress 6over4 tunnels is disabled.

NOTE

The CE5880EI, CE6863, CE6863K, CE6881E, CE6820, CE6881, CE6881K, and CE6880EI do not support this command.

Format

debugging ipv6 tunnel packet [**interface** *interface-type interface-number*] [**verbose** [*verbose-length*]] [**number** *print-number*]

undo debugging ipv6 tunnel packet

Parameters

Parameter	Description	Value
interface *interface-type*	Specifies the interface type.	-
interface-number	Specifies the interface number.	-
verbose *verbose-length*	Specifies the length of the detailed packet information.	The value is an integer ranging from 1 to 64, bytes. The default value is **64**.
number *print-number*	Specifies the number of the packet information.	The value is an integer ranging from 1 to 4294967295, bytes. By default, the value is **10**.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging ipv6 tunnel packet** command displays the information about ingress and egress 6over4 tunnels for locating system exceptions.

Example

```
<HUAWEI> debugging ipv6 tunnel packet interface Tunnel 10
```

2.6 IP Unicast Routing Debugging Commands

2.6.1 OSPF Debugging Commands

2.6.1.1 debugging ospf import

Function

The **debugging ospf import** command enables debugging for the routes imported by an OSPF process.

The **undo debugging ospf import** command disables debugging for the routes imported by an OSPF process.

By default, debugging is disabled for the routes imported by an OSPF process.

Format

debugging ospf [*process-id*] **import** [**filter ip-prefix** *ip-prefix-name*]

undo debugging ospf [*process-id*] **import** [**filter ip-prefix** *ip-prefix-name*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPF process.	The value is an integer ranging from 1 to 4294967295.
filter	Indicates a policy to be used to filter debugging information.	-

Parameter	Description	Value
ip-prefix *ip-prefix-name*	Specifies the name of an IPv4 prefix list to be used to filter debugging information.	The name is a string of 1 to 169 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string.

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable debugging for the routes imported by an OSPF process so that debugging information can be displayed to help troubleshooting, run the **debugging ospf import** command.

Example

# Enable debugging for the routes imported by all OSPF processes.

```
<HUAWEI> debugging ospf import
Sep  4 2017 03:28:19.379 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=Admin-VS-CID=0x808400ae;
FileID: 0x2a Line: 3115 Level: 0x5 OSPF 1 Add new route [6.6.6.6/32] into routing table.
Sep  4 2017 03:28:19.379 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=Admin-VS-CID=0x808400ae;
FileID: 0x2a Line: 2697 Level: 0x5 OSPF 1 Create new route source success.
Sep  4 2017 03:28:19.379 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=Admin-VS-CID=0x808400ae;
FileID: 0x2a Line: 2751 Level: 0x5 OSPF 1 Imported one new route.<Sum:1>
Sep  4 2017 03:28:19.381 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=Admin-VS-CID=0x808400ae;
FileID: 0x2e Line: 1574 Level: 0x6 OSPF 1 [6.6.6.6/32] Update LSA.
```

Table 2-17 Description of the **debugging ospf import** command output

Item	Description
CID	Component ID
FileID	File ID
Line	Line ID
Level	Level

2.6.1.2 debugging ospf packet

Function

The **debugging ospf packet** command enables debugging of all types of sent and received OSPF packets.

The **undo debugging ospf packet** command disables debugging of sent and received OSPF packets.

Format

debugging ospf [*process-id*] **packet** [**ack** | **dd** | **hello** | **request** | **update**] [**interface** *interface-type interface-number*] [**brief**] [**filter** { { **src** | **nbr** } { *ip-address* | **acl** *acl-number* } }]

debugging ospf packet { **rcv-dump** [**error**] | **snd-dump** } [**interface** *interface-type interface-number*]

debugging ospf [*process-id*] **packet grace**

undo debugging ospf [*process-id*] **packet** [**ack** | **dd** | **hello** | **request** | **update**] [**interface** *interface-type interface-number*] [**brief**] [**filter** { { **src** | **nbr** } { *ip-address* | **acl** *acl-number* } }]

undo debugging ospf packet { **rcv-dump** [**error**] | **snd-dump** } [**interface** *interface-type interface-number*]

undo debugging ospf [*process-id*] **packet grace**

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPF process.	The value is an integer ranging from 1 to 4294967295.
ack	Displays debugging information about type-5 OSPF packets.	-
dd	Displays debugging information about type-2 OSPF packets.	-
hello	Displays debugging information about type-1 OSPF packets.	-
request	Displays debugging information about type-3 OSPF packets.	-
update	Displays debugging information about type-4 OSPF packets.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
brief	Displays the brief OSPF information.	-

Parameter	Description	Value
filter	Set filter policy.	-
src	Set filter policy of Self router.	-
nbr	Set filter policy of Neighbor router.	-
ip-address	Specifies the IP address.	-
acl *acl-number*	Specifies the number of a basic ACL.	The value is an integer ranging from 2000 to 2999.
rcv-dump	Displays debugging information about received connection packets. All received connection packets are output.	-
error	Displays debugging information about data-link bad packet receiving.	-
snd-dump	Displays debugging information about sent connection packets. All sent connection packets are output.	-
grace	Displays debugging information about Grace LSA in update packet.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospf	debug

Usage Guidelines

The **debugging ospf packet** command enables debugging of all types of sent and received OSPF packets. Generally, this command can be used to check the creation of neighbors and trace problems in the process. If a problem occurs, enable debugging of the related type of packet.

If the OSPF process ID is not specified, the packets of all OSPF processes are displayed.

OSPF packets are ack, dd, hello, request, and update packets. This command displays the information about all types of packets.

Debugging affects the system performance. After debugging is complete, run the **undo debugging ospf packet** command in time to disable debugging of OSPF packets.

Example

# Enable debugging of OSPF hello packets.

```
<HUAWEI> debugging ospf packet hello
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VR=0-CID=2156013447;
FileID: 0x13 Line: 832 Level: 0x5
  OSPFv2 1 SEND Packet, Interface: Eth3/0/3
  Source Address: 172.16.1.2
  Destination Address: 224.0.0.5
  Ver# 2, Type: 1 (Hello)
  Length: 44, Router: 172.16.1.2
  Area: 0.0.0.1, Chksum: d375
  AuType: 1
  Key(ascii): 61 62 63 0 0 0 0 0
  Net Mask: 255.255.255.0
  Hello Int: 10, Option: _E_
  Rtr Priority: 1, Dead Int: 40
  DR: 172.16.1.2
  BDR: 0.0.0.0
  # Attached Neighbors: 0
```

Table 2-18 Description of the **debugging ospf packet** command output

Item	Description
Interface	Interface index
Source Address	Source address of packets
Destination Address	Destination address of packets
Ver	OSPF version
Type	OSPF packet type ● 1: hello packet ● 2: database description packet ● 3: connection status request packet ● 4: connection status update packet ● 5: connection status acknowledgment packet
Length	Length of an OSPF protocol packet
Router	ID of the source device
Area	Area ID of a packet
Chksum	Standard IP checksum of packet content
AuType	Authentication type used for a packet
Net Mask	Mask of the related interface
Hello Int	Interval of sending hello packets

Item	Description
Option	Option of the source device
Rtr Priority	Priority of the source device
Dead Int	Interval of neighbor disconnection
DR	Specified router in the network segment of the interface
BDR	Specified backup router in the network segment of the interface

2.6.1.3 debugging ospf route-calc

Function

The **debugging ospf route-calc** command enables debugging of route calculation of all OSPF processes.

The **undo debugging ospf route-calc** command disables debugging of route calculation of all OSPF processes.

Format

debugging ospf *process-id* **route-calc** { **ase** | **intra-area** | **inter-area** | **nssa** } **filter** { *address-ipv4 mask-ipv4* | **acl** { *acl-number* | *acl-name* } }

debugging ospf [*process-id*] **route-calc** { **all** | **asbr** | **ase** | **intra-area** | **inter-area** | **nssa** }

debugging ospf *process-id* **route-calc asbr filter** { *address-ipv4* | **acl** { *acl-number* | *acl-name* } }

undo debugging ospf *process-id* **route-calc** { **ase** | **intra-area** | **inter-area** | **nssa** } **filter** { *address-ipv4 mask-ipv4* | **acl** { *acl-number* | *acl-name* } }

undo debugging ospf [*process-id*] **route-calc** { **all** | **asbr** | **ase** | **intra-area** | **inter-area** | **nssa** }

undo debugging ospf *process-id* **route-calc asbr filter** { *address-ipv4* | **acl** { *acl-number* | *acl-name* } }

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPF process.	The value is an integer ranging from 1 to 4294967295.
acl *acl-number*	Specifies the number of a basic ACL.	The value is an integer ranging from 2000 to 2999.

Parameter	Description	Value
acl *acl-name*	Specifies the name of a named ACL.	The value is a string of 1 to 32 case-sensitive characters except spaces. The value must start with a letter or digit, and cannot contain only digits.
all	Displays the debugging information about all types of OSPF route calculation, including class-4 LSA, class-5 LSA, intra-area LSA, and inter-area LSA.	-
asbr	Displays debugging information about route calculation of class-4 LSA.	-
ase	Displays debugging information about route calculation of class-5 LSA.	-
intra-area	Displays debugging information about route calculation of LSAs in the area.	-
inter-area	Displays debugging information about route calculation of LSAs between areas.	-
filter	Indicates the filtering policy.	-
address-ipv4	Indicates Destination of IP address.	-
mask-ipv4	Indicates length of IP address mask.	-
nssa	Displays debugging information about NSSA route.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospf	debug

Usage Guidelines

The **debugging ospf route-calc** command helps locate faults.

Example

# Enable debugging of route calculation of class-5 LSAs in OSPF processes.

```
<HUAWEI> debugging ospf route-calc ase
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80842783;
FileID: 0x4c Line: 531 Level: 0x5 OSPF 1
  Route 10.1.1.1/32 is Updated (All Attributes Changed)  Type: EXTERNAL, Prio: 3, UpdateNum : 3109490,
LockInfo: 0x04, LSAInfo: 0x00, Flags: 0x61008063 Cost: 1, Cost Type: 2, Area: 0.0.0.0, Transit Area: 0.0.0.0,
Tag: 1, Nexthops: 1, IID: 2332033081, Base Count: 1, Direct Count: 0

2011-07-27 04:15:28 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80842783;
FileID: 0x19 Line: 729 Level: 0x5 OSPF 1
  External Route 10.1.1.1/32 IID 2332033081 added/Updated to RM

2011-07-27 04:15:28 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80842783;
FileID: 0x4b Line: 2586 Level: 0x5 OSPF 1
  Best path calculation end for Route 10.1.1.1/32
```

Table 2-19 Description of the **debugging ospf route-calc** command output

Item	Description
VS	Virtual system
CID	Component ID
FileID	File ID
Line	Line number
Level	Level
OSPF	Process ID
Route	Route
Type	Route type
Prio	Priority
UpdateNum	Number of update times
LockInfo	Lock information
LSAInfo	LSA information
Flags	Flag bit
Cost	Cost
Type	● 1: class-1 cost of LSAs outside the autonomous system ● 2: class-2 cost of LSAs outside the autonomous system
Area	Area number
Transit Area	Transmission area
Tag	Label

Item	Description
Nexthops	Next hop
IID	ecmp-group ID
Base Count	Number of routes in the base table
Direct Count	Number of routes in the direct-connect table

2.6.1.4 debugging ospf spf

Function

The **debugging ospf spf** command enables debugging of the shortest path tree calculation and next hops of all OSPF processes.

The **undo debugging ospf spf** command disables debugging of the shortest path tree calculation and next hops of all OSPF processes.

Format

debugging ospf [*process-id*] **spf**

undo debugging ospf [*process-id*] **spf**

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPF process.	The value is an integer ranging from 1 to 4294967295.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospf	debug

Usage Guidelines

After the *process-id* parameter is specified, the **debugging ospf spf** command enables debugging of the shortest path tree calculation and next hops of the specified OSPF process.

Example

# Enable debugging of the shortest path tree calculation and next hops of all OSPF processes.

```
<HUAWEI> debugging ospf spf
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x22 Line:
811 Level: 0x5 OSPF 1  Area: 1 MT: 0 basemt SPF calculation started.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1 Line: 145
Level: 0x5 OSPF 1  Area: 1 MT: 0 ver_type: 1 ver_id: 0x1010101 Add root vertex to SPF tree
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1 Line: 686
Level: 0x5 OSPF 1  Area: 1 MT: 0 vertex_type: 1 vertex_id: 1.1.1.1 vertex_option: 0 vertex_teif_cnt: 0
spf_cost: 0 spf_clc_num: 4 te_cost_diff: 0 path_cost: 0 Add vertex to SPF.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1 Line: 745
Level: 0x5 OSPF 1  Area: 1 MT: 0 ver_type: 2 ver_id: 0x6060602  cost: 1 Add vertex to candidate list.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1 Line: 787
Level: 0x5 OSPF 1  Area: 1 MT: 0 vertex_type: 2 vertex_id: 6.6.6.2 vertex_option: 0 vertex_teif_cnt: 0
spf_cost: 1 spf_clc_num: 4 te_cost_diff: 0 path_cost: 16777215 Add vertex to SPF.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787; NON-TE Nexthop
if_index: 6  ip_address: 6.6.6.1  if_type: TRANSIT.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1 Line: 838
Level: 0x5 OSPF 1  Area: 1 MT: 0 ver_type: 1 ver_id: 0x1010102  cost: 1 Add vertex to candidate list.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1 Line: 686
Level: 0x5 OSPF 1  Area: 1 MT: 0 vertex_type: 1 vertex_id: 1.1.1.2 vertex_option: 0 vertex_teif_cnt: 0
spf_cost: 1 spf_clc_num: 4 te_cost_diff: 0 path_cost: 16777215 Add vertex to SPF.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787; NON-TE Nexthop
if_index: 6  ip_address: 6.6.6.2  if_type: TRANSIT.
Dec 24 2011 21:11:460 HUAWEI %%01OSPF/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x22 Line:
820 Level: 0x5 OSPF 1  Area: 1 MT: 0 basemt SPF calculation completed
```

Table 2-20 Description of the **debugging ospf spf** command output

Item	Description
ver_type	- 1: router node - 2: network node
ver_id	Node ID
vertex_option	Node option (the same as the option in LSA)
spf_cost	Cost of the calculated node
spf_clc_num	Number of calculation times of this node

2.6.1.5 debugging packet ospf

Function

Using the **debugging packet ospf** command, you can enable the debugs to show the packet processing within the system.

Using the **undo debugging packet ospf** command, you can disable the debugging of OSPF packets.

By default, the debugging of the process of transmitting OSPF packets is disabled.

Format

debugging packet ospf interface { *interface-type interface-number* } [**nsr**]

undo debugging packet ospf interface { *interface-type interface-number* } [**nsr**]

Parameters

Parameter	Description	Value
interface-type interface-number	Indicates the type and number of interface.	-
nsr	Indicates the Non-stop routing information.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospf	debug

Usage Guidelines

When ospf peer is formed, enabling of these debugs will show the complete packet flow in the system.

Example

# Enable OSPF packet debugging.

```
<HUAWEI> debugging packet ospf interface 10GE 1/0/13
SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x80782728
VS              : 0
Handle          : 3
TraceNum        : 2
Direction       : Up
Status          : 0
Time            : 2015-8-4 12:1:11 19
Data            :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x8082043e
VS              : 0
```

```
Handle          : 3
TraceNum        : 2
Direction       : Up
Status          : 0
Time            : 2015-8-4 12:1:11 19
Data            :
---------------------------------------------

OSPF:
---------------------------------------------
My Cid           : 0x8082043E
Peer Cid         : 0x8065042B
VS               : 0
Handle           : 3
TraceNum         : 2
Time             : 2015-8-4 12:1:11 19
Direction        : Up
Status           : 0
Version          : OSPF
Type             : Hello
Packet length    : 48
Router ID        : 10.1.4.4
Area ID          : 0.0.0.0
Checksum         : 0xC56F
---------------------------------------------

OSPF:
---------------------------------------------
My Cid           : 0x8082043E
Peer Cid         : 0x8065042B
VS               : 0
Handle           : 3
TraceNum         : 3
Time             : 2015-8-4 12:1:14 582
Direction        : Down
Status           : 0
Version          : OSPF
Type             : Hello
Packet length    : 48
Router ID        : 192.168.80.120
Area ID          : 0.0.0.0
Checksum         : 0xC56F
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x8082043e
VS              : 0
Handle          : 3
TraceNum        : 3
Direction       : Down
Status          : 0
Time            : 2015-8-4 12:1:14 582
Data            :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x80782728
VS              : 0
Handle          : 3
TraceNum        : 3
Direction       : Down
Status          : 0
Time            : 2015-8-4 12:1:14 582
Data            :
---------------------------------------------
```

```
LDM:
---------------------------------------------
My Cid          : 0x80782728
Peer Cid        : 0x650418
VS            : 0
Handle          : 3
TraceNum         : 2
Direction       : Up
Status         : 0
Interface index : 17
Link type       : ETH
Source mac       : e4 68 a3 56 0d d2
Dest mac        : 01 00 5e 00 00 05
Link protocol   : 0x0800
Protocol        : IPV4
Time           : 2015-8-4 12:1:11 18
Data           : 0x45C00044046A00000159C82F0A010202E0000005
---------------------------------------------

LDM:
---------------------------------------------
My Cid          : 0x80782728
Peer Cid        : 0x8065042B
VS            : 0
Handle          : 3
TraceNum         : 3
Direction       : Down
Status         : 0
Interface index : 17
Link type       : -
Protocol        : IPV4
Time           : 2015-8-4 12:1:14 579
Data           : 0x45C000440466000001159C8340A010201E0000005
---------------------------------------------
```

Table 2-21 Description of the **debugging packet ospf** command output

Item	Description
My Cid	Self Component Id
Peer Cid	Peer Component Id
VS	Virtual system
Handle	Handle value
TraceNum	Trace id
Direction	Direction of packet flow. It can be : ● Up ● Down
Status	Status of the packet processing
Data	Current date
Time	Current time
Version	OSPF version number
OSPF Type	OSPF packet type

Item	Description
Packet length	Length of packet
Router ID	Router-Id of originating device
Area ID	Area Id
Checksum	Checksum value of packet

2.6.1.6 display debugging ospf

Function

The **display debugging ospf** command displays information about current OSPF debugging functions.

Format

display debugging ospf

Parameters

None.

Views

All views

Default Level

1: Monitor level

Usage Guidelines

When a large amount of information is output, the **display debugging ospf** command can be used to view information about the enabled OSPF debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current OSPF debugging functions.

```
<HUAWEI> display debugging ospf
OSPF ROUTE-CALC debugging switch is on
```

2.6.2 OSPFv3 Debugging Commands

NOTE

CE6810LI does not support OSPFv3 debugging commands.

2.6.2.1 debugging ospfv3 import

Function

The **debugging ospfv3 import** command enables debugging for the routes imported by an OSPFv3 process.

The **undo debugging ospfv3 import** command disables debugging for the routes imported by an OSPFv3 process.

By default, debugging is disabled for the routes imported by an OSPFv3 process.

Format

debugging ospfv3 [*process-id*] **import** [**filter ipv6-prefix** *ipv6-prefix-name*]

undo debugging ospfv3 [*process-id*] **import** [**filter ipv6-prefix** *ipv6-prefix-name*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPFv3 process.	The value is an integer ranging from 1 to 4294967295.
filter	Indicates a policy to be used to filter debugging information.	-
ipv6-prefix *ipv6-prefix-name*	Specifies the name of an IPv6 prefix list to be used to filter debugging information.	The name is a string of 1 to 169 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string.

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable debugging for the routes imported by an OSPFv3 process so that debugging information can be displayed to help troubleshooting, run the **debugging ospfv3 import** command.

Example

# Enable debugging for the routes imported by all OSPFv3 processes.

```
<HUAWEI> debugging ospfv3 import
Sep  4 2017 03:22:32.872 HUAWEI %%01OSPFV3/6/OSPFV3_DEBUG(d):VS=Admin-VS-
CID=0x808400ae;FileID: 0x2a Line: 3115 Level: 0x5 OSPFv3 1 Add new route [5::5/128] into routing table.
Sep  4 2017 03:22:32.872 HUAWEI %%01OSPFV3/6/OSPFV3_DEBUG(d):VS=Admin-VS-
CID=0x808400ae;FileID: 0x2a Line: 2697 Level: 0x5 OSPFv3 1 Create new route source success.Sep  4 2017
03:22:32.872 HUAWEI %%01OSPFV3/6/OSPFV3_DEBUG(d):VS=Admin-VS-CID=0x808400ae;FileID: 0x2a
Line: 2751 Level: 0x5 OSPFv3 1 Imported one new route.<Sum:1>
Sep  4 2017 03:22:32.875 HUAWEI %%01OSPFV3/6/OSPFV3_DEBUG(d):VS=Admin-VS-
CID=0x808400ae;FileID: 0x2e Line: 1574 Level: 0x6 OSPFv3 1 [5::5/128] Update LSA.
```

Table 2-22 Description of the **debugging ospfv3 import** command output

Item	Description
CID	Component ID
FileID	File ID
Line	Line ID
Level	Level

2.6.2.2 debugging ospfv3 packet

Function

The **debugging ospfv3 packet** command enables debugging of all types of sent and received OSPFv3 packets.

The **undo debugging ospfv3 packet** command disables debugging of all types of sent and received OSPFv3 packets.

Format

debugging ospfv3 *process-id* **packet** { **ack** | **dd** | **hello** | **request** | **update** | **all** } [**received** | **sent**] [*interface-type interface-number* [*nbrrouter-id*] | **area** *area-id*] [**verbose**]

undo debugging ospfv3 *process-id* **packet** { **ack** | **dd** | **hello** | **request** | **update** | **all** } [**received** | **sent**] [*interface-type interface-number* [*nbrrouter-id*] | **area** *area-id*] [**verbose**]

debugging ospfv3 packet { **ack** | **dd** | **hello** | **request** | **update** | **all** } [**received** | **sent**] [**verbose**]

undo debugging ospfv3 packet { **ack** | **dd** | **hello** | **request** | **update** | **all** } [**received** | **sent**] [**verbose**]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPFv3 process.	The value is an integer ranging from 1 to 4294967295.

Parameter	Description	Value
ack	Displays debugging information about type-5 OSPFv3 packets.	-
dd	Displays debugging information about type-2 OSPFv3 packets.	-
hello	Displays debugging information about type-1 OSPFv3 packets.	-
request	Displays debugging information about type-3 OSPFv3 packets.	-
update	Displays debugging information about type-4 OSPFv3 packets.	-
all	Displays debugging information about all types of OSPFv3 packets.	-
received	Displays debugging information about received connection packets.	-
sent	Displays debugging information about sent connection packets.	-
interface-type	Specifies the type of an interface.	-
interface-number	Specifies the number of an interface.	-
nbrrouter-id	Specifies the neighbor ID.	The value is in dotted decimal notation.
area *area-id*	Specifies the area ID for the area.	The value can be an integer in decimal notation (ranging from 0 to 4294967295) or an IPv4 address.
verbose	Displays detailed debugging information.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospfv3	debug

Usage Guidelines

The **debugging ospfv3 packet** command enables debugging of all types of sent and received OSPFv3 packets. Generally, this command can be used to check the creation of neighbors and trace problems in the process. If a problem occurs, enable debugging of the related type of packet.

If the OSPFv3 process ID is not specified, the packets of all OSPFv3 processes are displayed.

OSPFv3 packets are ack, dd, hello, request, and update packets. This command displays the information about all types of packets.

Debugging affects the system performance. After debugging is complete, run the **undo debugging ospfv3 packet** command in time to disable debugging of OSPFv3 packets.

Example

# Enable debugging of OSPFv3 hello packets.

```
<HUAWEI> debugging ospfv3 packet hello verbose
2011-02-10 18:08:04 VRPV8 %%01ospfv2comm/6/OSPF_DEBUG(d):VS=0-CID=2156013401;
FileID: 0x13 Line: 850 Level: 0x5
  OSPF 1 SEND Packet, Interface: Eth3/0/5
  Source Address: FE80::36CE:71FF:FE10:305
  Destination Address: FF02::5
  Ver# 3, Type: 1 (Hello)
  Length: 40, Router: 12.13.22.1
  Area: 0.0.0.0, Chksum: 0
  InstanceID: 2
  Interface Id: 9, Rtr Priority: 1
  Options: V6:1 E:1 N:0 R:1 DC:0
  Hello Int: 10, Dead Int: 40
  DR: 10.13.22.1
  BDR: 10.13.22.3
  # Attached Neighbors: 1
  Neighbor: 10.13.22.3
```

Table 2-23 Description of the **debugging ospfv3 packet** command output

Item	Description
Source Address	Source address of packets
Destination Address	Destination address of packets
Ver	OSPFv3 version
Type	OSPFv3 packet type ● 1: hello packet ● 2: database description packet ● 3: connection status request packet ● 4: connection status update packet ● 5: connection status acknowledgment packet
Length	Length of an OSPFv3 protocol packet

Item	Description
Router	Source router ID
Area	Area ID of a packet
Chksum	Standard IP checksum of packet content
Instance Id	OSPFv3 instance ID
Interface Id	Interface index
Hello Int	Interval of sending hello packets
Option	Option of the source device
Rtr Priority	Priority of the source device
Dead Int	Interval of neighbor disconnection
DR	ID of the specified router
BDR	ID of the specified backup router

2.6.2.3 debugging ospfv3 route-calc

Function

The **debugging ospfv3 route-calc** command enables debugging of route calculation of all OSPFv3 processes.

The **undo debugging ospfv3 route-calc** command disables debugging of route calculation of all OSPFv3 processes.

By default, the debugging of route calculation of all OSPFv3 processes is disabled.

Format

debugging ospfv3 [*process-id*] **route-calc** { **all** | **asbr** | **ase** | **intra-area** | **inter-area** | **nssa** }

debugging ospfv3 *process-id* **route-calc** { **ase** | **intra-area** | **inter-area** | **nssa** } **filter** *address-ipv6 mask-ipv6*

debugging ospfv3 *process-id* **route-calc asbr filter** *address-ipv4*

undo debugging ospfv3 [*process-id*] **route-calc** { **all** | **asbr** | **ase** | **intra-area** | **inter-area** | **nssa** }

undo debugging ospfv3 *process-id* **route-calc** { **ase** | **intra-area** | **inter-area** | **nssa** } **filter** *address-ipv6 mask-ipv6*

undo debugging ospfv3 *process-id* **route-calc asbr filter** *address-ipv4*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPFv3 process.	The value is an integer ranging from 1 to 4294967295.
all	Displays the debugging information about all types of OSPFv3 route calculation.	-
asbr	Displays debugging information about route calculation of ASBR LSA.	-
ase	Displays debugging information about route calculation of ASE LSA.	-
intra-area	Displays debugging information about route calculation of LSAs in the area.	-
inter-area	Displays debugging information about route calculation of LSAs between areas.	-
nssa	Displays debugging information about route calculation of NSSA-LSA.	-
filter	Indicates the filtering policy.	-
address-ipv6	Indicates Destination of IP address.	The value is a 32-digit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
mask-ipv6	Indicates length of IP address mask.	It is an integer ranging from 1 to 128.
address-ipv4	Indicates Destination of ASBR IP address.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospfv3	debug

Usage Guidelines

The **debugging ospfv3 route-calc** command helps locate faults.

Example

# Enable debugging of route calculation of class-5 LSAs in OSPFv3 processes.

```
<HUAWEI> debugging ospfv3 route-calc ase
2011-07-27 04:56:51 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80842783;
FileID: 0x4c Line: 531 Level: 0x5 OSPFv3 1
  Route 20::/54 is Updated (All Attributes Changed)  Type: EXTERNAL, Prio: 4, UpdateNum : 3111973,
LockInfo: 0x04, LSAInfo: 0x00, Flags: 0x61008083 Cost: 1, Cost Type: 2, Area: 0.0.0.0, Transit Area: 0.0.0.0,
Tag: 1, Nexthops: 1, IID: 2332033082, Base Count: 1, Direct Count: 0

2011-07-27 04:56:51 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80842783;
FileID: 0x19 Line: 729 Level: 0x5 OSPFv3 1
  External Route 20::/54 IID 2332033082 added/Updated to RM

2011-07-27 04:56:51 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80842783;
FileID: 0x4b Line: 2586 Level: 0x5 OSPFv3 1
  Best path calculation end for Route 20::/54
```

Table 2-24 Description of the **debugging ospfv3 route-calc** command output

Item	Description
VS	Virtual system
CID	Component ID
FileID	File ID
Line	Line number
Level	Level
OSPFv3	Process ID
Route	Route
Type	Route type
Prio	Priority
UpdateNum	Update times
LockInfo	Lock information
LSAInfo	LSA information
Flags	Flag bit
Cost	Cost
Type	● 1: class-1 cost of LSAs outside the autonomous system ● 2: class-2 cost of LSAs outside the autonomous system
Area	Area number

Item	Description
Transit Area	Transmission area
Tag	Label
Nexthops	Next hop
IID	ecmp-group ID
Base Count	Number of routes in the base table
Direct Count	Number of routes in the direct-connect table

2.6.2.4 debugging ospfv3 spf

Function

The **debugging ospfv3 spf** command enables debugging of the shortest path tree calculation and next hops of all OSPFv3 processes.

The **undo debugging ospfv3 spf** command disables debugging of the shortest path tree calculation and next hops of all OSPFv3 processes.

By default, the debugging of the shortest path tree calculation and next hops of all OSPFv3 processes is disabled.

Format

debugging ospfv3 [*process-id*] **spf**

undo debugging ospfv3 [*process-id*] **spf**

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an OSPFv3 process.	The value is an integer ranging from 1 to 4294967295.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospfv3	debug

Usage Guidelines

After the *process-id* parameter is specified, the **debugging ospfv3 spf** command enables debugging of the shortest path tree calculation and next hops of the specified OSPFv3 process.

Example

# Enable debugging of the shortest path tree calculation and next hops of all OSPFv3 processes.

```
<HUAWEI> debugging ospfv3 spf
2011-07-27 03:18:30 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x22
Line: 811 Level: 0x5 OSPFv3 1  Area: 0 MT: 0 basemt SPF calculation started.
2011-07-27 03:18:30 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1
Line: 145 Level: 0x5 OSPFv3 1  Area: 0 MT: 0 ver_type: 1 ver_id: 0x0 Add root vertex to SPF tree
2011-07-27 03:18:30 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x1
Line: 686 Level: 0x5 OSPFv3 1  Area: 0 MT: 0 vertex_type: 1 vertex_id: 0.0.0.0 vertex_option: 275
vertex_teif_cnt: 0      spf_cost: 0 spf_clc_num: 1 te_cost_diff: 0 path_cost: 16777215 Add vertex to SPF.
2011-07-27 03:18:30 HUAWEI %%01OSPFV2COMM/6/OSPF_DEBUG(d):VS=0-CID=0x80852787;FileID: 0x22
Line: 820 Level: 0x5 OSPFv3 1  Area: 0 MT: 0 basemt SPF calculation completed
```

Table 2-25 Description of the **debugging ospfv3 spf** command output

Item	Description
ver_type	● 1: router node ● 2: network node
ver_id	Node ID
vertex_option	Node option (the same as the option in LSA)
spf_cost	Cost of the calculated node
spf_clc_num	Number of calculation times of this node

2.6.2.5 debugging packet ospfv3

Function

Using the **debugging packet ospfv3** command, you can enable the debugs to show the packet processing within the system.

Using the **undo debugging packet ospfv3** command, you can disable the debugging of OSPFv3 packets.

By default, the debugging of the process of transmitting OSPFv3 packets is disabled.

Format

debugging packet ospfv3 interface *interface-type interface-number* [**nsr**]

undo debugging packet ospfv3 interface *interface-type interface-number* [**nsr**]

Parameters

Parameter	Description	Value
interface-type interface-number	Indicates the type and number of interface.	-
nsr	Indicates the Non-stop routing information.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ospfv3	debug

Usage Guidelines

The **debugging packet ospfv3** command helps locate faults.

Example

# Enable OSPFv3 packet debugging.

```
<HUAWEI> debugging packet ospfv3 interface 10GE 1/0/3
OSPF:
---------------------------------------------
My Cid          : 0x8082043E
Peer Cid        : 0x8065042B
VS            : 0
Handle          : 2
TraceNum          : 141
Time            : 2015-8-4 11:11:46 389
Direction        : Down
Status          : 0
Version          : OSPFv3
Type            : Hello
Packet length    : 40
Router ID        : 10.10.10.1
Area ID          : 0.0.0.0
Checksum          : 0x8976
Instance ID      : 0
---------------------------------------------

SOCKET: -
```

```
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x8082043e
VS            : 0
Handle          : 2
TraceNum          : 141
Direction       : Down
Status         : 0
Time           : 2015-8-4 11:11:46 389
Data           :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x80782728
VS            : 0
Handle          : 2
TraceNum          : 141
Direction       : Down
Status         : 0
Time           : 2015-8-4 11:11:46 389
Data           :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x80782728
VS            : 0
Handle          : 2
TraceNum          : 127
Direction       : Up
Status         : 0
Time           : 2015-8-4 11:11:46 509
Data           :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x8065042b
Peer Cid        : 0x8082043e
VS            : 0
Handle          : 2
TraceNum          : 127
Direction       : Up
Status         : 0
Time           : 2015-8-4 11:11:46 509
Data           :
---------------------------------------------

OSPF:
---------------------------------------------
My Cid           : 0x8082043E
Peer Cid         : 0x8065042B
VS             : 0
Handle           : 2
TraceNum           : 127
Time            : 2015-8-4 11:11:46 509
Direction        : Up
Status          : 0
Version          : OSPFv3
Type            : Hello
Packet length    : 40
Router ID        : 10.10.10.2
Area ID          : 0.0.0.0
Checksum          : 0x254C
Instance ID      : 0
---------------------------------------------
```

```
LDM:
----------------------------------------------
My Cid          : 0x80782728
Peer Cid        : 0x8065042B
VS            : 0
Handle          : 2
TraceNum          : 141
Direction       : Down
Status          : 0
Interface index : 7
Link type       : -
Protocol        : IPV6
Time            : 2015-8-4 11:11:46 393
Data            : 0x6C00000000285901FE80000000000000002259E00
----------------------------------------------
```

Table 2-26 Description of the **debugging packet ospfv3** command output

Item	Description
My Cid	Self component ID
Peer Cid	Peer component ID
VR	VR ID
Handle	Handle value
TraceNum	Trace ID
Direction	Direction of packet flow: ● Up ● Down
Status	Status of the packet processing
Interface index	Interface index
Link type	Link type
Source mac	Source MAC address
Dest mac	Destination MAC address
Link protocol	Link layer protocol
Protocol	Protocol type: ● IPv4 ● IPv6
Time	Current time
Data	IP packet
Version	Version number
Type	OSPFv3 packet type
Packet length	Length of a packet

Item	Description
Router ID	Router ID of the originating router
Area ID	Area ID
Checksum	Checksum value of a packet
Instance ID	Instance ID

2.6.2.6 display debugging ospfv3

Function

The **display debugging ospfv3** command displays information about OSPFv3 debugging functions.

Format

display debugging ospfv3

Parameters

None.

Views

All views

Default Level

1: Monitor level

Usage Guidelines

When a large amount of information is output, the **display debugging ospfv3** command can be used to view information about the enabled OSPFv3 debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about OSPFv3 debugging functions.

```
<HUAWEI> display debugging ospfv3
OSPFv3 ROUTE-CALC debugging switch is on
```

2.6.3 RIP Debugging Commands

2.6.3.1 debugging rip

Function

The **debugging rip** command enables RIP debugging.

The **undo debugging rip** command disables RIP debugging.

By default, RIP debugging is disabled.

Format

debugging rip *process-id* [**error** | **event** | **jobs** | **backup**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

undo debugging rip *process-id* [**error** | **event** | **jobs** | **backup**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an RIP process.	The value is an integer ranging from 1 to 4294967295.
error	Enables error debugging.	-
event	Enables event debugging.	-
jobs	Enables job debugging.	-
backup	Enables backup debugging.	-
interface *interface-type interface-number*	Specifies the interface on which enables debugging.	-
peer *peer-address*	Specifies the IP address of an RIP neighbor.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
rip	debug

Usage Guidelines

None.

Example

# Enable RIP debugging.

```
<HUAWEI> debugging rip 1
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;RIP [ 1 ]: Received
add interface for interface index [ 5 ]
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Import
Cache Initialized
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Interface
Filter Policy Initialized
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Added
interface [ 0x5 ] to the process
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Received
Physical State [ 0 ] for the Interface [ 0x5 ]
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Received
MTU [ 1500 ] for Interface [ 0x5 ]
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Received
bandwidth [ 0x5f5e10000000 ] for Interface [ 0x5 ]
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;IP address change
notification is received for [ Ethernet3/0/0 ]
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Interface up
notification received for [ Ethernet3/0/0 ]
Dec 30 2011 04:39:48 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Received
Add Intf Network Configuration
```

2.6.3.2 debugging rip miscellaneous

Function

The **debugging rip miscellaneous** command enables debugging of information about interaction between the RIP device and other devices in the system.

The **undo debugging rip miscellaneous** command disables debugging of information about interaction between the RIP device and other devices in the system.

By default, debugging of information about interaction between the RIP device and other devices in the system is disabled.

Format

debugging rip miscellaneous

undo debugging rip miscellaneous

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
rip	debug

Usage Guidelines

Precautions

Too many debugging affects the system performance.

Example

# Enable debugging of information about interaction between the local device of RIP and other devices in the system.

```
<HUAWEI> debugging rip miscellaneous
Dec 30 2011 04:46:51 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Received message
from CFG
Dec 30 2011 04:46:51 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Sending message to
RM4. ( MsgType [ 12 ], TransNum [ 11 ], CID [ 0 ], MySeqNum [ 0 ] )
Dec 30 2011 04:46:51 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Adding Subscription
info for import configuration, TableID [ 1 ], ProtocolId [ 4 ], SubProtoId [ 0 ], ProcessId [ 0 ], PolicyId
[ 4294967295 ]
Dec 30 2011 04:46:51 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Applying Consumer
service for ProcessId [ 1 ], TopoId [ 0 ], TableId [ 3 ] to RM4
```

2.6.3.3 debugging rip packet

Function

The **debugging rip packet** command enables debugging of sent and received RIP packets.

The **undo debugging rip packet** command disables debugging of sent and received RIP packets.

By default, debugging of sent and received RIP packets is disabled.

Format

debugging rip *process-id* **packet** [**send** | **receive**] [**error**] [**verbose**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

undo debugging rip *process-id* **packet** [**send** | **receive**] [**error**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an RIP process.	The value is an integer ranging from 1 to 4294967295.
send	Displays sent RIP packets.	-
receive	Displays received RIP packets.	-
error	Displays error information about RIP packets.	-
verbose	Displays detailed information about RIP packets.	-
interface *interface-type interface-number*	Displays the interface type and interface number of a packet.	-
peer *peer-address*	Displays the IP address of an RIP neighbor in debugging information. If this parameter is not specified, the information about all neighbors is displayed by default.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
rip	debug

Usage Guidelines

Prerequisites

The RIP process has been enabled.

Precautions

Too many debugging affects the system performance.

Example

# Display the RIP packets sent and received by RIP process 1.

```
<HUAWEI> debugging rip 1 packet
Dec 30 2011 04:42:27 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;RIP Packet sent to
destination [ 255.255.255.255 ], from source [ 10.1.1.1 ]
Dec 30 2011 04:42:27 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;RIP [ 1 ]: Packet
sent out successfully!!
Dec 30 2011 04:42:27 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;RIP [ 1 ]: Sending
v1 response to 255.255.255.255 (IfIndx  [ 5 ]) from 10.1.1.1 with 1 RTE
Dec 30 2011 04:42:27 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;Packet:vers 1, cmd
response, length 24
Dec 30 2011 04:42:27 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;Dest 10.1.1.0, cost
1
```

2.6.3.4 debugging rip route

Function

The **debugging rip route** command enables debugging of RIP routes.

The **undo debugging rip route** command disables debugging of RIP routes.

By default, debugging of RIP routes is disabled.

Format

debugging rip *process-id* **route** [**error** | **backup**] [**imported** | { **interface** *interface-type interface-number* [**peer** *peer-address*] }]

undo debugging rip *process-id* **route** [**error** | **backup**] [**imported** | { **interface** *interface-type interface-number* [**peer** *peer-address*] }]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an RIP process.	The value is an integer ranging from 1 to 4294967295.
error	Displays error information about RIP routes.	-
backup	Displays information about the backup route.	-
imported	Displays information about imported routes.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Parameter	Description	Value
peer *peer-address*	Specifies the IP address of an RIP neighbor. If this parameter is not specified, debugging of all neighbors is enabled by default.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
rip	debug

Usage Guidelines

Prerequisites

The RIP process has been enabled.

Precautions

Too many debugging affects the system performance.

Example

# Display the route information about RIP process 1.

```
<HUAWEI> debugging rip 1 route
Dec 30 2011 04:44:03 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Route
[ 10.1.1.0/24 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ 10.1.1.2 ], received by DV
Dec 30 2011 04:44:30 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Route
[ 10.1.1.0/24 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ 10.1.1.2 ], received by DV
Dec 30 2011 04:44:37 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Route
[ 10.0.0.0/8 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ 10.1.1.2 ], received by DV
Dec 30 2011 04:44:37 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Path of
Route [ 10.0.0.0/8 ], NextHop [ 10.1.1.2 ] moved from [ no ] to [ age ] queue
Dec 30 2011 04:44:37 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Path of
Route [ 10.0.0.0/8 ], NextHop [ 0.0.0.0 ] moved from [ no ] to [ permanent ] queue
Dec 30 2011 04:44:37 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIP [ 1 ]: Backing
up learnt route [ 10.0.0.0/8 ], action [ ADD ]
```

2.6.4 RIPng Debugging Commands

CE6810LI does not support RIPng debugging commands.

2.6.4.1 debugging ripng

Function

The **debugging ripng** command enables RIPng debugging.

The **undo debugging ripng** command disables RIPng debugging.

By default, RIPng debugging is disabled.

Format

debugging ripng *process-id* [**error** | **event** | **jobs** | **backup**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

undo debugging ripng *process-id* [**error** | **event** | **jobs** | **backup**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an RIPng process.	The value is an integer ranging from 1 to 4294967295.
error	Enables error debugging.	-
event	Enables event debugging.	-
jobs	Enables job debugging.	-
backup	Enables backup debugging.	-
interface *interface-type interface-number*	Specifies the interface on which enables debugging.	-
peer *peer-address*	Specifies the IP address of an RIPng neighbor.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ripng	debug

Usage Guidelines

None.

Example

# Enable RIPng debugging.

```
<HUAWEI> debugging ripng 1
Dec 30 2011 06:22:15 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8016278d;RIPng [ 1 ]:
Periodic timer expired for target (Source Address [ FE80::3609:4FF:FE11:300 ])
```

2.6.4.2 debugging ripng packet

Function

The **debugging ripng packet** command enables debugging of sent and received RIPng packets.

The **undo debugging ripng packet** command disables debugging of sent and received RIPng packets.

By default, debugging of sent and received RIPng packets is disabled.

Format

debugging ripng *process-id* **packet** [**send** | **receive**] [**error**] [**verbose**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

undo debugging ripng *process-id* **packet** [**send** | **receive**] [**error**] [**interface** *interface-type interface-number* [**peer** *peer-address*]]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an RIPng process.	The value is an integer ranging from 1 to 4294967295.
send	Displays sent RIPng packets.	-
receive	Displays received RIPng packets.	-
error	Displays error information about RIPng packets.	-

Parameter	Description	Value
verbose	Displays detailed information about RIPng packets.	-
interface *interface-type interface-number*	Displays the interface type and interface number of a packet.	-
peer *peer-address*	Displays the IP address of an RIPng neighbor in debugging information. If this parameter is not specified, the information about all neighbors is displayed by default.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ripng	debug

Usage Guidelines

Prerequisites

The RIPng process has been enabled.

Precautions

Too many debugging affects the system performance.

Example

# Display the information of packets received over interface GigabitEthernet 6/0/0 in RIPng process 10.

```
<HUAWEI> debugging ripng 10 packet receive interface GigabitEthernet 6/0/0
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]:
Receiving v1 response on IfIndx [ 5 ] from FE80::3635:79FF:FE21:302 with 3 RTEs
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;Nexthop address
is ::
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;Dest
FC00:0:0:10::/24, cost 1, tag 0
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;Dest
FC00:0:0:60::/24, cost 1, tag 0
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801627a5;RIPng Packet sent
to destination [ FF02::9 ], from source [ FE80::3635:79FF:FE11:300 ]
```

```
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801627a5;RIPng [ 1 ]: Packet
sent out successfully!!
Aug 16 2011 23:48:08 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801627a5;
0x02010000  0x00000000  0x00000000  0x00000000
0x00000000  0x000000ff  0x00100000  0x00000000
0x00000000  0x00000000  0x00001801 [ 2814710437 ]: Packet sent out successfully!!
```

2.6.4.3 debugging ripng route

Function

The **debugging ripng route** command enables debugging of RIPng routes.

The **undo debugging ripng route** command disables debugging of RIPng routes.

By default, debugging of RIPng routes is disabled.

Format

debugging ripng *process-id* **route** [**error** | **backup**] [**imported** | { **interface** *interface-type interface-number* [**peer** *peer-address*] }]

undo debugging ripng *process-id* **route** [**error** | **backup**] [**imported** | { **interface** *interface-type interface-number* [**peer** *peer-address*] }]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an RIPng process.	The value is an integer ranging from 1 to 4294967295.
error	Displays error information about RIPng routes.	-
backup	Displays information about the backup route.	-
imported	Displays information about imported routes.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
peer *peer-address*	Specifies the IP address of an RIPng neighbor. If this parameter is not specified, debugging of all neighbors is enabled by default.	The value is in dotted decimal notation.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ripng	debug

Usage Guidelines

Prerequisites

The RIPng process has been enabled.

Precautions

Too many debugging affects the system performance.

Example

# Display the error route information about RIPng process 1.

```
<HUAWEI> debugging ripng 1 route error
Aug 16 2011 23:51:28 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]: Route
[ FC00:0:0:10::/64 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ FE80::3635:79FF:FE21:302 ], received by DV
Aug 16 2011 23:51:28 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]: Route
[ FC00:0:0:60::/64 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ FE80::3635:79FF:FE21:302 ], received by DV
Aug 16 2011 23:51:28 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]: Path of
Route [ FC00:0:0:60::/64 ], NextHop [ FE80::3635:79FF:FE21:302 ] moved from [ age ] to [ age ] queue
Aug 16 2011 23:51:31 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]: Route
[ FC00:0:0:10::/64 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ FE80::3635:79FF:FE21:300 ], received by DV
Aug 16 2011 23:51:31 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]: Route
[ FC00:0:0:60::/64 ], Cost [ 1 ], Tag [ 0 ], Neighbor [ FE80::3635:79FF:FE21:300 ], received by DV
Aug 16 2011 23:51:31 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x801527a7;RIPng [ 1 ]: Path of
Route [ FC00:0:0:60::/64 ], NextHop [ FE80::3635:79FF:FE21:300 ] moved from [ age ] to [ age ] queue
```

2.6.4.4 debugging ripng miscellaneous

Function

The **debugging ripng miscellaneous** command enables debugging of information about interaction between the RIPng device and other devices in the system.

The **undo debugging ripng miscellaneous** command disables debugging of information about interaction between the RIPng device and other devices in the system.

By default, debugging of information about interaction between the RIPng device and other devices in the system is disabled.

Format

debugging ripng miscellaneous

undo debugging ripng miscellaneous

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ripng	debug

Usage Guidelines

Precautions

Too many debugging affects the system performance.

Example

# Enable debugging of information about interaction between the local device of RIPng and other devices in the system.

```
<HUAWEI> debugging ripng miscellaneous
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Sending message to
RM6. ( MsgType [ 12 ], TransNum [ 17 ], CID [ 0 ], MySeqNum [ 0 ] )
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Adding Subscription
info for import configuration, TableID [ 1 ], ProtocolId [ 4 ], SubProtoId [ 0 ], ProcessId [ 0 ], PolicyId
[ 4294967295 ]
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Sending message to
RM6. ( MsgType [ 12 ], TransNum [ 18 ], CID [ 0 ], MySeqNum [ 0 ] )
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Adding Subscription
info for import configuration, TableID [ 3 ], ProtocolId [ 4 ], SubProtoId [ 0 ], ProcessId [ 0 ], PolicyId
[ 4294967295 ]
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIPNG [ 1 ]:
registered as consumer for TableID [ 1 ]
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Subscribing import
service for ProcessId [ 1 ], TopoId [ 0 ], TableId [ 1 ], ProtocolId [ 4 ] to RM6
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Sending message to
RM6. ( MsgType [ 14 ], TransNum [ 19 ], CID [ 6 ], MySeqNum [ 0 ] )
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Received Message
from RM6. ( MsgType [ 12 ], TransNum [ 18 ], RetCode [ 0 ] )
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;RIPNG [ 1 ]:
registered as consumer for TableID [ 3 ]
Dec 30 2011 06:27:11 HUAWEI %%01RIP/7/RIP_DBG_STRING(d):VS=0-CID=0x8015278f;Subscribing import
service for ProcessId [ 1 ], TopoId [ 0 ], TableId [ 3 ], ProtocolId [ 4 ] to RM6
```

2.6.5 IS-IS Debugging Commands

2.6.5.1 debugging isis adjacency

Function

The **debugging isis adjacency** command enables debugging of IS-IS adjacency.

The **undo debugging isis adjacency** command disables debugging of IS-IS adjacency.

By default, debugging of the IS-IS adjacency is disabled.

Format

debugging isis adjacency *process-id* [**interface** *interface-type interface-number*]

undo debugging isis adjacency *process-id* [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
interface-type interface-number	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-

Views

User view

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
isis	read

Usage Guidelines

The **debugging isis adjacency** command enables debugging of IS-IS adjacency so that you can locate the problem of neighbor disconnection.

Example

# Enable debugging of IS-IS adjacency on the Vlanif100 interface in an IS-IS process.

```
<HUAWEI> debugging isis adjacency 1 interface vlanif 100
```

# Send a level-1-2 hello packet from the broadcast network.

```
Dec 24 2011 14:45:11.466 HUAWEI %%01ISIS/6/TX_LAN_IIH(d):CID=0x80890423;ISIS-1-ADJ: Sending Lan
Level-1 IIH. (IfName=Vlanif100, LocalSnpa=38.00.10.03.00.11)
Dec 24 2011 14:45:11.466 HUAWEI %%01ISIS/6/TX_LAN_IIH(d):CID=0x80890423;ISIS-1-ADJ: Sending Lan
Level-2 IIH. (IfName=Vlanif100, LocalSnpa=38.00.10.03.00.11)
```

# Receive a level-1-1 hello packet from the broadcast network.

```
Dec 24 2011 14:56:11.237 HUAWEI %%01ISIS/6/RX_LAN_IIH(d):CID=0x80890423;ISIS-1-ADJ: Received Lan
Level-1 IIH. (IfName=Vlanif100, RemoteSnpa=38.00.10.03.00.05)
```

# Set the level of a neighbor.

```
Dec 24 2011 14:56:12.487 HUAWEI %%01ISIS/6/LAN_ADJ_USAGE(d):CID=0x80890423;ISIS-1-ADJ: Set LAN
ADJ usage to level-1-2. (IfName=Vlanif100, CircLevel=1-2, PduCircType =3)
```

Table 2-27 Description of the **debugging isis adjacency** command output

Item	Description
IfName	Interface name
LocalSnpa	MAC address of a local interface
RemoteSnpa	MAC address of a remote interface
CircLevel	Local port level
PduCircType	Local port type

2.6.5.2 debugging isis bfd

Function

The **debugging isis bfd** command enables debugging of IS-IS BFD.

The **undo debugging isis bfd** command disables debugging of IS-IS BFD.

By default, debugging of IS-IS BFD is disabled.

Format

debugging isis bfd *process-id*

undo debugging isis bfd *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis bfd** command enables debugging of IS-IS BFD so that you can detect changes of link status.

Example

# Enable debugging of IS-IS BFD.

```
<HUAWEI> debugging isis bfd 1
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/DIS_CHG_DISABLE_ADJ_BFD(d):VS=0-CID=2156472153;ISIS-1-
BFD: Circuit DIS change, disable adj bfd. (IfName=Vlanif100, Level=2)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/DIS_CHG_ENABLE_ADJ_BFD(d):VS=0-CID=2156472153;ISIS-1-
BFD: Circuit DIS change, enable adj bfd. (IfName=Vlanif100, Level=2)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/ENABLE_IPV4_ADJ_BFD(d):VS=0-CID=2156472153;ISIS-1-BFD:
Enable IPv4 adj bfd session. (IfName=Vlanif100, AdjSysId=2222.2222.2222, DestIpAddr=192.168.1.2, Level=2)
```

Table 2-28 Description of the **debugging isis bfd** command output

Item	Description
IfName	Interface name
Level	IS-IS level
AdjSysId	Adjacent system ID
DestIpAddr	Destination IP address

2.6.5.3 debugging isis circuit-information

Function

The **debugging isis circuit-information** command enables debugging of the IS-IS interface.

The **undo debugging isis circuit-information** command disables debugging of the IS-IS interface.

By default, debugging of the IS-IS interface is disabled.

Format

debugging isis circuit-information *process-id* [**interface** *interface-type interface-number*]

undo debugging isis circuit-information *process-id* [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
interface-type interface-number	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis circuit-information** command enables debugging of the IS-IS interface so that you can check whether the interface status is up or down.

Example

# Enable debugging of the IS-IS interface.

```
<HUAWEI> debugging isis circuit-information 1
```

# The current interface status is down.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/CIRC_STATE_DOWN(d):VS=0-CID=2156275547;ISIS-1-CIRC:
The state of circuit is down. (IfName=Vlanif100, AddrType=IPv4)
```

# The status of the IPv4 link on the interface changes from down to up.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/CIRC_LINK_UP(d):VS=0-CID=2156275547;ISIS-1-CIRC: Circuit
IPv4  link state change from down to up. (IfName=Vlanif100, OldCircState=126, NewCircState=127)
```

# The current interface status is up.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/CIRC_STATE_UP(d):VS=0-CID=2156275547;ISIS-1-CIRC: The
state of circuit is up. (IfName=Vlanif100, AddrType=IPv4)
```

# The status of the IPv4 link on the interface changes from up to down.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/CIRC_LINK_DOWN(d):VS=0-CID=2156275547;ISIS-1-CIRC:
Circuit IPv4  link state change from up to down. (IfName=Vlanif100)
```

Table 2-29 Description of the **debugging isis circuit-information** command output

Item	Description
IfName	Interface name
AddrType	Type of an IP address
OldCircState	Status before update
NewCircState	Current interface status

2.6.5.4 debugging isis import

Function

The **debugging isis import** command enables debugging for the routes imported by an IS-IS process.

The **undo debugging isis import** command disables debugging for the routes imported by an IS-IS process.

By default, debugging is disabled for the routes imported by an IS-IS process.

Format

debugging isis import *process-id* [**policy** { **ip-prefix** *ip-prefix-name* | **ipv6-prefix** *ipv6-prefix-name* }]

undo debugging isis import *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
policy	Indicates a policy to be used to filter debugging information.	-
ip-prefix *ip-prefix-name*	Specifies the name of an IPv4 prefix list to be used to filter debugging information.	The name is a string of 1 to 169 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string.
ipv6-prefix *ipv6-prefix-name*	Specifies the name of an IPv6 prefix list to be used to filter debugging information.	The name is a string of 1 to 169 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string.

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable debugging for the routes imported by an IS-IS process so that debugging information can be displayed to help troubleshooting, run the **debugging isis import** command.

Example

# Enable debugging for the routes imported by all IS-IS processes.

```
<HUAWEI> debugging isis import 1
Sep  2 2017 08:52:09.278 HUAWEI %%01ISIS/7/RM_RECV_MSG_NOTIFY(d):VS=Admin-VS-
CID=0x808700cf;ISIS-1-RM: Receive message from RM. (MsgType=subscribe batch update, Class=single
route, Message=)
Sep  2 2017 08:52:09.278 HUAWEI %%01ISIS/7/RM_MSG_FILTER_NOTIFY(d):VS=Admin-VS-
CID=0x808700cf;ISIS-1-RM: Communicate with RM through message. (MsgType=update, Class=single route,
ProtocolID=0, ProcessID=0, DestAddr=5.5.5.5, MaskLen=32)
Sep  2 2017 08:52:09.278 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=path nofrr, Iid=3254779923, Aid=0)
Sep  2 2017 08:52:09.278 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=path nofrr, Iid=3254779923, Aid=0)
Sep  2 2017 08:52:09.278 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=subscribe update,ClassID=import
path,ProcessID=0,ProtoID=4,Level=2,Metric=0)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=subscribe update,ClassID=import
path,ProcessID=0,ProtoID=4,Level=2,Metric=0)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=subscribe update,ClassID=import
path,ProcessID=0,ProtoID=4,Level=2,Metric=0)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=subscribe update,ClassID=import
path,ProcessID=0,ProtoID=4,Level=2,Metric=0)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=query,ClassID=single route)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_RECV_MSG_NOTIFY(d):VS=Admin-VS-
CID=0x808700cf;ISIS-1-RM: Receive message from RM. (MsgType=subscribe batch end, Class=null,
Message=)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=subscribe batch update
end,ClassID=subscribe policy)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_RECV_MSG_NOTIFY(d):VS=Admin-VS-
CID=0x808700cf;ISIS-1-RM: Receive message from RM. (MsgType=subscribe update, Class=import IID,
Message=)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_RECV_MSG_NOTIFY(d):VS=Admin-VS-
CID=0x808700cf;ISIS-1-RM: Receive message from RM. (MsgType=subscribe update, Class=update IID,
Message=iid=3254779923)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_MSG_NOTIFY(d):VS=Admin-VS-CID=0x808700cf;ISIS-1-
RM: Communicate with RM through message.(Message=MsgType=query,ClassID=subscribe policy)
Sep  2 2017 08:52:09.279 HUAWEI %%01ISIS/7/RM_RECV_MSG_NOTIFY(d):VS=Admin-VS-
CID=0x808700cf;ISIS-1-RM: Receive message from RM. (MsgType=subscribe batch end, Class=null,
Message=)
```

Table 2-30 Description of the **debugging isis import** command output

Item	Description
CID	Component ID
MsgType	Message type
Class	Class
ProtocolID	Protocol ID
ProcessID	Process ID
DestAddr	Destination IP address
MaskLen	Mask length
Iid	ECMP group ID
Aid	Attribute ID
ClassID	Class ID
ProtoID	Protocol ID

Item	Description
Level	IS-IS level
Metric	IS-IS cost

2.6.5.5 debugging isis receiving-packet-content

Function

The **debugging isis receiving-packet-content** command enables debugging of received IS-IS binary packets.

The **undo debugging isis receiving-packet-content** command disables debugging of received IS-IS binary packets.

By default, debugging of received IS-IS binary packets is disabled.

Format

debugging isis receiving-packet-content *process-id* [**interface** *interface-type interface-number*]

undo debugging isis receiving-packet-content *process-id* [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
interface-type interface-number	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis receiving-packet-content** command enables debugging of received IS-IS binary packets. When the binary format of received hello or LSP packets is configured, packets are not output if only **receiving-packet-content** is configured. If **adjacency** and **receiving-packet-content** are configured, the received hello packets are output. If **update-packet** and **receiving-packet-content** are configured, the received LSP and SNP packets are output.

Example

# Enable debugging of received IS-IS binary packets and adjacency.

```
<HUAWEI> debugging isis receiving-packet-content 1
<HUAWEI> debugging isis adjacency 1
```

# Receive hello packets in the binary format from the broadcast network.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/RX_LAN_IIH(d):VS=0-CID=2156472153;ISIS-1-ADJ: Received
Lan Level-2 IIH. (IfIndex=12, RemoteSnpa=36.33.8d.10.03.00)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0010 :03 83 1b 01 06
10 01 00 03 02 00 00 00 00 00 50
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0020 :00 1e 05 d9 40
00 00 00 00 00 01 01 01 02 01 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0030 :06 06 36 34 8d
10 03 00 84 04 01 01 01 09 e8 10
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0040 :fe 80 00 00 00
00 00 00 36 33 8d ff fe 10 03 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0050 :81 02 cc 8e e5
04 00 00 00 02 d3 03 00 00 00 08
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0060 :0e fe 0c 00 00
18 63 00 00 00 00 66 f1 68 88 08
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0070 :ff 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0080 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0090 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-00a0 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-00b0 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
```

Table 2-31 Description of the **debugging isis receiving-packet-content** and debugging isis adjacency command output

Item	Description
IfIndex	Index of an interface
RemoteSnpa	SNP packet at the remote end

2.6.5.6 debugging isis self-originate-update

Function

The **debugging isis self-originate-update** command enables debugging of IS-IS self-originate-update.

The **undo debugging isis self-originate-update** command disables debugging of IS-IS self-originate-update.

By default, debugging of IS-IS self-originate-update is disabled.

Format

debugging isis self-originate-update *process-id*

undo debugging isis self-originate-update *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis self-originate-update** command enables debugging of IS-IS self-originate-update so that you can view the information about IS-IS LSP update.

Example

# Enable debugging of IS-IS self-originate-update.

```
<HUAWEI> debugging isis self-originate-update 1
2011-02-25 01:41:07 HUAWEI %%01isiscomm/7/SELF_LSP_ADD_IP(d):VS=0-CID=2156472153;ISIS-1-UPDT:
Add IP address into LSP. (TlvType=128, Level=1, IPAddr=10.0.0.0)
2011-02-25 01:41:07 HUAWEI %%01isiscomm/7/RECV_CIRC_CHANGE(d):VS=0-CID=2156472153;ISIS-1-
UPDT: Rxed Ckt Down. (IfName=Ethernet3/0/2)
```

```
2011-02-25 01:41:07 HUAWEI %%01isiscomm/7/SELF_LSP_TIMER_EXPIRE(d):VS=0-CID=2156472153;ISIS-1-
UPDT: Lsp generation Intelligent timer expired. (Level=1)
```

Table 2-32 Description of the **debugging isis self-originate-update** command output

Item	Description
IfName	Interface name
Level	IS-IS level
TlvType	TLV type
IpAddr	Destination IP address

2.6.5.7 debugging isis sending-packet-content

Function

The **debugging isis sending-packet-content** command enables debugging of sent IS-IS binary packets.

The **undo debugging isis sending-packet-content** command disables debugging of sent IS-IS binary packets.

By default, debugging of sent IS-IS binary packets is disabled.

Format

debugging isis sending-packet-content *process-id* [**interface** *interface-type interface-number*]

undo debugging isis sending-packet-content *process-id* [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
interface-type interface-number	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis sending-packet-content** command enables debugging of sent IS-IS binary packets. When the binary format of sent hello or LSP packets is configured, the packets are not output if only **sending-packet-content** is configured. If **debugging isis adjacency** and **debugging isis sending-packet-content** are configured, the sent hello packets are output. If **debugging isis update packet** and **debugging isis sending-packet-content** are configured, the sent LSP and SNP packets are output.

Example

# Enable debugging of sent IS-IS binary packets and adjacency.

```
<HUAWEI> debugging isis sending-packet-content 1
<HUAWEI> debugging isis adjacency 1
```

# Send hello packets in the binary format to the broadcast network.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/TX_LAN_IIH(d):VS=0-CID=2156472153;ISIS-1-ADJ: Sending
Lan Level-2 IIH. (IfName=Vlanif100, LocalSnpa=36.19.8d.20.03.08)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0010 :03 83 1b 01 06
10 01 00 03 03 11 11 11 11 11 11
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0020 :00 1e 05 d9 40
11 11 11 11 11 11 02 01 02 01 10
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0030 :84 04 bd 10 8e
15 81 01 cc e5 02 00 00 d3 03 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0040 :00 00 08 0e fe
0c 00 00 46 2b 00 00 00 00 0d 95
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0050 :c0 b6 08 ff 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0060 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0070 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0080 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/IS_PDU(d):VS=0-CID=2156472153;ISIS-1-0090 :00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
```

Table 2-33 Description of the **debugging isis sending-packet-content** and **debugging isis adjacency** command output

Item	Description
IfName	Interface name
LocalSnpa	Local SNPA address

2.6.5.8 debugging isis snp-packet

Function

The **debugging isis snp-packet** command enables debugging of IS-IS SNP packets.

The **undo debugging isis snp-packet** command disables debugging of IS-IS SNP packets.

By default, debugging of IS-IS SNP packets is disabled.

Format

debugging isis snp-packet *process-id* [**interface** *interface-type interface-number*]

undo debugging isis snp-packet *process-id* [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
interface-type interface-number	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis snp-packet** command enables debugging of IS-IS SNP packets so that you can view sending, receiving, and processing of CSNP and PSNP packets.

Example

# Enable debugging of IS-IS SNP packets.

```
<HUAWEI> debugging isis snp-packet 1
```

# Send a packet with a complete serial number on the interface.

```
2011-02-24 12:13:54 HUAWEI %%01isiscomm/6/SEND_SNP_OK(d):VS=0-CID=2156275618;ISIS-1-SNP:
Succeed to send CSNP on circuit. (IfName=Ethernet3/0/2, Level=1)
```

# Enter the overload status, not setting the SRM flag.

```
2011-03-03 05:28:46 HUAWEI %%01isiscomm/6/FLOOD(d):VS=0-CID=2156275605;ISIS-2-SNP: Set SRM fail,
enter into overload state. (LspId=2222.2222.2222.00-00, PduLevel=2)
```

Table 2-34 Description of the **debugging isis snp-packet** command output

Item	Description
IfName	Interface name
Level	Interface level
PduLevel	Packet level

2.6.5.9 debugging isis spf-event

Function

The **debugging isis spf-event** command enables debugging of SPF events.

The **undo debugging isis spf-event** command disables debugging of SPF events.

By default, debugging of SPF events is disabled.

Format

debugging isis spf-event *process-id*]

undo debugging isis spf-event *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295. No default value is provided.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis spf-event** command enables debugging of SPF events so that you can view the creation and deletion of an SPF tree.

Example

# Enable debugging of SPF events.

```
<HUAWEI> debugging isis spf-event 1
```

# Delete the link between the source and the neighbor that is disconnected.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/SPF_LNK_PRO(d):VS=0-CID=2156406620;ISIS-1-SPF: Link
process, Destroy link. (Level=1, MtId=0, SrcNode=1111.1111.1111.00, DstNode=2222.2222.2222.02, Cost=10)
```

# Delete the direct-connect ID of the node when the neighbor is disconnected.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/SPF_NODE_PRO(d):VS=0-CID=2156406620;ISIS-1-SPF: Node
process, Clear DIRECT flag on node. (Level=1, MtId=0, Node=2222.2222.2222.02, Dist=10)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/SPF_DIS_PRO(d):VS=0-CID=2156406620;ISIS-1-SPF: DIS is
other system. (LocSysId=1111.1111.1111, LanId=2222.2222.2222.02)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/SPF_LNK_PRO(d):VS=0-CID=2156406620;ISIS-1-SPF: Link
process, Deattach adj on this link. (Level=1, MtId=0, SrcNode=2222.2222.2222.02,
DstNode=1111.1111.1111.00, Cost=0)
```

Table 2-35 Description of the **debugging isis spf-event** command output

Item	Description
Level	IS-IS level
MtId	Multi-topology ID

Item	Description
SrcNode	System ID of a local node
DstNode	System ID of a remote node
cost	Cost

2.6.5.10 debugging isis spf-prc

Function

The **debugging isis spf-prc** command enables debugging of the spf-prc calculation process. The **policy** parameter can be used to to filter the route calculation debugging information about the routing policy.

The **undo debugging isis spf-prc** command disables debugging of the spf-prc calculation process.

By default, debugging of spf-prc calculation process is disabled.

Format

debugging isis spf-prc *process-id* [**policy** { { **ip-prefix** | **ipv6-prefix** } *prefix-name* | { **ipv4-acl**| **ipv6-acl** } *acl-number* }]

undo debugging isis spf-prc *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
policy	Specifies a routing policy.	-
ip-prefix	Specifies an IPv4 prefix.	-
ipv6-prefix	Specifies an IPv6 prefix.	-
prefix-name	Specifies the prefix list name.	The prefix list name must already exist.
ipv4-acl	Specifies an IPv4 ACL.	-
ipv6-acl	Specifies an IPv6 ACL.	-

Parameter	Description	Value
acl-number	Specifies the basic ACL number.	The value is an integer ranging from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis spf-prc** command enables debugging of the spf-prc calculation process. For the IS-IS SPF route processing, the **policy** parameter can be used to filter the route calculation debugging information about the routing policy.

Example

# Enable debugging of the spf-prc calculation process.

```
<HUAWEI> debugging isis spf-prc 1
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/ICRM_RT_PRIORITY(d):CID=2156341082;ISIS-1-PRC: The
priority of 192.168.2.0/24 is 0.(MtId=0, Level=1)
```

# Instruct the route management module to delete the specified route.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/PRC_DEL_ROUT_TO_RM(d):CID=2156341082;ISIS-1-PRC:
Delete route entry to RM. (Addr=192.168.2.0, MaskLen=24)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/PRC_DEL_ROUT_TO_RM(d):CID=2156341082;ISIS-1-PRC:
Delete route entry to RM. (Addr=192.168.2.0, MaskLen=24)
```

# Add a route destined for the route management module.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/PRC_ADD_ROUT_TO_RM(d):CID=2156341082;ISIS-1-PRC: Add
route entry to RM. (Addr=192.168.2.0, MaskLen=24)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/ICRM_PROC_RT_ENTRY(d):CID=2156341082;ISIS-1-PRC:
Process L1 type 1 route entry 192.168.1.0/24 for layer 128.
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/ICRM_RT_PRIORITY(d):CID=2156341082;ISIS-1-PRC: The
priority of 192.168.1.0/24 is 0.(MtId=0, Level=1)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/PRC_DEL_ROUT_TO_RM(d):CID=2156341082;ISIS-1-PRC:
Delete route entry to RM. (Addr=192.168.1.0, MaskLen=24)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/PRC_DEL_ROUT_TO_RM(d):CID=2156341082;ISIS-1-PRC:
Delete route entry to RM. (Addr=192.168.1.0, MaskLen=24)
```

Table 2-36 Description of the **debugging isis spf-prc** command output

Item	Description
MtId	Multi-topology ID
Level	IS-IS level
Addr	IP Address

2.6.5.11 debugging isis update-packet

Function

The **debugging isis update-packet** command enables debugging of IS-IS update packets.

The **undo debugging isis update-packet** command disables debugging of IS-IS update packets.

By default, debugging of IS-IS update packets is disabled.

Format

debugging isis update-packet *process-id* [**interface** *interface-type interface-number*]

undo debugging isis update-packet *process-id* [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.
interface	Indicates that packets are displayed by interface.	-
interface-type interface-number	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

The **debugging isis update-packet** command enables debugging of IS-IS update packets. The packets received by the LSDB include LSP and SNP packets.

Example

# Enable debugging of IS-IS update packets.

```
<HUAWEI> debugging isis update-packet 1
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/SEND_PDU_ON_CIRC(d):VS=0-CID=2156275618;ISIS-1-SEND:
Send CSNP on circuit. (IfName=Vlanif100, Level=1)
```

# Delete a neighboring TLV.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/LSP_DEL_OPT(d):VS=0-CID=2156275618;ISIS-1-LSP: Delete
option from option list in option group. (TlvType=2)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/LSP_SET_SRM(d):VS=0-CID=2156275618;ISIS-1-LSP: Set SRM
flag. (LspId=2222.2222.2222.02-00, IfName=Vlanif100)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/LSP_CLER_SSN(d):VS=0-CID=2156275618;ISIS-1-LSP: Clear
SSN flag. (LspId=2222.2222.2222.02-00, IfName=Vlanif100)
```

# When there is no neighbor, LSP flooding is not performed.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/FLOOD_FAIL(d):VS=0-CID=2156275618;ISIS-1-SNP: Don't
flood, reason: no nbr, Not Flooding. (LspId=2222.2222.2222.02-00, PduLevel=2, IfName=Vlanif100)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/LSP_CLER_SSN(d):VS=0-CID=2156275618;ISIS-1-LSP: Clear
SSN flag. (LspId=2222.2222.2222.02-00, IfName=Vlanif100)
```

Table 2-37 Description of the **debugging isis update-packet** command output

Item	Description
IfName	Interface name
Level	IS-IS level
TlvType	TLV type
LspId	System ID of a received LSP packet

2.6.5.12 debugging isis update-process

Function

The **debugging isis update-process** command enables debugging of the IS-IS update process.

The **undo debugging isis update-process** command disables debugging of the IS-IS update process.

By default, debugging of the IS-IS update process is disabled.

Format

debugging isis update-process *process-id* [**policy** { { **ip-prefix** | **ipv6-prefix** } *prefix-name* | { **ipv4-acl** | **ipv6-acl** } *acl-number* }]

undo debugging isis update-process *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295. No default value is provided.
policy	Specifies the route policy, the commnad enables the debugging funciton for routes which pass the policy.	-
ip-prefix *prefix-name*	Specifies the name of the IPv4 prefix list.	The value is a string of case-sensitive characters without space and ranges from 1 to 169.
ipv6-prefix *prefix-name*	Specifies the name of the IPv6 prefix list.	The value is a string of case-sensitive characters without space and ranges from 1 to 169.
ipv4-acl *acl-number*	Specifies the IPv4 basic ACL.	The value is an integer ranging from 2000 to 2999.
ipv6-acl *acl-number*	Specifies the IPv6 basic ACL.	The value is an integer ranging from 2000 to 2999.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging isis update-process** command enables debugging of the IS-IS update process so that you can view the processing after the LSDB receives a packet.

Example

# Enable debugging of the IS-IS update process.

```
<HUAWEI> debugging isis update-process 1
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/RECV_CIRC_CHANGE(d):VS=0-CID=2156275547;ISIS-1-UPDT:
Receive circ change. (IfName=Vlanif100, ChangeType=Circ up)
```

# In the update flow, the notification indicating that the interface status is up is received.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/RECV_CIRC_CHANGE(d):VS=0-CID=2156275547;ISIS-1-UPDT:
Receive circ change. (IfName=Vlanif100, ChangeType=Circ up)
```

# The IS-IS does not generate any LSP packet if the minimum generation time is not reached.

```
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/MINLSPGEN_TMR_NOT_EXPIRED(d):VS=0-
CID=2156275547;ISIS-1-UPDT: MinLspGen Timer hasn not expired. Not generating LSP.
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/7/ADD_NBR_OPTION_IN_LSP(d):VS=0-CID=2156275547;ISIS-1-
UPDT: Add neighbour option in LSP. (TlvType=2, Level=1, NbrId=2222.2222.2222.01)
Dec 24 2011 21:11:460 HUAWEI %%01ISIS/6/RCV_LSP_NBR_SUCCESS(d):VS=0-CID=2156275547;ISIS-1-
UPDT: Succeed to parse Lsp neighbor.(LspId=2222.2222.2222.00-00, Neighbor=2222.2222.2222.01)
```

Table 2-38 Description of the **debugging isis update-process** command output

Item	Description
IfName	Interface name
ChangeType	Change type of an interface
TLVType	TLV type
LspId	ID of the local system
Neighbor	System ID of a neighbor

2.6.5.13 debugging packet isis interface

Function

The **debugging packet isis interface** command enables the function to debug packets on an IS-IS interface.

The **undo debugging packet isis interface** command disables the function to debug packets on an IS-IS interface.

By default, the function to debug packets on an IS-IS interface is disabled.

Format

debugging packet isis interface *interface-type interface-number* [**verbose**]

undo debugging packet isis interface *interface-type interface-number*

Parameters

Parameter	Description	Value
interface-type interface-number	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

View information about IS-IS packets on an interface.

Example

# Enable the function to debug packets on an IS-IS interface.

```
<HUAWEI> debugging packet isis interface Vlanif 100
```

2.6.5.14 display debugging isis

Function

The **display debugging isis** command displays information about current IS-IS debugging functions.

Format

display debugging isis

Parameters

None.

Views

All views

Default Level

1: Monitor level

Usage Guidelines

When a large amount of information is output, the **display debugging isis** command can be used to view information about the enabled IS-IS debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current IS-IS debugging functions, and you can view that the two debugging functions SPF-EVENTS and SPF-PRC are enabled.

```
<HUAWEI> display debugging isis
ISIS-1 SPF-EVENTS related debugging switch is on
ISIS-1 SPF-PRC debugging switch is on
```

2.6.5.15 undo debugging isis all

Function

The **undo debugging isis all** command disables all debugging functions of IS-IS.

Format

undo debugging isis all *process-id*

Parameters

Parameter	Description	Value
process-id	Specifies the ID of an IS-IS process.	The value is an integer ranging from 1 to 4294967295.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
isis	debug

Usage Guidelines

You can run the **undo debugging isis all** command to disable all debugging functions at a time, instead of disabling these functions one by one.

Example

# Disable all debugging functions of IS-IS.

```
<HUAWEI> undo debugging isis all 1
```

2.6.6 BGP Debugging Commands

NOTE

Only the CE5880EI, CE6850HI, CE6850U-HI, CE6851HI, CE6855HI, CE6856HI, CE6857EI, CE6860EI, CE6865EI, CE6870EI, CE6875EI, CE6880EI, CE6881, CE6881K, CE6820, CE6863, CE6863K, CE6881E, CE7850EI, CE7855EI, CE8850EI, CE8860EI, CE8861EI, CE8861P, and CE8868EI switches support multi-instance and instance parameter.

2.6.6.1 debugging bgp all

Function

The **debugging bgp all** command enables all BGP debugging functions.

The **undo debugging bgp all** command disables all BGP debugging functions.

Format

debugging bgp all [**peer** { *ipv4-address* | *ipv6-address* }]

undo debugging bgp all [**peer** { *ipv4-address* | *ipv6-address* }]

debugging bgp all [**vpn-instance** *vpn-instance-name* [**peer** *ipv4-address*]]

undo debugging bgp all [**vpn-instance** *vpn-instance-name* [**peer** *ipv4-address*]]

debugging bgp instance *instance-name* **all** [**peer** *ipv4-address*]

undo debugging bgp instance *instance-name* **all** [**peer** *ipv4-address*]

Parameters

Parameter	Description	Value
peer *ipv4-address*	Specifies the IP address of an IPv4 peer.	The value is in dotted decimal notation.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The VPN must already exist.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp all** command enables all BGP debugging functions. The debugging functions help you view the information about BGP packet sending and receiving, interaction with the socket process, change of neighbor state machines, and next-hop iteration.

When much information is output, you can filter the output information by VPN instance, peer.

Follow-up Procedure

Run the **undo debugging bgp all** command to disable all BGP debugging functions.

Example

# Enable all BGP debugging functions.

```
<HUAWEI>debugging bgp all
Dec 24 2011 15:12:06.528 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738910;
BGP(VPN 0): 10.1.1.2 changes state from ESTABLISHED to ESTABLISHED on event KA_TIMER.
```

2.6.6.2 debugging bgp event

Function

The **debugging bgp event** command enables debugging of BGP neighbor events.

The **undo debugging bgp event** command disables debugging of BGP neighbor events.

Format

debugging bgp event [**peer** { *ipv4-address* | *ipv6-address* }]

undo debugging bgp event [**peer** { *ipv4-address* | *ipv6-address* }]

debugging bgp event [**vpn-instance** *vpn-instance-name* [**peer** { *ipv4-address* | *ipv6-address* }]]

undo debugging bgp event [**vpn-instance** *vpn-instance-name* [**peer** { *ipv4-address* | *ipv6-address* }]]

debugging bgp instance *instance-name* **event** [**peer** *ipv4-address*]

undo debugging bgp instance *instance-name* **event** [**peer** *ipv4-address*]

Parameters

Parameter	Description	Value
peer *ipv4-address*	Specifies the IP address of an IPv4 peer.	The value is in dotted decimal notation.
peer *ipv6-address*	Specifies the address of the IPv6 peer.	The value is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The VPN must already exist.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value of *instance-name* can be an integer 1 or a string of 1 to 31 case-sensitive characters without spaces. The string can contain spaces if it is enclosed with double quotation marks ("). **NOTE** Each device can have only one BGP instance specified.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp event** command enables debugging of BGP neighbor events so that you can view changes of the neighbor state machine.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp event** command to disable debugging of BGP neighbor events.

Example

# Enable debugging of BGP neighbor events.

```
<HUAWEI>debugging bgp event
Apr  1 2014 14:19:10.941 CE12804-9183 %%01BGP/3/DEBUG_INFO(d):CID=0x8013044c;
 BGP(VPN 0): 10.1.1.2 changes state from CONNECT to CONNECTPEND on event TCP_SUC
CEED. (main socket)
```

2.6.6.3 debugging bgp graceful-restart

Function

The **debugging bgp graceful-restart** command enables debugging of the BGP graceful restart feature.

The **undo debugging bgp graceful-restart** command disables debugging of the graceful restart feature.

Format

debugging bgp graceful-restart [**peer** { *ipv4-address* | *ipv6-address* }]

undo debugging bgp graceful-restart [**peer** { *ipv4-address* | *ipv6-address* }]

debugging bgp graceful-restart [**vpn-instance** *vpn-instance-name* [**peer** { *ipv4-address* | *ipv6-address* }]]

undo debugging bgp graceful-restart [**vpn-instance** *vpn-instance-name* [**peer** { *ipv4-address* | *ipv6-address* }]]

debugging bgp instance *instance-name* **graceful-restart** [**peer** *ipv4-address*]

undo debugging bgp instance *instance-name* **graceful-restart** [**peer** *ipv4-address*]

Parameters

Parameter	Description	Value
peer *ipv4-address*	Specifies the IP address of an IPv4 peer.	The value is in dotted decimal notation.
peer *ipv6-address*	Specifies the address of the IPv6 peer.	The value is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The VPN must already exist.

Parameter	Description	Value
instance *instance-name*	Specifies the name of a BGP instance.	The value of *instance-name* can be an integer 1 or a string of 1 to 31 case-sensitive characters without spaces. The string can contain spaces if it is enclosed with double quotation marks ("). **NOTE** Each device can have only one BGP instance specified.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp graceful restart** command enables debugging of the graceful restart feature so that you can view the information during the BGP GR processing.

When much information is output, you can filter the output information by VPN instance, peer.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp graceful-restart** command to disable debugging of the graceful restart feature.

Example

# Enable debugging of the BGP graceful restart feature.

```
<HUAWEI>debugging bgp graceful-restart
Apr  1 2014 14:29:24.095 CE12804-9183 %%01BGP/3/DEBUG_INFO(d):CID=0x8013044c;
 BGP.GR(VPN 0): Recv OPEN with 'F' flag from 10.1.1.2, and negotiated address fa
```

```
mily success, AFI/SAFI: 1/1(IPv4-unicast).

Apr  1 2014 14:29:24.116 CE12804-9183 %%01BGP/3/DEBUG_INFO(d):CID=0x8014044d;
 BGP.GR(VPN 0)(IPv4-unicast): 10.1.1.2 received EOR
```

2.6.6.4 debugging bgp lsp

Function

The **debugging bgp lsp** command enables debugging of LSP and label related information in BGP.

The **undo debugging bgp lsp** command disables debugging of LSP and label related information in BGP.

Format

debugging bgp lsp

undo debugging bgp lsp

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp lsp** command enables debugging of LSP and label related information in BGP so that you can view the creation, deletion, and modification of a BGP LSP.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp lsp** command to disable debugging of LSP and label related information in BGP.

Example

# Enable debugging of LSP and label related information in BGP.

```
<HUAWEI>debugging bgp lsp
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148804447;
BGP.LSP(VPN 1)(IPv4-unicast): Delete ILM for 10.9.9.9/32,
         XcType  : bgp-lsp
         XcRole  : egress
         XcIndex1: 16
         XcIndex2: 0
        Inlabel : 0
         IidIndex: 0
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148804447;
BGP.LSP(VPN 1)(IPv4-unicast): Free label for 10.9.9.9/32, label: 16
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148804447;
BGP.LSP(VPN 1)(IPv4-unicast): Apply label for 10.9.9.9/32, label: 4294967295
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148804447;
BGP.LSP(VPN 1)(IPv4-unicast): Use label for 10.9.9.9/32, label: 48
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148804447;
BGP.LSP(VPN 1)(IPv4-unicast): Update ILM for 10.9.9.9/32,
         XcType  : bgp-lsp
         XcRole  : egress
         XcIndex1: 48
         XcIndex2: 0
         Inlabel : 48
         IidIndex: 3087007885
```

2.6.6.5 debugging bgp next-hop

Function

The **debugging bgp next-hop** command enables debugging of the BGP next hop.

The **undo debugging bgp next-hop** command disables debugging of the BGP next hop.

Format

debugging bgp [**instance** *instance-name*] **next-hop**

undo debugging bgp [**instance** *instance-name*] **next-hop**

Parameters

Parameter	Description	Value
instance *instance-name*	Specifies the name of a BGP instance.	The value of *instance-name* can be an integer 1 or a string of 1 to 31 case-sensitive characters without spaces. The string can contain spaces if it is enclosed with double quotation marks ("). **NOTE** Each device can have only one BGP instance specified.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging bgp next-hop** command enables debugging of the BGP next hop so that you can check the information during the BGP next hop processing.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp next-hop** command to disable debugging of the BGP next hop.

Example

# Enable debugging of the BGP next hop.

```
<HUAWEI>debugging bgp next-hop
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2150246245;
BGP.NHM(VPN 0)(IPv4-unicast): Successfully add nexthop 10.3.3.9(ip)
```

2.6.6.6 debugging bgp packet

Function

The **debugging bgp packet** command enables debugging of BGP packets.

The **undo debugging bgp packet** command disables debugging of BGP packets.

Format

debugging bgp packet { **all** | **keepalive** | **open** | **route-refresh** } [**peer** { *ipv4-address* | *ipv6-address* }] [**receive** | **send**]

undo debugging bgp packet { **all** | **keepalive** | **open** | **route-refresh** } [**peer** { *ipv4-address* | *ipv6-address* }] [**receive** | **send**]

debugging bgp packet { **all** | **keepalive** | **open** | **route-refresh** } [**vpn-instance** *vpn-instance-name* [**peer** *ipv4-address*]] [**receive** | **send**]

undo debugging bgp packet { **all** | **keepalive** | **open** | **route-refresh** } [**vpn-instance** *vpn-instance-name* [**peer** *ipv4-address*]] [**receive** | **send**]

debugging bgp packet update

undo debugging bgp packet update

debugging bgp packet update ipv4 { **unicast** | **multicast** } [**peer** *ipv4-address* | **ip-prefix** *ip-prefix-name*] [**receive** | **send**]

undo debugging bgp packet update ipv4 { **unicast** | **multicast** } [**peer** *ipv4-address* | **ip-prefix** *ip-prefix-name*] [**receive** | **send**]

debugging bgp packet update ipv6 [**peer** { *ipv4-address* | *ipv6-address* } | **ipv6-prefix** *ipv6-prefix-name*] [**receive** | **send**]

undo debugging bgp packet update ipv6 [**peer** { *ipv4-address* | *ipv6-address* } | **ipv6-prefix** *ipv6-prefix-name*] [**receive** | **send**]

debugging bgp packet update vpn-instance *vpn-instance-name* **ipv4-family** [**peer** *ipv4-address* | **ip-prefix** *ip-prefix-name*] [**receive** | **send**]

undo debugging bgp packet update vpn-instance *vpn-instance-name* **ipv4-family** [**peer** *ipv4-address* | **ip-prefix** *ip-prefix-name*] [**receive** | **send**]

debugging bgp packet update vpn-instance *vpn-instance-name* **ipv6-family** [**peer** *ipv6-address* | **ipv6-prefix** *ipv6-prefix-name*] [**receive** | **send**]

undo debugging bgp packet update vpn-instance *vpn-instance-name* **ipv6-family** [**peer** *ipv6-address* | **ipv6-prefix** *ipv6-prefix-name*] [**receive** | **send**]

debugging bgp packet update vpnv4 [**peer** *ipv4-address* | **ip-prefix** *ip-prefix-name*] [**receive** | **send**]

undo debugging bgp packet update vpnv4 [**peer** *ipv4-address* | **ip-prefix** *ip-prefix-name*] [**receive** | **send**]

debugging bgp packet update evn [**peer** *ipv4-address*] [**receive** | **send**]

undo debugging bgp packet update evn [**peer** *ipv4-address*] [**receive** | **send**]

debugging bgp [**instance** *instance-name*] **packet update evpn** [**peer** *ipv4-address*] [**receive** | **send**]

undo debugging bgp [**instance** *instance-name*] **packet update evpn** [**peer** *ipv4-address*] [**receive** | **send**]

debugging bgp instance *instance-name* **packet** { **all** | **keepalive** | **open** | **route-refresh** } [[**vpn-instance** *vpn-instance-name*] **peer** *ipv4-address*] [**receive** | **send**]

undo debugging bgp instance *instance-name* **packet** { **all** | **keepalive** | **open** | **route-refresh** } [[**vpn-instance** *vpn-instance-name*] **peer** *ipv4-address*] [**receive** | **send**]

debugging bgp packet update l2vpn-ad [**peer** *ipv4-address*] [**receive** | **send**]

undo debugging bgp packet update l2vpn-ad [**peer** *ipv4-address*] [**receive** | **send**]

debugging bgp packet update vpnv6 [**peer** *ipv4-address* | **ipv6-prefix** *ipv6-prefix-name*] [**receive** | **send**]

undo debugging bgp packet update vpnv6 [**peer** *ipv4-address* | **ipv6-prefix** *ipv6-prefix-name*] [**receive** | **send**]

debugging bgp packet update link-state unicast [**receive** | **send**] [**peer** *ipv4-address*]

undo debugging bgp packet update link-state unicast [**receive** | **send**] [**peer** *ipv4-address*]

debugging bgp packet update mvpn [**peer** *ipv4-address*] [**receive** | **send**]

undo debugging bgp packet update mvpn [**peer** *ipv4-address*] [**receive** | **send**]

Parameters

Parameter	Description	Value
all	Outputs debugging information about all packets.	-
update	Outputs debugging information about update packets.	-
open	Outputs debugging information about open packets.	-
keepalive	Outputs debugging information about Keepalive packets.	-
route-refresh	Outputs debugging information about route-refresh packets.	-
peer *ipv4-address*	Specifies the IP address of an IPv4 peer.	The value is in dotted decimal notation.
ip-prefix *ip-prefix-name*	Specifies the name of an IPv4 prefix list.	The ip-prefix-name must already exist.
peer *ipv6-address*	Specifies the IP address of an IPv6 peer.	The value is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
ipv6-prefix *ipv6-prefix-name*	Specifies the name of an IPv6 prefix list.	The ipv6-prefix-name must already exist.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The VPN must already exist.

Parameter	Description	Value
ipv4-family	Outputs debugging information about the IPv4 address family in the specified instance.	-
unicast	Outputs debugging information about the specified unicast address family.	-
multicast	Outputs debugging information about the specified multicast address family.	-
receive	Outputs debugging information about the specified packet receiving direction.	-
send	Outputs debugging information about the specified packet sending direction.	-
vpnv4	Indicates to display information about peers in a VPNv4 instance.	-
vpnv6	Indicates to display information about peers in a VPNv6 instance.	-
l2vpn-ad	Outputs debugging information about the L2VPN-AD route. **NOTE** Only the CE6850HI, CE6850U-HI, CE6851HI, CE6855HI, CE6856HI, CE6857EI, CE6860EI, CE6865EI, CE6870EI, CE6875EI, CE7850EI, CE7855EI, CE8850EI, CE8860EI, CE8861EI, CE8861P, and CE8868EI support this parameter.	-
evn	Outputs debugging information about the EVN route. **NOTE** Only the CE5880EI, CE6850HI, CE6850U-HI, CE6851HI, CE6855HI, CE6856HI, CE6857EI, CE6860EI, CE6865EI, CE6870EI, CE6875EI, CE6880EI, CE7850EI, CE7855EI, CE8860EI, CE8861EI, and CE8868EI support this parameter.	-

Parameter	Description	Value
link-state unicast	Outputs debugging information about the BGP-LS address family.	-
instance *instance-name*	Specifies the name of a BGP instance.	The value of *instance-name* can be an integer 1 or a string of 1 to 31 case-sensitive characters without spaces. The string can contain spaces if it is enclosed with double quotation marks ("). **NOTE** Each device can have only one BGP instance specified.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp packet** command enables debugging of BGP packets so that you can view the information during BGP packet receiving and sending.

When much information is output, you can filter the output information by VPN instance, peer, and address family.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp packet** command to disable debugging.

Example

# Display debugging information about BGP packets.

```
<HUAWEI> debugging bgp packet all
# A notification packet carrying the error code of 6/4 is received from peer 10.3.3.9.
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
 BGP.NM(VPN 0): Received NOTIFICATION from 10.3.3.9, Length: 21

Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
 Err/SubErr: 6/4 (CEASE/Administrative Reset)
 Error data: .
# An Open packet is sent to peer 10.3.3.9.
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
 BGP.NM(VPN 0): Sent OPEN to 10.3.3.9, Length: 45
 Version: 4, Remote AS: 100, HoldTime: 180, Router ID: 10.1.1.9
 TotOptLen: 16

        OPT TYPE:   2 (Capability)     , OPT LEN: 14
                CAP TYPE:   1 (Multiprotocol)  , CAP LEN:   4
                        MP-ext cap for IPv4-unicast
                CAP TYPE:   2 (RouteRefresh)   , CAP LEN:   0
                CAP TYPE:  65 (4-byte-as)       , CAP LEN:   4
                        AS NUMBER: 100
# An Open packet is received from peer 10.3.3.9.
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
BGP.NM(VPN 0): Received OPEN from 10.3.3.9, Length: 45
 Version: 4, Remote AS: 100, HoldTime: 180, Router ID: 10.3.3.9
 TotOptLen: 16

        OPT TYPE:   2 (Capability)     , OPT LEN: 14
                CAP TYPE:   1 (Multiprotocol)  , CAP LEN:   4
                        MP-ext cap for IPv4-unicast
                CAP TYPE:   2 (RouteRefresh)   , CAP LEN:   0
                CAP TYPE:  65 (4-byte-as)       , CAP LEN:   4
                        AS NUMBER: 100
# A Keepalive packet is sent to peer 10.3.3.9.
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
 BGP.NM(VPN 0): Sent KEEPALIVE to 10.3.3.9, Length: 19
# A Keepalive packet is received from peer 10.3.3.9.
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
 BGP.NM(VPN 0): Received KEEPALIVE from 10.3.3.9, Length: 19
# An Update packet with the prefix of 33.1.1.1/32 is received from peer 10.3.3.9.
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
 BGP.NM(VPN 0): Received UPDATE from 10.3.3.9, Length: 56

Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2150246245;
 BGP.RM(VPN 0): Received UPDATE from 10.3.3.9, Length: 56,
 Address family: IPv4-unicast

Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2150246245;
        Origin    : Incomplete
        As path   : NIL
        Next hop  : 10.3.3.9
        Med       : 88
        Local pref: 100

Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2150246245;
 NLRI Length: 5
 33.1.1.1/32,
```

Table 2-39 Description of the debugging bgp packet all command output

Item	Description
OPEN	Open message is the first message that is sent after a TCP connection is set up, and is used to set up BGP peer relationships.
KEEPALIVE	Keepalive messages are sent to the peer to ensure the connection validity.
UPDATE	Update messages are used to exchange routes between BGP peers.
NOTIFICATION	When a BGP device detects an error state, it sends a notification message to its peer. Then, the BGP connection between this BGP device and its peer will be closed.

2.6.6.7 debugging bgp raw-packet

Function

The **debugging bgp raw-packet** command enables debugging of BGP original packets.

The **undo debugging bgp raw-packet** command disables debugging of BGP original packets.

Format

debugging bgp raw-packet [**peer** *ipv4-address*] [**receive** | **send**]

undo debugging bgp raw-packet [**peer** *ipv4-address*] [**receive** | **send**]

debugging bgp raw-packet [**vpn-instance** *vpn-instance-name* [**peer** *ipv4-address*]] [**receive** | **send**]

undo debugging bgp raw-packet [**vpn-instance** *vpn-instance-name* [**peer** *ipv4-address*]] [**receive** | **send**]

Parameters

Parameter	Description	Value
peer *ipv4-address*	Specifies the IP address of an IPv4 peer.	The value is in dotted decimal notation.

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
receive	Outputs debugging information about the specified packet receiving direction.	-
send	Outputs debugging information about the specified packet sending direction.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp raw-packet** command enables debugging of BGP original packets so that you can view the information during BGP original packet receiving and sending.

When much information is output, you can filter the output information by VPN instance, peer.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp raw-packet** command to disable debugging of BGP original packets.

Example

# Enable debugging of BGP original packets.

```
<HUAWEI>debugging bgp raw-packet
Dec 24 2011 21:11:460 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=0-CID=2148738916;
BGP(VPN 0): Received message from 10.3.3.9
(Displaying bytes from 1 to 21)
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00 15 03 06 04
```

2.6.6.8 debugging bgp socket-process

Function

The **debugging bgp socket-process** command enables debugging of socket processing events for BGP neighbors.

The **undo debugging bgp socket-process** command disables debugging of socket processing events for BGP neighbors.

Format

debugging bgp socket-process [**peer** { *ipv4-address* | *ipv6-address* }]

undo debugging bgp socket-process [**peer** { *ipv4-address* | *ipv6-address* }]

debugging bgp socket-process [**vpn-instance** *vpn-instance-name* [**peer** { *ipv4-address* | *ipv6-address* }]]

undo debugging bgp socket-process [**vpn-instance** *vpn-instance-name* [**peer** { *ipv4-address* | *ipv6-address* }]]

debugging bgp instance *instance-name* **socket-process** [**peer** *ipv4-address*]

undo debugging bgp instance *instance-name* **socket-process** [**peer** *ipv4-address*]

Parameters

Parameter	Description	Value
peer *ipv4-address*	Specifies the IP address of an IPv4 peer.	The value is in dotted decimal notation.
peer *ipv6-address*	Specifies the address of the IPv6 peer.	The value is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The VPN must already exist.

Parameter	Description	Value
instance *instance-name*	Specifies the name of a BGP instance.	The value of *instance-name* can be an integer 1 or a string of 1 to 31 case-sensitive characters without spaces. The string can contain spaces if it is enclosed with double quotation marks ("). **NOTE** Each device can have only one BGP instance specified.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

The **debugging bgp socket-process** command enables debugging of socket processing events for BGP neighbors so that you can view the information about interaction between BGP and the socket process.

Precautions

Debugging information is output on the screen. Do not output too much information for other purposes than debugging so that the performance is not affected.

Follow-up Procedure

Run the **undo debugging bgp socket-process** command to disable debugging of socket processing events for BGP neighbors.

Example

# Enable debugging of socket processing events for BGP neighbors.

```
<HUAWEI>debugging bgp socket-process
Apr  1 2014 14:39:54.174 CE12804-9183 %%01BGP/3/DEBUG_INFO(d):CID=0x8013044c;
 BGP(VPN 0): 19 bytes are read on socket(1426) from 10.1.1.2.
```

2.6.6.9 debugging packet bgp

Function

The **debugging packet bgp** command enables the debugging of BGP packets.

The **undo debugging packet bgp** command disables the debugging of BGP packets.

By default, debugging of BGP packets is disabled.

Format

debugging packet bgp { *ipv4-address* | *ipv6-address* } [**verbose**]

undo debugging packet bgp { *ipv4-address* | *ipv6-address* } [**vpn-instance** *vpn-instance-name*]

Parameters

Parameter	Description	Value
ipv4-address	Specifies the IPv4 address of a peer.	The value is in dotted decimal notation.
ipv6-address	Specifies the IPv6 address of a peer.	The value is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.This parameter takes effect only in the diagnosis view.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
verbose	Displays detailed information.	-
nsr	Indicates non-stop routing.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
bgp	debug

Usage Guidelines

Usage Scenario

To check all-process information about BGP packets, run the **debugging packet bgp** command to enable the debugging function.

Precautions

After a debugging function is enabled, a great amount of debugging information will be displayed, degrading system performance. Therefore, disable the debugging function after it is complete.

Follow-up Procedure

Run the **undo debugging packet bgp** command to disable the debugging of BGP packets.

Example

# Display all-process information about BGP packets received by the peer with IP address 192.168.1.102.

```
<HUAWEI>debugging packet bgp 192.168.1.102
BGP:
----------------------------------------------
My Cid         : 0x8013041A
Peer Cid       : 0x80650402
VS             : 0
Handle         : 1
TraceNum       : 0
Direction      : Up
Status         : 0
Time           : 2013-3-13 18:54:8 915
Data           :
----------------------------------------------
```

2.6.6.10 debugging bmp packet all

Function

The **debugging bmp packet all** command enables BMP packet output debugging.

The **undo debugging bmp packet all** command disables BMP packet output debugging.

By default, BMP packet output debugging is disabled.

Format

debugging bmp packet all [**session** { *ipv4-address* | *ipv6-address* } [**alias** *alias-name*]]

undo debugging bmp packet all [**session** { *ipv4-address* | *ipv6-address* } [**alias** *alias-name*]]

Parameters

Parameter	Description	Value
session	Specifies the IPv4 address used for the BMP session.	-
ipv4-address	Specifies the IPv4 address used for the BMP session.	The value is in dotted decimal notation.
ipv6-address	Specifies the IPv6 address used for the BMP session.	The value is a 32-bit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
alias *alias-name*	Specifies the alias of a session.	The value is a string of 1 to 31 case-sensitive characters, spaces not supported. When double quotation marks are used around the string, spaces are allowed in the string.

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

If an exception occurs in a BMP session, run the **debugging bmp packet all** command to enable BMP packet output debugging. The command output can help you locate the problem.

Precautions

After a debugging function is enabled, a great amount of debugging information will be displayed, degrading system performance. Therefore, disable the debugging function after it is complete.

Example

# Enable BMP packet output debugging for the BMP session with 10.1.1.2 as the session address.

```
<HUAWEI> debugging bmp packet all session 10.1.1.2
Jul 31 2013 09:44:15.339 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=Admin-VS-CID=0x806c0430;
 BMP(VPN 0): Sent TERMINATION message to 10.1.1.2, Length: 12

Jul 31 2013 09:44:15.339 HUAWEI %%01BGP/3/DEBUG_INFO(d):VS=Admin-VS-CID=0x806c0430;
```

```
 (Displaying bytes from 1 to 12)
03 00 00 00 0C 05 00 01 00 02 00 00
```

2.6.6.11 display debugging bgp

Function

The **display debugging bgp** command displays information about the enabled BGP debugging functions.

Format

display debugging bgp

Parameters

None

Views

All views

Default Level

1: Monitor level

Task Name and Operations

Task Name	Operations
bgp	read

Usage Guidelines

When a large amount of information is output, the **display debugging bgp** command can be used to display information about the enabled BGP debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# View information about the enabled debugging functions.

```
<HUAWEI> display debugging bgp
 BGP Keepalives debugging is on:
    global
 BGP Opens debugging is on:
    global
 BGP Route-refresh debugging is on:
    global
 BGP Updates debugging is on:
    global
```

# View information about the enabled debugging functions.

```
<HUAWEI> display debugging bgp
```

```
BGP events debugging is on:
    peer 1.1.1.1
BGP Opens debugging is on:
    vrf1
```

# View information about the enabled debugging functions.

```
<HUAWEI> display debugging bgp
 BGP Updates debugging is on:
    _public_ ipv4 unicast
```

Table 2-40 Description of the display debugging bgp command output

Items	Description
BGP Keepalives debugging is on	BGP keepalive debugging is enabled.
BGP Opens debugging is on	BGP open debugging is enabled.
BGP Route-refresh debugging is on	BGP route refresh debugging is enabled.
BGP Updates debugging is on	BGP update debugging is enabled.
global	Global debugging is enabled. The contents of this field are described as follows: ● peer X.X.X.X: specifies the IP address of a peer. ● _public_ ipv4 unicast: enables debugging for packets in the BGP-IPv4 unicast address family. ● _public_ ipv6 unicast: enables debugging for packets in the BGP-IPv6 unicast address family. ● _public_ ipv4 multicast: enables debugging for packets in the BGP-IPv4 multicast address family. ● _public_ vpnv4: enables debugging for packets in the BGP-VPNv4 address family. ● _public_ vpnv6: enables debugging for packets in the BGP-VPNv6 address family. ● vrf1: enables debugging for packets in the VPN instance named **vrf1**. ● vrf1 ipv4 unicast: enables debugging for packets in the specified BGP-VPN instance IPv4 address family. ● vrf1 ipv6 unicast: enables debugging for packets in the specified BGP-VPN instance IPv6 address family.

2.6.6.12 display debugging bmp

Function

The **display debugging bmp** command displays information about enabled BMP debugging functions.

Format

display debugging bmp

Parameters

None

Views

All views

Default Level

1: Monitoring level

Usage Guidelines

When a large amount of information is output, you can run the **display debugging bmp** command to display information about enabled BMP debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about enabled BMP debugging functions.

```
<HUAWEI> display debugging bmp
[BMP]:session-msg(NULL) debugging switch is on
[BMP]:peer-notification(NULL) debugging switch is on
[BMP]:state-report(NULL) debugging switch is on
[BMP]:route-monitor(NULL) debugging switch is on
```

Table 2-41 Description of the **display debugging bmp** command output

Item	Description
session-msg(NULL) debugging switch is on	BMP session message debugging
peer-notification(NULL) debugging switch is on	BMP Peer-Notification message debugging
state-report(NULL) debugging switch is on	BMP Status-Report message debugging

Item	Description
route-monitor(NULL) debugging switch is on	BMP Route-Monitor message debugging

2.6.7 Route Management Debugging Commands

2.6.7.1 debugging directrt

Function

The **debugging directrt** command enables debugging of direct routes.

The **undo debugging directrt** command disables debugging of direct routes.

Format

debugging directrt

undo debugging directrt

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
route-base	debug

Usage Guidelines

To enable debugging of direct route components, run the **debugging directrt** command. In addition, you also need to run the **terminal debugging** command to enable terminal output of the system. Then the debugging information about the IPv4 component component of direct routes is displayed.

To disable debugging of direct route components, run the **undo debugging directrt** command. Then run the **undo terminal debugging** command to disable terminal output of the system.

Example

# Enable debugging of direct routes.

```
<HUAWEI>debugging directrt
<HUAWEI>terminal debugging
Info: Current terminal debugging is on.
<HUAWEI>
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Receive IFM message: MSG_IFMI_REAL_UPDATE(MSG_FLAG_COMMON).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: return ack for realnotify message to IFM.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: begin parse the real notify message!!!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Interface (10) exist.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Process interface(10) infomation(10)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: IFinfo Type is:(10).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: DR_IIFM_GetAddrInfo: Ifinfo type is (10)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: DR_IIFM_GetAddrInfo: address(3.3.3.3), masklen(24), flag(0), IPType(1).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: DR_IIFM_TransferIPv4Origin: The IP Address Origin is(0)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Add new address(3.3.3.3, 24).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: The flag of address is 0.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Create and Start Timer(Name: Request IID)!!!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Send IID Request Message.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Interface (10) exist.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Process interface(10) infomation(6)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: IF State: Old(0), New(1)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: the state of interface has changed from oldState(0) to newState(1)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Create JOB for updating route, VRF:0, TOPO:0 !.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Received MSG_DRTHAI_JOB_SCHEDULE Message.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Schedule for JOB(2).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: the prefix list is empty!!!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: DR_IGRESM_ProcMsg:uiMsgHeadLen(12), uiMsgLen(52).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Receive GRESM message: ENUM_MSG_GRESMI_APPLY_RESOURCE(GRESM_MSG_ACK).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: DR_IGRESM_ProcMsgTLV:Tlv type is (10)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Delete Timer!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Received GRESM RES_GRESMI_ALLOC_OK Message!!!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: save iid(4211081287) for address(3.3.3.3)!.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: save IID(4211081287) for address(3.3.3.3).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: INFO...DR_IM_SaveAddrIID: Save IID successfully.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Generate route to addtion for address 3.3.3.3(24,4211081287).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
```

```
DRT4: Generate host route  for address 3.3.3.3(24).
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: Generate host route for 3.3.3.3.
Dec 24 2011 21:11:460 HUAWEI %%01RM/6/DRT_COMMON_MSG_INFO(d):VS=0-CID=2154768191;
DRT4: pstRoutePrefix->auiPrefix = 3.3.3.3.
```

2.6.7.2 debugging rm

Function

The **debugging rm** command enables route management debugging. After the function is enabled, debugging information about route management is output on a screen.

The **undo debugging rm** command disables route management debugging.

Format

debugging rm ip { { **all** | **download** | **backup** | **producer** | **importer** | **subscriber** } [**ip-prefix** *ip-prefix-name*] | **event** }

undo debugging rm ip { **all** | **download** | **backup** | **producer** | **importer** | **subscriber** | **event** }

debugging rm ipv6 { { **all** | **download** | **backup** | **producer** | **importer** | **subscriber**} [**ipv6-prefix** *ipv6-prefix-name*] | **event** }

undo debugging rm ipv6 { **all** | **download** | **backup** | **producer** | **importer** | **subscriber** | **event** }

Parameters

Parameter	Description	Value
ip	Enables IP route debugging of the route management module.	-
all	Enables all debugging functions.	-
event	Enables information debugging for event handling processes defined in the route management module.	-
download	Enables information debugging for the route delivery process.	-
backup	Enables information debugging for the master/ slave backup process.	-
producer	Enables information debugging for producer-related processes.	-
importer	Enables information debugging for importer-related processes.	-
subscriber	Enables information debugging for subscriber-related processes.	-

Parameter	Description	Value
ip-prefix *ip-prefix-name*	Outputs only the debugging information corresponding to an IP address prefix.	-
ipv6	Enables IPv6 route debugging of the route management module.	-
ipv6-prefix *ipv6-prefix-name*	Outputs only the debugging information corresponding to an IPv6 address prefix.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
route-base	debug

Usage Guidelines

Usage Scenario

To check an internal route management process, you can enable debugging for the process, so that information about the process is output to a screen.

Different debugging functions are designed for different processes. The following debugging options are available:

- backup: If you select this option, debugging information about the master/slave backup process is displayed. This option can be selected to output relevant information for analysis when master/slave backup fails or data is inconsistent.
- subscriber: If you select this option, debugging information about subscriber-related processes is displayed. This option can be selected to output relevant information for analysis when a route subscriber cannot subscribe to routes.
- importer: If you select this option, debugging information about importer-related processes is displayed. This option can be selected to output relevant information for analysis when a route importer cannot import routes.
- download: If you select this option, debugging information about the route delivery process is displayed. This option can be selected to output relevant information for analysis when routes forwarded by the bottom layer are inconsistent with routes in route management.
- event: If you select this option, debugging information about event handling processes defined in the route management module is displayed. This option can be used with other options, so that event debugging is also enabled after debugging functions are enabled.

- producer: If you select this option, debugging information about producer-related processes is displayed. This option can be selected to output relevant information for analysis when the routing protocol exists but the route management module does not contain information about a route.
- all: If you select this option, debugging information about all processes is displayed. This option can be selected to output relevant information for analysis when you are not sure which option to select or the cause of a fault is not clear.

Prerequisites

Global terminal debugging must be simultaneously enabled for relevant debugging information to be output on the screen.

Example

# Enable all the debugging functions of the route management module.

```
<HUAWEI> debugging rm ip all
<HUAWEI> terminal debugging
Info: Current terminal debugging is on.
Dec 24 2011 21:11:460 HUAWEI %%01RM/3/RM_DEG_STRING(d):VS=0-CID=2154899267;A message was
received. (SN=[0], SendID=[0x6f2735],
INTF=[0x3], SUBINTF=[0x0], TotalLen=[60], MsgLen=[60], TransNo=[0],
MsgType=[MSG_RMI_ADD_PRODUCER], ucReserve=[0x0])
Dec 24 2011 21:11:460 HUAWEI %%01RM/3/RM_DEG_STRING(d):VS=0-CID=2154899267;A producer was
created.
PID=0x6f2735,VPID=0x0,Protocol=1,Process=0,Service=0x80000000,Ver=0,Vrf=1,AF=4,Topo=0,Table=1
```

# An error message has been sent.

```
0x6f2735 send a wrong msg
```

# A consumer message has been received but the protocol is unavailable.

```
Receive0x6f2735 consumer message, partner not avaliable
```

Table 2-42 Description of the **debugging rm** command output

Item	Description
SN	Sequence number
SendID	ID of the message sender
INTF	Interface ID
SUBINTF	Layer 3 Sub-interface ID
TotalLen	Total length of the message
MsgLen	Message length internally processed
TransNo	Transmit sequence number of the message
MsgType	Message type
ucReserve	Reserved bits
PID	Process ID

Item	Description
VPID	Virtual process ID used for iteration result query
Protocol	Protocol number
Process	Protocol process number
Service	Service type
Ver	Protocol version number
Vrf	VPN Instance ID
AF	IPv4 address family
Topo	Topology ID
Table	Table ID

2.6.7.3 debugging route-policy

Function

The **debugging route-policy** command enables debugging of the routing policy. After debugging of the routing policy is enabled, the debugging information about the routing policy is output on the screen.

The **undo debugging route-policy** command disables debugging of the routing policy.

Format

debugging route-policy

undo debugging route-policy

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
route-base	debug

Usage Guidelines

Usage Scenario

The **debugging route-policy** command enables debugging of the routing policy so that you can view the internal processes of the routing policy. The process information is output on the screen.

Prerequisites

Global terminal debugging has been enabled.

Example

# Enable debugging of the routing policy.

```
<HUAWEI>debugging route-policy
<HUAWEI>terminal debugging
Info: Current terminal debugging is on.
<HUAWEI>
Dec 24 2011 21:11:460 HUAWEI %%01RTP/7/DEBUG_RTP_ALL(d):VS=0-CID=2154899269;
RTP: rtp_api.c, 1588, Start to create config, filter type(0), index(4)

Dec 24 2011 21:11:460 HUAWEI %%01RTP/7/DEBUG_RTP_ALL(d):VS=0-CID=2154899269;
RTP: rtp_api.c, 1241, Increase filter reference Count, the input index=4, filter Type=0.

Dec 24 2011 21:11:460 HUAWEI %%01RTP/7/DEBUG_RTP_ALL(d):VS=0-CID=2154899269;
RTP: rtp_rp.c, 385, Create a new Route-policy (4)

Dec 24 2011 21:11:460 HUAWEI %%01RTP/7/DEBUG_RTP_ALL(d):VS=0-CID=2154899269;
RTP: rtp_rp.c, 2415, Increase route policy reference Count. name(), index(4), current count(1)

Dec 24 2011 21:11:460 HUAWEI %%01RTP/7/DEBUG_RTP_ALL(d):VS=0-CID=2154899269;
RTP: rtp_cfg_restore.c, 150, restore class = 0x938a21a, uiDBId = 0, usTblId = 538
```

2.6.7.4 debugging static-route

Function

The **debugging static-route** command enables debugging of static routes. After this command is run, the debugging information about the IPv4 component and the IPv6 component of static routes are output on the screen.

The **undo debugging static-route** command disables debugging of static routes.

Format

debugging static-route { **all** | **bfd** | **nexthop** | **prefix-update** | **iid-update** | **config** | **message** }

undo debugging static-route { **all** | **bfd** | **nexthop** | **prefix-update** | **iid-update** | **config** | **message** }

Parameters

Parameter	Description	Value
all	Enables all debugging functions.	-
bfd	Indicates information about BFD debugging.	-
nexthop	Indicates information about outbound interface state and next-hop iteration.	-
prefix-update	Indicates information about prefix update.	-
iid-update	Indicates information about IID update.	-
config	Indicates configuration information about static routes.	-
message	Indicates information about abnormal message receiving and sending.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
route-base	debug

Usage Guidelines

To enable debugging of the components of static routes, run the **debugging static-route** command. In addition, the **terminal debugging** command must be run to enable terminal output of the system.

To disable debugging of the components of static routes, run the **undo debugging static-route** command. Then run the **undo terminal debugging** command to disable terminal output of the system.

Example

# Enable debugging of static routes to view all of debugging information about static routes.

```
<HUAWEI>debugging static-route all
```

```
Dec 24 2011 21:11:460 HUAWEI %%01STATICRTBASE/6/SRT_STRING_INFO(d):VS=0-CID=0x80702741;[Add
route] Prefix=10.123.1.1/32(vrf=0,
topo=0, table=1);nexthop=10.59.60.60(vrf=0, topo=0, table=1), ifIndex=0xffffffff(phyType=0, linkProt=255);

Dec 24 2011 21:11:460 HUAWEI %%01STATICRTBASE/6/SRT_STRING_INFO(d):VS=0-CID=0x80702741;
[Route Attribute] preference=60, tag=0,
bfdEnable=0, localAddr=0.0.0.0, minRx=0, minTR=0, multi=0.
```

2.7 IP Multicast Debugging Commands

2.7.1 Debugging Commands of IGMP

NOTE

The CE6810LI does not support this feature.

2.7.1.1 debugging igmp all

Function

The **debugging igmp all** command enables all IGMP debugging functions.

The **undo debugging igmp all** command disables all IGMP debugging functions.

By default, all debugging functions of IGMP are disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **all**

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **all**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp all** command displays all IGMP signaling interworking between a switch and the member host. Output information includes events, Leave packets, Report packets, Query packets, timers, and SSM mapping.

Example

# Enable all IGMP debugging functions in the public network instance. The example shows the debugging information output when the group 225.0.0.1 is to be deleted.

```
<HUAWEI> debugging igmp all
2011-07-20 23:06:34 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Deleting group(225.0.0.1) on interface 0x5(736)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Deleting group(225.0.0.1) on interface 0x5
Line number	736

2.7.1.2 debugging igmp event

Function

The **debugging igmp event** command enables debugging of IGMP events.

The **debugging igmp event** command disables debugging of IGMP events.

By default, debugging of IGMP events is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **event** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **event** *advanced-acl-number*

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **event**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp event** command enables debugging of IGMP events.

Example

# Enable debugging of IGMP events in the public network instance. The example shows that the group 255.0.0.1 is created on the 0x5 interface.

```
<HUAWEI> debugging igmp event
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Creating group(225.0.0.1) for interface 0x5(885)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Creating group(225.0.0.1) for interface 0x5
Line number	885

2.7.1.3 debugging igmp leave

Function

The **debugging igmp leave** command enables debugging of IGMP Leave packets.

The **undo debugging igmp leave** command disables debugging of IGMP Leave packets.

By default, debugging of IGMP Leave packets is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **leave** [**group** *group-address* | **interface** *interface-type interface-number*] *

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **leave** *basic-acl-number*

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **leave**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-
basic-acl-number	Specifies the number of the basic ACL.	The value is an integer that ranges from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp leave** command enables debugging of IGMP Leave packets.

Example

# Enable debugging of IGMP Leave packets in the public network instance.

```
<HUAWEI> debugging igmp leave
2011-07-20 23:06:34 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Received LEAVE for group(225.0.0.1) on interface 0x5(20.0.5.121)(2545)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Received LEAVE for group(225.0.0.1) on interface 0x5(20.0.5.121)
Line number	2545

2.7.1.4 debugging igmp nsr

Function

The **debugging igmp nsr** command enables IGMP NSR debugging.

The **undo debugging igmp nsr** command disables IGMP NSR debugging.

By default, IGMP NSR debugging is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr** { **all** | **event** | **message** } [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.

Parameter	Description	Value
all-instance	Displays information about static IGMP entries in all instances.	-
all	Enables all IGMP NSR debugging.	-
event	Enables IGMP NSR event debugging.	-
message	Enables IGMP NSR message debugging.	-
source *source-address*	Specifies a multicast source address.	The value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group.	The value ranges from 224.0.1.0 to 239.255.255.255, in dotted decimal notation.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Precautions

- IGMP is enabled on at least one interface before the **debugging igmp nsr** command is run.
- If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically.

Example

# Enable all IGMP NSR debugging in the public network instance.

```
<HUAWEI> debugging igmp nsr all
Mar 8 2012 10:41:05.681 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=Admin-VS-
CID=0x80e22777;(VRFID=0): Backup the querier on interface 6(20.0.12.7). (2545)
```

Debugging Information	Description
Component ID	0x80e22777
VRF ID	0
Event	Backup the querier on interface 6(20.0.12.7).
Line number	2545

2.7.1.5 debugging igmp query

Function

The **debugging igmp query** command enables debugging of IGMP Query packets.

The **undo debugging igmp query** command disables debugging of IGMP Query packets.

By default, debugging of IGMP Query packets is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **query** [**group** *group-address* | [**receive** | **send**] | **interface** *interface-type interface-number*] *

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **query** *basic-acl-number* [**receive** | **send**]

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **query** [**receive** | **send**]

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
receive	Debugs received IGMP Query packets.	-
send	Debugs sent IGMP Query packets.	-
interface *interface-type interface-number*	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-
basic-acl-number	Specifies the number of the basic ACL.	The value is an integer that ranges from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp query** command enables debugging of IGMP Query packets.

Example

# Enable debugging of IGMP Query packets in the public network instance.

```
<HUAWEI> debugging igmp query
2011-07-20 23:07:18 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Send version 2 general query on (20.0.5.121) to destination 224.0.0.1(765)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Send version 2 general query on (20.0.5.121) to destination 224.0.0.1
Line number	765

2.7.1.6 debugging igmp report

Function

The **debugging igmp report** command enables debugging of IGMP Report packets.

The **undo debugging igmp report** command disables debugging of IGMP Report packets.

By default, debugging of IGMP Report packets is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **report** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*]*

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **report** *advanced-acl-number*

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **report**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp report** command enables debugging of IGMP Report packets.

Example

# Enable debugging of IGMP Report packets in the public network instance.

```
<HUAWEI> debugging igmp report
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Received v2 report for group 225.0.0.1 on interface 0x5(2205)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Received v2 report for group 225.0.0.1 on interface 0x5
Line number	2205

2.7.1.7 debugging igmp ssm-mapping

Function

The **debugging igmp ssm-mapping** command enables debugging of SSM mapping.

The **undo debugging igmp ssm-mapping** command disables debugging of SSM mapping.

By default, debugging of SSM mapping is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **ssm-mapping** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*]*

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **ssm-mapping** *advanced-acl-number*

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **ssm-mapping**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp ssm-mapping** command enables debugging of SSM mapping.

Example

# Enable debugging of IGMP SSM mapping in the public network instance. In the example, the timer of the IGMPV1 host expires and the group 225.0.0.1 does not support the IGMPV1 host any more.

```
<HUAWEI> debugging igmp ssm-mapping
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Changed compatibility for group 225.0.0.1 from v1 to v2(3225)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Changed compatibility for group 225.0.0.1 from v1 to v2
Line number	3225

2.7.1.8 debugging igmp timer

Function

The **debugging igmp timer** command enables debugging of IGMP timers.

The **undo debugging igmp timer** command disables debugging of IGMP timers.

By default, debugging of IGMP timers is disabled.

Format

debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **timer**

undo debugging igmp [**vpn-instance** *vpn-instance-name* | **all-instance**] **timer**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging igmp timer** command enables debugging of IGMP timers.

Example

# Enable debugging of IGMP timers in the public network instance.

```
<HUAWEI> debugging igmp timer
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Deleting v2 host timer for group 225.0.0.1(3244)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Deleting v2 host timer for group 225.0.0.1
Line number	3244

2.7.1.9 debugging packet igmp

Function

The **debugging packet igmp** command enables debugging of IGMP packet sending and receiving on the interface.

The **undo debugging packet igmp** command disables debugging of IGMP packet sending and receiving on the interface.

By default, debugging of IGMP packet sending and receiving on the interface is disabled.

Format

debugging packet igmp interface *interface-type interface-number* [**verbose**]

undo debugging packet igmp interface *interface-type interface-number* [**verbose**]

Parameters

Parameter	Description	Value
interface-type interface-number	Specifies the type and number of an interface.	-
verbose	Displays detailed information.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging packet igmp** command enables debugging of IGMP packet sending and receiving on the interface to support tracing of IGMP packet sending and receiving on the interface and display the information about sending and receiving of IGMP packets.

Example

# Enable debugging of packet sending and receiving on interface VLANIF 10.

```
<HUAWEI> debugging packet igmp interface vlanif 10 verbose
IGMP:
---------------------------------------------
My Cid          : 0x80E22789
Peer Cid        : 0x80652775
Handle          : 1
TraceNum        : 4
Direction       : Down
Status          : 0
PduLen          : 32
PduType         : Query
PduContent      : (Only Body)
11 64 ee 9b 00 00 00 00
---------------------------------------------


SOCKET: -
---------------------------------------------
My Cid          : 0x80652775
Peer Cid        : 0x80e22789
VS              : 0
Handle          : 1
TraceNum        : 4
Direction       : Down
Status          : 0
Data            :
46 c0 20 00 00 00 00 00 01 02 00 00 c0 a8 66 05
e0 00 00 01 94 04 00 00 11 64 ee 9b 00 00 00 00

---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x80652775
Peer Cid        : 0x782741
VS              : 0
Handle          : 1
TraceNum        : 4
Direction       : Down
Status          : 0
Data            :
46 c0 00 20 05 04 00 00 01 02 18 65 c0 a8 66 05
e0 00 00 01 94 04 00 00 11 64 ee 9b 00 00 00 00

---------------------------------------------

LDM:
---------------------------------------------
My Cid          : 0x80782776
Peer Cid        : 0x80652775
VS              : 0
Handle          : 1
TraceNum        : 4
Direction       : Down
Status          : 0
Interface index : 6
Link type       : -
Protocol        : IPV4
Time            : 2011-8-27 1:18:52 124
Data            : 0x46C00020050400000102 1865C0A86605E0000001940400001164EE9B0000
```

```
0000
---------------------------------
```

Item	Description
My Cid	ID of the message receiving component
Peer Cid	ID of the message sending component
Handle	Message handle
TraceNum	Sequence number of output messages
Direction	Sending direction
Status	Current message status
PduLen	Message length
PduType	Message type
PduContent	Message content
Interface index	Interface index
Protocol	Protocol type of a packet
Link type	Link type
Time	Packet timestamp

2.7.1.10 display debugging igmp

Function

The **display debugging igmp** command displays information about current IGMP debugging functions.

Format

display debugging igmp

Parameters

None.

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
igmp	debug

Usage Guidelines

When a large amount of information is output, the **display debugging igmp** command can be used to view information about the enabled IGMP debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current IGMP debugging functions.

```
<HUAWEI> display debugging igmp
```

2.7.2 Debugging Commands of MLD

NOTE

The CE6880EI, CE6810LI, CE5880EI and CE5855EI do not support this feature.

2.7.2.1 debugging mld all

Function

The **debugging mld all** command enables all MLD debugging functions.

The **undo debugging mld all** command disables all MLD debugging functions.

By default, all MLD debugging functions are disabled.

Format

debugging mld all

undo debugging mld all

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

The **debugging mld all** command enables all MLD debugging functions so that you can view all MLD signaling interworking processes between a router and the host, and sending as well as receiving of MLD protocol packets.

Example

# Enable all MLD debugging functions.

```
<HUAWEI> debugging mld all
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Send version 2 source-group specific query with s-bit on 0x5(20.0.5.121) for group(225.0.0.1),
source count 100(1704)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Send version 2 source-group specific query with s-bit on 0x5(20.0.5.121) for group(225.0.0.1), source count 100
Line number	1704

2.7.2.2 debugging mld done

Function

The **debugging mld done** command enables debugging of MLD Done packets.

The **undo debugging mld done** command disables debugging of MLD Done packets.

By default, debugging of MLD Done packets is disabled.

Format

debugging mld done [**group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging mld done *basic-acl-number*

undo debugging mld done

Parameters

Parameter	Description	Value
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-
basic-acl-number	Specifies the number of the basic ACL.	The value is an integer that ranges from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging mld done** command enables debugging of MLD Done packets.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of MLD Done packets.

```
<HUAWEI> debugging mld done
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Ignoring MLD done message on interface 0x5 from itself(2448)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Ignoring MLD done message on interface 0x5 from itself
Line number	2448

2.7.2.3 debugging mld event

Function

The **debugging mld event** command enables debugging of MLD events.

The **undo debugging mld event** command disables debugging of MLD events.

By default, debugging of MLD events is disabled.

Format

debugging mld event [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging mld event *advanced-acl-number*

undo debugging mld event

Parameters

Parameter	Description	Value
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

When a fault occurs after an IPv6 host joins a multicast group, the **debugging mld event** command enables debugging of MLD events so that you can view the processing of MLD events.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of MLD events.

```
<HUAWEI> debugging mld event
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Ignoring MLD version 1 report message on interface 0x5, source(20.0.5.121) is not proper(2511)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Ignoring MLD version 1 report message on interface 0x5, source(20.0.5.121) is not proper
Line number	2511

2.7.2.4 debugging mld nsr

Function

The **debugging mld nsr** command enables the debugging of MLD NSR.

The **undo debugging mld nsr** command disables the debugging of MLD NSR.

By default, the debugging of MLD NSR is disabled.

Format

debugging mld nsr { **all** | **event** | **message** } [**source** *ipv6–source-address* | **group** *ipv6–group-address* | **interface** *interface-type interface-number*]

debugging mld nsr { **all** | **event** | **message** } *advanced-acl-number*

undo debugging mld nsr { **all** | **event** | **message** }

Parameters

Parameter	Description	Value
all	Enables all the debugging of MLD NSR.	-
event	Enables the debugging of MLD NSR events.	-
message	Enables the debugging of MLD NSR messages.	-
source *ipv6–source-address*	Specifies the IPv6 address of a multicast source.	The value is in hexadecimal notation.
group *ipv6–group-address*	Specifies the IPv6 address of a multicast group.	The value is in hexadecimal notation and in the format of FFxA:xxxx:xxxx::xxxx, of which x ranges from 0 to F and A is 0 or ranges from 3 to F.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Precautions

- MLD is enabled on at least one interface before the **debugging mld nsr** command is run.
- If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically.

Example

# Enable all the debugging of MLD NSR.

```
<HUAWEI> debugging mld nsr all
Mar  8 2012 11:45:58.833 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=Admin-VS-
CID=0x80e2277d;IPv6:(VRFID=0): Backup the querier on interface 6(FE80:2000:37::7). (3589)
```

Debugging Information	Description
Component ID	0x80e2277d
VRF ID	0
Event	Backup the querier on interface 6(FE80:2000:37::7).
Line number	3589

2.7.2.5 debugging mld query

Function

The **debugging mld query** command enables debugging of MLD Query packets.

The **undo debugging mld query** command disables debugging of MLD Query packets.

By default, debugging of MLD Query packets is disabled.

Format

debugging mld query [**group** *ipv6-group-address* | [**receive** | **send**] | **interface** *interface-type interface-number*] *

debugging mld query *basic-acl-number* [**receive** | **send**]

undo debugging mld query [**receive** | **send**]

Parameters

Parameter	Description	Value
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
receive	Enables debugging of received MLD Query packets.	-
send	Enables debugging of sent MLD Query packets.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
basic-acl-number	Specifies the number of the basic ACL.	The value is an integer that ranges from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging mld query** command enables debugging of MLD Query packets.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of MLD Query packets.

```
<HUAWEI> debugging mld query
2011-07-20 23:07:18 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Send version 2 general query on 0x5(1::1) to destination FF02::1(1518)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Send version 2 general query on 0x5(1::1) to destination FF02::1
Line number	1518

2.7.2.6 debugging mld report

Function

The **debugging mld report** command enables debugging of MLD Report packets.

The **undo debugging mlp report** command disables debugging of MLD Report packets.

By default, debugging of MLD Report packets is disabled.

Format

debugging mld report [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging mld report *advanced-acl-number*

undo debugging mld report

Parameters

Parameter	Description	Value
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of the interface from which packets are received. *interface-type* indicates the type of the interface, and *interface-number* indicates the number of the interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

When a fault occurs after a user in the multicast network joins or leaves a multicast group, the **debugging mld report** command enables debugging of MLD Report packets so that you can view the detailed information about Report packets and locate the fault.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of MLD Report packets.

```
<HUAWEI> debugging mld report
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Receiving v2 report on interface 0x5 for destination address(FF02::16)(6355)
```

Debugging information	Description
VS	0

Debugging information	Description
Component ID	2162304911
VRF	0
Event	Receiving v2 report on interface 0x5 for destination address(FF02::16)
Line number	6355

2.7.2.7 debugging mld ssm-mapping

Function

The **debugging mld ssm-mapping** command enables debugging of IPv6 SSM mapping.

The **undo debugging mld ssm-mapping** command disables debugging of IPv6 SSM mapping.

By default, debugging of IPv6 SSM mapping is disabled.

Format

debugging mld ssm-mapping [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

undo debugging mld ssm-mapping

Parameters

Parameter	Description	Value
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

When the SSM mapping rule is configured but MLDv1 users cannot join a multicast group in the SSM, the **debugging mld ssm-mapping** command enables debugging of IPv6 SSM mapping so that you can locate faults based on the debugging information.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of MLD SSM mapping.

```
<HUAWEI> debugging mld ssm-mapping
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Changed compatibility for group FF33::0 from v1 to v2(3225)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Changed compatibility for group FF33::0 from v1 to v2
Line number	3225

2.7.2.8 debugging mld timer

Function

The **debugging mld timer** command enables debugging of MLD timers.

The **undo debugging mld timer** command disables debugging of MLD timers.

By default, debugging of MLD timers is disabled.

Format

debugging mld timer

undo debugging mld timer

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

When the group member relationship is abnormal, the **debugging mld timer** command enables debugging of MLD timers so that you can locate the fault together with other debugging information.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of MLD timers.

```
<HUAWEI> debugging mld timer
2011-07-21 01:01:12 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162304911;
(VRFID=0): Deleting v2 host timer for group FF56::1(3244)
```

Debugging information	Description
VS	0
Component ID	2162304911
VRF	0
Event	Deleting v2 host timer for group FF56::1

Debugging information	Description
Line number	3244

2.7.2.9 display debugging mld

Function

The **display debugging mld** command displays information about current MLD debugging functions.

Format

display debugging mld

Parameters

None.

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
mld	debug

Usage Guidelines

When a large amount of information is output, the **display debugging mld** command can be used to view information about the enabled MLD debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current MLD debugging functions.

```
<HUAWEI> display debugging mld
```

2.7.3 Debugging Commands of PIM (IPv4)

NOTE

The CE6810LI does not support this feature.

2.7.3.1 debugging packet pim

Function

The **debugging packet pim** command enables debugging of packet sending and receiving on PIM-enabled interfaces.

The **undo debugging packet pim** command disables debugging of packet sending and receiving from PIM-enabled interfaces.

By default, debugging of packet sending and receiving on PIM-enabled interfaces is disabled.

Format

debugging packet pim interface *interface-type interface-number* [**verbose** | **nsr**]

undo debugging packet pim interface *interface-type interface-number* [**verbose** | **nsr**]

Parameters

Parameter	Description	Value
interface-type interface-number	Specifies the type and number of an interface.	-
verbose	Displays detailed information.	-
nsr	Enables debugging of PIM NSR.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging packet pim** command enables debugging of packet sending and receiving on PIM-enabled interfaces to support tracing of PIM packet sending and receiving on the interface and display the information about sending and receiving of PIM packets.

Example

# Enable debugging of packet sending and receiving on the interface VLANIF 10.

```
<HUAWEI> debugging packet pim interface vlanif 10 verbose
PIM:
---------------------------------------------
My Cid          : 0x80DF2788
Peer Cid        : 0x80652775
Handle          : 2
TraceNum        : 8225
Direction       : Down
Status          : 0
PduLen          : 54
PduType         : HELLO
---------------------------------------------


SOCKET: -
---------------------------------------------
My Cid          : 0x80652775
Peer Cid        : 0x80df2788
VS              : 0
Handle          : 2
TraceNum        : 8225
Direction       : Down
Status          : 0
Data            :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid          : 0x80652775
Peer Cid        : 0x782741
VS              : 0
Handle          : 2
TraceNum        : 8225
Direction       : Down
Status          : 0
Data            :
---------------------------------------------

LDM:
---------------------------------------------
My Cid          : 0x80782776
Peer Cid        : 0x80652775
VS              : 0
Handle          : 2
TraceNum        : 8225
Direction       : Down
Status          : 0
Interface index : 6
Link type       : -
Protocol        : IPV4
Time            : 2011-8-27 2:21:57 204
---------------------------------------------
```

Item	Description
My Cid	ID of the message receiving component
Peer Cid	ID of the message sending component

Item	Description
Handle	Message handle
TraceNum	Sequence number of output messages
Direction	Sending direction
Status	Current message status
PduLen	Message length
PduType	Message type
PduContent	Message content
Interface index	Interface index
Protocol	Protocol type of a packet
Link type	Link type
Time	Packet timestamp

2.7.3.2 debugging pim all

Function

The **debugging pim all** command enables all debugging functions of the PIM protocol.

The **undo debugging pim all** command disables all debugging functions of the PIM protocol.

By default, all debugging functions of the PIM protocol are disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **all**

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **all**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.

Parameter	Description	Value
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim all** command displays the creation of an entire PIM network and outputs information such as creation of a PIM neighbor, transfer of RP information, RP election, multicast source registration, creation of a multicast distribution tree, assert election mechanism, state refresh, PIM BFD, PIM routing table, interworking between PIM and MSDP, and events.

Example

# Enable all debugging functions of the PIM protocol in the public network instance.

```
<HUAWEI> debugging pim all
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM ver 2 (null) sending 10.0.5.121 -> 10.5.5.5.(430)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM ver 2 (null) sending 10.0.5.121 -> 10.5.5.5.
Line number	430

2.7.3.3 debugging pim assert

Function

The **debugging pim assert** command enables debugging related to assert information in the PIM protocol.

The **undo debugging pim assert** command disables debugging related to assert information in the PIM protocol.

By default, debugging related to assert information in the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **assert** [[**receive** | **send**] | [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*]*]*

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **assert** [**receive** | **send**] *advanced-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **assert** [**receive** | **send**]

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
receive	Enables debugging related to received assert information in the PIM protocol.	-
send	Enables debugging related to sent assert information in the PIM protocol.	-

Parameter	Description	Value
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim assert** command enables debugging related to assert information in the PIM protocol.

Example

# Enable debugging related to assert information in the PIM protocol in the public network instance.

```
<HUAWEI> debugging pim assert
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Assert (10.0.5.121,225.1.1.1) 0x5 (10.1.1.1) FSM Winner->Winner, Inf (S, G) Asrt Rcvd.
(1354)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Assert (10.0.5.121,225.1.1.1) 0x5 (10.1.1.1) FSM Winner->Winner, Inf (S, G) Asrt Rcvd
Line number	1354

2.7.3.4 debugging pim bfd

Function

The **debugging pim bfd** command enables debugging of PIM BFD to debug creation and deletion of PIM BFD.

The **undo debugging pim bfd** command disables debugging of PIM BFD.

By default, debugging of PIM BFD is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **bfd** { **all** | **create** | **delete** | **event** } [**interface** *interface-type interface-number*]

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **bfd** { **all** | **create** | **delete** | **event** }

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-

Parameter	Description	Value
all	Enables all debugging functions of PIM BFD.	-
create	Enables debugging of PIM BFD session creation.	-
delete	Enables debugging of PIM BFD session deletion.	-
event	Enables debugging of PIM BFD session events.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

If PIM BFD sessions cannot be created or deleted or a PIM BFD session down event is received, the **debugging pim bfd** command enables debugging of creation and deletion of PIM BFD sessions and receiving of the PIM BFD session down event.

Example

# Enable debugging of creation of PIM BFD sessions in the public network instance.

```
<HUAWEI> debugging pim bfd create
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Create PIM BFD session for interface 0x19 and nbr 10.0.5.120.(97)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Create PIM BFD session for interface 0x19 and nbr 10.0.5.120.
Line number	97

2.7.3.5 debugging pim df

Function

The **debugging pim df** command enables debugging of Bidir-PIM messages for DF elections, including Offer, Backoff, Win, and Pass messages.

The **undo debugging pim df** command disable debugging of Bidir-PIM messages for DF elections.

By default, debugging of Bidir-PIM DF election is disabled.

Format

debugging pim df [[**send** | **receive**] | **rp** *rp-address* | **interface** *interface-type interface-number*] *

undo debugging pim df [**send** | **receive**]

Parameters

Parameter	Description	Value
send	Indicates debugging information about sent Bidir-PIM messages.	-
receive	Indicates debugging information about received Bidir-PIM messages.	-
rp *rp-address*	Specifies an RP address.	The value is in dotted decimal notation.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

None

Example

# Enable debugging of Bidir-PIM messages for DF elections.

```
<HUAWEI> debugging pim df
Mar 28 2013 05:43:49.926 HUAWEI %%01MCASTBASE/7/MCAST_DEBUG_INFO(d):CID=0x80de274c;
(VRFID=0): BIDIR PIM-SM: Send a Offer MessageRP(10.5.5.5), SenderPrefer(-1), S
enderMetric(-1), IfIndex(30)(369)
```

2.7.3.6 debugging pim event

Function

The **debugging pim event** command enables event debugging of the PIM protocol.

The **undo debugging pim event** command disables event debugging of the PIM protocol.

By default, event debugging of the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **event** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **event**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim event** command enables event debugging of the PIM protocol.

Example

# Enable event debugging of the PIM protocol in the public network instance.

```
<HUAWEI> debugging pim event
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0):PIM-SM: Downstream (10.0.5.0,225.1.1.1,rpt) FSM on interface 19 (10.0.5.121) transited
fromPruneTmp to NoInfo.(1086)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Downstream (10.0.5.0,225.1.1.1,rpt) FSM on interface 19 (10.0.5.121) transited from PruneTmp to NoInfo.
Line number	1086

2.7.3.7 debugging pim join-prune

Function

The **debugging pim join-prune** command enables debugging related to join and prune in the PIM protocol.

The **undo debugging pim join-prune** command disables debugging related to join and prune in the PIM protocol.

By default, debugging related to join and prune in the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **join-prune** [[**receive** | **send**] | [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *] *

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **join-prune** [**receive** | **send**] *advanced-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **join-prune** [**send** | **receive**]

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
send	Enables debugging related to sent neighbor information in the PIM protocol.	-
receive	Enables debugging related to received neighbor information in the PIM protocol.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim join-prune** command enables debugging related to join and prune in the PIM protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to join and prune in the PIM protocol in public network instances.

```
<HUAWEI> debugging pim join-prune
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Recv (10.0.5.100,225.0.0.0) join received on 19, upnrb(10.3.3.3) is not me.(558)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Recv (10.0.5.100,225.0.0.0) join received on 19, upnrb(10.3.3.3) is not me.
Line number	558

2.7.3.8 debugging pim msdp

Function

The **debugging pim msdp** command enables debugging of the information about interworking between MSDP and the PIM protocol.

The **undo debugging pim msdp** command disables debugging of the information about interaction between MSDP and the PIM protocol.

By default, debugging of the information about interworking between MSDP and the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **msdp** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **msdp** *advanced-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **msdp**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim msdp** command enables debugging of the information about interworking between MSDP and the PIM protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of the information about interworking between MSDP and the PIM protocol in public network instances.

```
<HUAWEI> debugging pim msdp
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Set 2MSDP flag for (10.1.1.1,225.1.1.1).(785)
```

2.7.3.9 debugging pim neighbor

Function

The **debugging pim neighbor** command enables debugging related to neighbor information in the PIM protocol.

The **undo debugging pim neighbor** command disables debugging related to neighbor information in the PIM protocol.

By default, debugging related to neighbor information in the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **neighbor** [[**send** | **receive**] | **source** *source-address* | **interface** *interface-type interface-number*] *

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **neighbor** [**receive** | **send**] *basic-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **neighbor** [**send** | **receive**]

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
basic-acl-number	Specifies the number of the basic ACL.	The value is an integer that ranges from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim neighbor** command enables debugging related to neighbor information in the PIM protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to neighbor information in the PIM protocol in public network instances.

```
<HUAWEI> debugging pim neighbor
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Too many neighbors, ignoring new neighbor 10.2.2.2.(1164)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Too many neighbors, ignoring new neighbor 10.2.2.2.
Line number	1164

2.7.3.10 debugging pim nsr

Function

The **debugging pim nsr** command enables debugging related to the NSR process.

The **undo debugging pim nsr** command disables debugging related to the NSR process.

By default, debugging related to the NSR process is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr** { **all** | **event** | **message** } [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr** { **all** | **event** | **message** } *advanced-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr** { **all** | **event** | **message** }

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
all	Enables all NSR debugging functions.	-
event	Enables the event debugging function.	-
message	Enables the message debugging function.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim nsr** command enables debugging related to the NSR process.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to the NSR process in public network instances.

```
<HUAWEI> debugging pim nsr all
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162173823;
(VRFID=0): PIM-SM: Start up Backup AUTO-RP job.(2393)
```

Debugging information	Description
VS	0
Component ID	2162173823
VRF	0
Event	PIM-SM: Start up Backup AUTO-RP job.
Line number	2393

2.7.3.11 debugging pim register

Function

The **debugging pim register** command enables debugging related to registration information in the PIM protocol.

The **undo debugging pim register** command disables debugging related to registration information in the PIM protocol.

By default, debugging related to registration information in the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **register** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **register** *advanced-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **register**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim register** command enables debugging related to registration information in the PIM protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to registration information in the PIM protocol in public network instances.

```
<HUAWEI> debugging pim register
2011-07-20 22:43:47 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Register:Encapsulated ip (10.0.5.100,225.1.1.1),len: 20. Border bit: false, Null bit: true(443)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Register:Encapsulated ip (10.0.5.100,225.1.1.1),len: 20. Border bit: false, Null bit: true
Line number	443

2.7.3.12 debugging pim routing-table

Function

The **debugging pim routing-table** command enables debugging of status change of the PIM routing table.

The **undo debugging pim routing-table** command disables debugging of status change of the PIM routing table.

By default, debugging of status change of the PIM routing table is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **routing-table** [**source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **routing-table** *advanced-acl-number*

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **routing-table**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim routing-table** command enables debugging of status change of the PIM routing table.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of status change of the PIM routing table in the public network instance.

```
<HUAWEI> debugging pim routing-table
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Deleting iif = 10.3.3.3 from (10.0.5.12,225.1.1.1).(439)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Deleting iif = 10.3.3.3 from (10.0.5.12,225.1.1.1).
Line number	439

2.7.3.13 debugging pim rp

Function

The **debugging pim rp** command enables debugging related to BSR and RP in the PIM protocol.

The **undo debugging pim rp** command disables debugging related to BSR and RP in the PIM protocol.

By default, debugging related to BSR and RP in the PIM protocol is disabled.

Format

debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **rp** [**send** | **receive**] [**interface** *interface-type interface-number*]

undo debugging pim [**vpn-instance** *vpn-instance-name* | **all-instance**] **rp** [**send** | **receive**]

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
send	Enables debugging related to sent RP information in the PIM protocol.	-
receive	Enables debugging related to received RP information in the PIM protocol.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim rp** command enables debugging related to BSR and RP in the PIM protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to BSR and RP in the PIM protocol in public network instances.

```
<HUAWEI> debugging pim rp
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Received a scoped BSM but we are not in any admin-scope region, ignored.(1465)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Received a scoped BSM but we are not in any admin-scope region, ignored.
Line number	1465

2.7.3.14 debugging pim state-refresh

Function

The **debugging pim state-refresh** command enables the debugging of the State-Refresh messages.

The **undo debugging pim state-refresh** command disables the debugging of the State-Refresh messages.

By default, the debugging of the State-Refresh messages is disabled.

Format

debugging pim state-refresh [[**receive** | **send**] | [**group** *group-address* | **source** *source-address* | **interface** *interface-type interface-number*] *] *

undo debugging pim state-refresh [**receive** | **send**]

Parameters

Parameter	Description	Value
receive	Enables the debugging of received State-Refresh messages.	-
send	Enables the debugging of sent State-Refresh messages.	-
group *group-address*	Enables a multicast group address. *group-address* specifies the address of a multicast group.	The value ranges from 224.0.1.0 to 239.255.255.255, in dotted decimal notation.
source *source-address*	Enables a multicast source address. *source-address* specifies the address of a multicast source.	The value is in dotted decimal notation.
interface *interface-type interface-number*	Enables messages on a specified interface. *interface-type interface-number* specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable the debugging of the State-Refresh messages, run the **debugging pim state-refresh** command.

Example

# Enable the debugging of the State-Refresh messages.

```
<HUAWEI> debugging pim state-refresh
Jul  6 2014 07:14:43.344 RT5 %%01PIM/7/PIMCORE_DEBUG_INFO(d):CID=0x80de2734;
(VRFID=0):
PIM ver 2 SRM receiving on interface 13,
Source address: 192.168.5.100, Group address: 225.1.1.1/32 flags:
00000000,
Originator address: 192.168.7.7, preference: 10, metric: 1, mask length:
24,
ttl: 254, prune indicator: set, prune now: unset, assert override: unset, interval: 60(622)
```

2.7.3.15 display debugging pim

Function

The **display debugging pim** command displays information about current PIM debugging functions.

Format

display debugging pim

Parameters

None.

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
pim	debug

Usage Guidelines

When a large amount of information is output, the **display debugging pim** command can be used to view information about the enabled PIM debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current PIM debugging functions.

```
<HUAWEI> display debugging pim
PIM(_public_) nsr message [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) nsr event [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) bfd event debugging switch is on
PIM(_public_) bfd delete debugging switch is on
PIM(_public_) bfd create debugging switch is on
PIM(_public_) rp receive debugging switch is on
PIM(_public_) rp send debugging switch is on
PIM(_public_) routing-table [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) register [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) neighbor receive [ Filter:(source=*) ] debugging switch is on
PIM(_public_) neighbor send [ Filter:(source=*) ] debugging switch is on
PIM(_public_) msdp [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) join-prune receive [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) join-prune send [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) event [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) assert receive [ Filter:(source=*, group=*) ] debugging switch is on
PIM(_public_) assert send [ Filter:(source=*, group=*) ] debugging switch is on
```

2.7.4 Debugging Commands of PIM (IPv6)

NOTE

The CE6880EI, CE6810LI, CE5880EI and CE5855EI do not support this feature.

2.7.4.1 debugging pim ipv6 all

Function

The **debugging pim ipv6 all** command enables all debugging functions of the PIM IPv6 protocol.

The **undo debugging pim ipv6 all** command disables all debugging functions of the PIM IPv6 protocol.

By default, all debugging functions of the PIM IPv6 protocol are disabled.

Format

debugging pim ipv6 all

undo debugging pim ipv6 all

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 all** command displays the creation of an entire PIM network and outputs information such as creation of a PIM neighbor, transfer of RP information, RP election, multicast source registration, creation of multicast distribution tree, assert election mechanism, PIM routing table, and events.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable all debugging functions of the PIM IPv6 protocol in public network instances.

```
<HUAWEI> debugging pim ipv6 all
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Received hello packet on 19 from non-local source: FC00:0:0:1::2.(761)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Received hello packet on 19 from non-local source: FC00:0:0:1::2.
Line number	761

2.7.4.2 debugging pim ipv6 assert

Function

The **debugging pim ipv6 assert** command enables debugging related to assert information in the PIM IPv6 protocol.

The **undo debugging pim ipv6 assert** command disables debugging related to assert information in the PIM IPv6 protocol.

By default, debugging related to assert information in the PIM IPv6 protocol is disabled.

Format

debugging pim ipv6 assert [[**send** | **receive**] | **source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging pim ipv6 assert [**send** | **receive**] *advanced-acl-number*

undo debugging pim ipv6 assert [**receive** | **send**]

Parameters

Parameter	Description	Value
send	Enables debugging of sent packets.	-

Parameter	Description	Value
receive	Enables debugging of received packets.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 assert** command enables debugging related to assert information in the PIM IPv6 protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to assert information in the PIM IPv6 protocol.

```
<HUAWEI> debugging pim ipv6 assert
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Assert (FC00:0:0:2000::5, FF56::1) 0x5 (FC00:0:0:1::1) FSM Winner->Winner, Inf (S, G)
Asrt Rcvd.(1354)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Assert (FC00:0:0:2000::5, FF56::1) 0x5 (FC00:0:0:1::1) FSM Winner->Winner, Inf (S, G) Asrt Rcvd.
Line number	1354

2.7.4.3 debugging pim ipv6 bfd

Function

The **debugging pim ipv6 bfd** command enables the debugging of IPv6 PIM BFD.

The **undo debugging pim ipv6 bfd** command disables the debugging of IPv6 PIM BFD.

By default, the debugging of IPv6 PIM BFD is disabled.

Format

debugging pim ipv6 bfd { **all** | **create** | **delete** | **event** } [**interface** *interface-type interface-number*]

undo debugging pim ipv6 bfd { **all** | **create** | **delete** | **event** }

Parameters

Parameter	Description	Value
all	Enables all the debugging of IPv6 PIM BFD.	-
create	Enables the debugging of IPv6 PIM BFD session creation.	-
delete	Enables the debugging of IPv6 PIM BFD session deletion.	-

Parameter	Description	Value
event	Enables the debugging of IPv6 PIM BFD session events.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

If you cannot create or delete an IPv6 PIM BFD session or receive an IPv6 PIM BFD session Down event, you can run the **debugging pim ipv6 bfd** command to enable the debugging of IPv6 PIM BFD.

Precautions

If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically.

Example

# Enable the debugging of IPv6 PIM BFD session creation in the public network instance.

```
<HUAWEI> debugging pim ipv6 bfd create
Mar 3 2012 05:00:51.123 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=Admin-VS-
CID=0x80eb277b;IPv6:(VRFID=0): Create PIM BFD session for interface 8 and nbr FE80:2000:35::5(100)
```

Debugging Information	Description
VS ID	0
Component ID	0x80eb277b
VRF ID	0
Event	Create PIM BFD session for interface 8 and nbr FE80:2000:35::5

Debugging Information	Description
Line number	100

2.7.4.4 debugging pim ipv6 df

Function

The **debugging pim ipv6 df** command enables the debugging of DF election-related control messages in IPv6 BIDIR-PIM, including Offer, Backoff, Win, and Pass messages.

The **undo debugging pim ipv6 df** command disables the debugging of DF election-related control control messages in IPv6 BIDIR-PIM.

By default, the debugging of DF election-related control messages is disabled.

Format

debugging pim ipv6 df [[**send** | **receive**] | **rp** *rp-address* | **interface** *interface-type interface-number*] *

undo debugging pim ipv6 df [**send** | **receive**]

Parameters

Parameter	Description	Value
send	Enables the debugging of sent IPv6 BIDIR-PIM control messages.	-
receive	Enables the debugging of received IPv6 BIDIR-PIM control messages.	-
rp *rp-address*	Enables the debugging for an RP with the specified address.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

To debug DF election-related control messages in IPv6 BIDIR-PIM, run the **debugging pim ipv6 df** command.

Example

# Enable the debugging of DF election-related control messages in IPv6 BIDIR-PIM in the public network instance.

```
<HUAWEI> debugging pim ipv6 df
Dec 10 1984 12:12:41.594 HUAWEI %%01MRM/7/MCAST_DEBUG_INFO(d):CID=0x80ea278f;IPv6:
(VRFID=0): BIDIR PIM-SM: Send a Offer MessageRP(FC00:0:0:2000:67::6), SenderPrefer(-1),
SenderMetric(-1), IfIndex(26)(404)
```

Item	Description
0x80ea278f	Component CID
0	VRF
BIDIR PIM-SM: Send a Offer MessageRP(FC00:0:0:2000:67::6), SenderPrefer(-1), SenderMetric(-1), IfIndex(26)	Event
404	Line number

2.7.4.5 debugging pim ipv6 event

Function

The **debugging pim ipv6 event** command enables event debugging of the PIM IPv6 protocol. When the features related to PIM such as PIM neighbor and PIM route are abnormal, the debugging information can be used to locate the fault.

The **undo debugging pim ipv6 event** command disables event debugging of the PIM IPv6 protocol.

By default, event debugging of the PIM IPv6 protocol is disabled.

Format

debugging pim ipv6 event [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging pim ipv6 event *advanced-acl-number*

undo debugging pim ipv6 event

Parameters

Parameter	Description	Value
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 event** command enables event debugging of the PIM IPv6 protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable event debugging of the PIM IPv6 protocol.

```
<HUAWEI> debugging pim ipv6 event
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0):PIM-SM: Downstream (FC00:0:0:2000::5, FF56::1, rpt) FSM on interface 19 (FC00:0:0:1::1)
transited from PruneTmp to NoInfo.(1086)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Downstream (FC00:0:0:2000::5, FF56::1, rpt) FSM on interface 19 (FC00:0:0:1::1) transited from PruneTmp to NoInfo.
Line number	1086

2.7.4.6 debugging pim ipv6 join-prune

Function

The **debugging pim ipv6 join-prune** command enables debugging related to join and prune in the PIM IPv6 protocol.

The **undo debugging pim ipv6 join-prune** command disables debugging related to join and prune in the PIM IPv6 protocol.

By default, debugging related to join and prune in the PIM IPv6 protocol is disabled.

Format

debugging pim ipv6 join-prune [[**send** | **receive**] | **source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging pim ipv6 join-prune [**send** | **receive**] *advanced-acl-number*

undo debugging pim ipv6 join-prune [**send** | **receive**]

Parameters

Parameter	Description	Value
send	Enables debugging of sent packets.	-
receive	Enables debugging of received packets.	-
source *source-address*	Specifies a multicast source.	*source-address* indicates the address of a multicast source and the value is in dotted decimal notation.

Parameter	Description	Value
group *group-address*	Specifies the address of a multicast group. In batch configuration mode, this parameter specifies the initial address of multicast group addresses.	*group-address* indicates the address of a multicast group. The value is in dotted decimal notation and ranges from 224.0.0.0 to 239.255.255.255.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 join-prune** command enables debugging related to join and prune in the PIM IPv6 protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to join and prune in the PIM IPv6 protocol.

```
<HUAWEI> debugging pim ipv6 join-prune
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Recv (FC00:0:0:1000::1,FF56::1) join received on 19, upnrb(FF80::1) is not me.(558)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Recv (1000::1,FF56::1) join received on 19, upnrb(FF80::1) is not me.
Line number	558

2.7.4.7 debugging pim ipv6 neighbor

Function

The **debugging pim ipv6 neighbor** command enables debugging related to neighbor information in the PIM IPv6 protocol.

The **undo debugging pim ipv6 neighbor** command disables debugging related to neighbor information in the PIM IPv6 protocol.

By default, debugging related to neighbor information in the PIM IPv6 protocol is disabled.

Format

debugging pim ipv6 neighbor [[**receive** | **send**] | [**source** *ipv6-source-address*] | [**interface** *interface-type interface-number*]] *

debugging pim ipv6 neighbor [**send** | **receive**] *basic-acl-number*

undo debugging pim ipv6 neighbor [**receive** | **send**]

Parameters

Parameter	Description	Value
receive	Enables debugging of received packets.	-
send	Enables debugging of sent packets.	-
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
basic-acl-number	Specifies the number of the basic ACL.	The value is an integer that ranges from 2000 to 2999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 neighbor** command enables debugging related to neighbor information in the PIM IPv6 protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to neighbor information in the PIM IPv6 protocol.

```
<HUAWEI> debugging pim ipv6 neighbor
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Too many neighbors, ignoring new neighbor FF80::401.(1164)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Too many neighbors, ignoring new neighbor FF80::401
Line number	1164

2.7.4.8 debugging pim ipv6 nsr

Function

The **debugging pim ipv6 nsr** command enables debugging related to the NSR process.

The **undo debugging pim ipv6 nsr** command disables debugging related to the NSR process.

By default, debugging related to the NSR process is disabled.

Format

debugging pim ipv6 nsr { **all** | **event** | **message** } [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging pim ipv6 nsr { **all** | **event** | **message** } *advanced-acl-number*

undo debugging pim ipv6 nsr { **all** | **event** | **message** }

Parameters

Parameter	Description	Value
all	Enables all NSR debugging functions.	-
event	Enables the event debugging function.	-
message	Enables the message debugging function.	-
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Parameter	Description	Value
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 nsr** command enables debugging related to the NSR process.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to the NSR process in public network instances.

```
<HUAWEI> debugging pim ipv6 nsr all source 1::1
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VSR=0-CID=2162173823;
(VRFID=0): PIM-SM: Start up Backup AUTO-RP job.(2393)
```

Debugging information	Description
VS	0
Component ID	2162173823
VRF	0
Event	PIM-SM: Start up Backup AUTO-RP job.
Line number	2393

2.7.4.9 debugging pim ipv6 register

Function

The **debugging pim ipv6 register** command enables debugging related to registration information in the PIM IPv6 protocol.

The **undo debugging pim ipv6 register** command disables debugging related to registration information in the PIM IPv6 protocol.

By default, debugging related to registration information in the PIM IPv6 protocol is disabled.

Format

debugging pim ipv6 register [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging pim ipv6 register *advanced-acl-number*

undo debugging pim ipv6 register

Parameters

Parameter	Description	Value
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 register** command enables debugging related to registration information in the PIM IPv6 protocol.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to registration information in the PIM IPv6 protocol in public network instances.

```
<HUAWEI> debugging pim ipv6 register
2011-07-20 22:43:47 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Register:Encapsulated ip (FC00:0:0:2000::1,FF56::1),len: 70. Border bit: false, Null bit: true(443)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Register:Encapsulated ip (FC00:0:0:2000::1,FF56::1),len: 70. Border bit: false, Null bit: true
Line number	443

2.7.4.10 debugging pim ipv6 routing-table

Function

The **debugging pim ipv6 routing-table** command enables debugging of status change of the PIM IPv6 routing table.

The **undo debugging pim ipv6 routing-table** command disables debugging of status change of the PIM IPv6 routing table.

By default, debugging of status change of the PIM IPv6 routing table is disabled.

Format

debugging pim ipv6 routing-table [**source** *ipv6-source-address* | **group** *ipv6-group-address* | **interface** *interface-type interface-number*] *

debugging pim ipv6 routing-table *advanced-acl-number*

undo debugging pim ipv6 routing-table

Parameters

Parameter	Description	Value
source *ipv6-source-address*	Specifies a multicast source.	*ipv6-source-address* indicates the address of a multicast source and the value is in hexadecimal.
group *ipv6-group-address*	Specifies the address of a multicast group.	*ipv6-group-address* indicates the address of a multicast group and the value is in hexadecimal.
interface *interface-type interface-number*	Specifies the type and number of an interface.	-
advanced-acl-number	Specifies the number of the advanced ACL.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 routing-table** command enables debugging of status change of the PIM IPv6 routing table.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging of status change of the PIM IPv6 routing table.

```
<HUAWEI> debugging pim ipv6 routing-table
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): PIM-SM: Deleting iif = FE80::26 from (FC00:0:0:2000::1,FF55::1).(439)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	PIM-SM: Deleting iif = FE80::26 from (FC00:0:0:2000::1,FF55::1).
Line number	439

2.7.4.11 debugging pim ipv6 rp

Function

The **debugging pim ipv6 rp** command enables debugging related to RP.

The **undo debugging pim ipv6 rp** command disables debugging related to RP.

By default, debugging related to RP is disabled.

Format

debugging pim ipv6 rp [[**send** | **receive**] | [**interface** *interface-type interface-number*]] *

undo debugging pim ipv6 rp [**receive** | **send**]

Parameters

Parameter	Description	Value
receive	Enables debugging of received packets.	-
send	Enables debugging of sent packets.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging pim ipv6 rp** command enables debugging related to RP.

Precautions

If debugging of all instances is enabled, debugging of new instances is enabled automatically.

Example

# Enable debugging related to BSR and RP in the PIM IPv6 protocol.

```
<HUAWEI> debugging pim ipv6 rp
2011-07-20 22:42:52 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162042753;
(VRFID=0): Received a scoped BSM but we are not in any admin-scope region, ignored.(1465)
```

Debugging information	Description
VS	0
Component ID	2162042753
VRF	0
Event	Received a scoped BSM but we are not in any admin-scope region, ignored.
Line number	1465

2.7.4.12 display debugging pim6

Function

The **display debugging pim6** command displays information about current PIM IPv6 debugging functions.

Format

display debugging pim6

Parameters

None.

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
pim6	debug

Usage Guidelines

When a large amount of information is output, the **display debugging pim6** command can be used to view information about the enabled PIM IPv6 debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current PIM IPv6 debugging functions.

```
<HUAWEI> display debugging pim6
PIM IPv6 bfd event debugging switch is on
PIM IPv6 bfd delete debugging switch is on
PIM IPv6 bfd create debugging switch is on
PIM IPv6 nsr message [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 nsr event [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 rp receive debugging switch is on
PIM IPv6 rp send debugging switch is on
PIM IPv6 routing-table [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 register [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 neighbor receive [ Filter:(source=*) ] debugging switch is on
PIM IPv6 neighbor send [ Filter:(source=*) ] debugging switch is on
PIM IPv6 join-prune receive [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 join-prune send [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 event [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 assert receive [ Filter:(source=*, group=*) ] debugging switch is on
PIM IPv6 assert send [ Filter:(source=*, group=*) ] debugging switch is on
```

2.7.5 Debugging Commands of MSDP

NOTE

The CE6810LI does not support this feature.

2.7.5.1 debugging msdp all

Function

The **debugging msdp all** command enables all MSDP debugging functions.

The **undo debugging msdp all** command disables all MSDP debugging functions.

By default, all MSDP debugging functions are disabled.

Format

debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **all**

undo debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **all**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging msdp all** command enables all MSDP debugging functions and outputs information such as connect, event, packet, and source-action.

Example

# Enable all MSDP debugging functions in the public network instance.

```
<HUAWEI> debugging msdp all
2011-07-23 03:07:02 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162370449;
(VRFID=0): 56.1.1.5: TCP listening. (783)
```

Debugging information	Description
VS	0
Component ID	2162370449
VRF	0
Event	56.1.1.5: TCP listening.
Line number	783

2.7.5.2 debugging msdp connect

Function

The **debugging msdp connect** command enables debugging of connection reset of MSDP peers.

The **undo debugging msdp connect** command disables debugging of connection reset of MSDP peers.

By default, debugging of connection reset of MSDP peers is disabled.

Format

debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **connect**

undo debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **connect**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging msdp connect** command enables debugging of connection reset of MSDP peers.

Example

# Enable debugging of connection reset of MSDP peers in the public network instance.

```
<HUAWEI> debugging msdp connect
2011-07-23 03:07:38 RT6 %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162370449;
(VRFID=0): 56.1.1.5: TCP connection established. (844)
```

Debugging information	Description
VS	0
Component CID	2162370449
VRF	0
Event	56.1.1.5: TCP connection established.
Line number	844

2.7.5.3 debugging msdp event

Function

The **debugging msdp event** command enables debugging of MSDP events.

The **undo debugging msdp event** command disables debugging of MSDP events.

By default, debugging of MSDP events is disabled.

Format

debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **event**

undo debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **event**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging msdp event** command enables debugging of MSDP events.

Example

# Enable debugging of MSDP events in the public network instance.

```
<HUAWEI> debugging msdp event
2011-07-23 03:07:02 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162370449;
(VRFID=0): 56.1.1.5: Peer's AS is the next-AS to RP 5.5.5.5. (290)
```

Debugging information	Description
VS	0
Component ID	2162370449

Debugging information	Description
VRF	0
Event	56.1.1.5: Peer's AS is the next-AS to RP 5.5.5.5.
Line number	290

2.7.5.4 debugging msdp nsr

Function

The **debugging msdp nsr** command enables debugging of MSDP NSR.

The **undo debugging msdp nsr** command disables debugging of MSDP NSR.

By default, debugging of MSDP NSR is disabled.

Format

debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr** { **all** | **event** | **message** }

undo debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **nsr** { **all** | **event** | **message** }

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Displays information about static IGMP entries in a specified VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Displays information about static IGMP entries in all instances.	-
all	Enables all NSR debugging functions.	-
event	Enables the event debugging function.	-

Parameter	Description	Value
message	Enables the message debugging function.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging msdp nsr** command enables debugging of MSDP NSR.

Example

# Enable debugging of MSDP NSR in the public network instance.

```
<HUAWEI> debugging msdp nsr all
2011-07-23 03:07:38 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162370449;
(VRFID=0): Receiving nsr event notification: MSG_HA_NEW_BACKUP. (64)
```

Debugging information	Description
VS	0
Component ID	2162370449
VRF	0
Event	Receiving nsr event notification: MSG_HA_NEW_BACKUP.
Line number	64

2.7.5.5 debugging msdp packet

Function

The **debugging msdp packet** command enables debugging of MSDP packets.

The **undo debugging msdp packet** command disables debugging of MSDP packets.

By default, debugging of MSDP packets is disabled.

Format

debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **packet**

undo debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **packet**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging msdp packet** command enables debugging of MSDP packets.

Example

# Enable debugging of MSDP packets in the public network instance.

```
<HUAWEI> debugging msdp packet
2011-07-23 03:10:38 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162370449;
(VRFID=0): 56.1.1.5: Received 3-bytes message 3 from peer. (461)
```

Debugging information	Description
VS	0
Component ID	2162370449
VRF	0
Event	56.1.1.5: Received 3-bytes message 3 from peer.
Line number	461

2.7.5.6 debugging msdp source-active

Function

The **debugging msdp source-active** command enables debugging of the MSDP active source.

The **undo debugging msdp source-active** command disables debugging of the MSDP active source.

By default, debugging of the MSDP active source is disabled.

Format

debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **source-active**

undo debugging msdp [**vpn-instance** *vpn-instance-name* | **all-instance**] **source-active**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Specifies all the instances.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mcast-common	debug

Usage Guidelines

Usage Scenario

The **debugging msdp source-active** command enables debugging of the MSDP active source.

Example

# Enable debugging of the MSDP active source in the public network instance.

```
<HUAWEI> debugging msdp source-active
2011-07-23 03:07:02 HUAWEI %%01MCASTAFSBASE/7/MCAST_DEBUG_INFO(d):VS=0-CID=2162370449;
(VRFID=0): 56.1.1.5: Originating SA message for peer. (925)
```

Debugging information	Description
VS	0
Component ID	2162370449
VRF	0
Event	56.1.1.5: Originating SA message for peer.
Line number	925

2.7.5.7 display debugging msdp

Function

The **display debugging msdp** command displays information about current MSDP debugging functions.

Format

display debugging msdp

Parameters

None.

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
msdp	debug

Usage Guidelines

When a large amount of information is output, the **display debugging msdp** command can be used to view information about the enabled MSDP debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current MSDP debugging functions.

```
<HUAWEI> display debugging msdp
```

2.7.6 Debugging Commands of IGMP Snooping

2.7.6.1 debugging igmp snooping

Function

The **debugging igmp snooping** command enables IGMP snooping debugging on the live network during operation and maintenance and outputs related debugging information.

The **undo debugging igmp snooping** command disables IGMP snooping debugging on the live network during operation and maintenance.

Format

debugging igmp snooping all

debugging igmp snooping { **event** | **timer** | **packet** | **report** | **query** | **nsr** { **all** | **event** | **message** } } [[**vlan** *vlan-id* | **bridge-domain** *bd-id*] | **interface** *interface-type interface-number* | **source** *ip-address* | **group** *group-address*] *

debugging igmp snooping leave [[**vlan** *vlan-id* | **bridge-domain** *bd-id*] | **interface** *interface-type interface-number* | **group** *group-address*] *

debugging igmp snooping { **leave** *basic-acl-number* | { **query** | **report** | **packet** } *advanced-acl-number* }

undo debugging igmp snooping { **all** | **event** | **timer** | **packet** | **report** | **query** | **leave** | **nsr** { **all** | **event** | **message** } }

Parameters

Parameter	Description	Value
all	Enables or disables all debugging functions of IGMP snooping.	-
event	Enables or disables debugging of IGMP snooping events.	-
timer	Enables or disables debugging of IGMP snooping timers, for example, debugging of Layer 2 multicast (IPv4) group timers, source timers, and inter-board communication timers.	-
packet	Enables or disables debugging of received IGMP snooping packets.	-
report	Enables or disables debugging of received IGMP snooping report packets.	-
query	Enables or disables debugging of received IGMP snooping query packets.	-
leave	Enables or disables debugging of received IGMP snooping leave packets.	-

Parameter	Description	Value
nsr { **all** | **event** | **message** }	Enables or disables debugging of IGMP snooping for active/standby switchovers. ● **nsr all** enables or disables debugging of IGMP snooping for active/standby switchover events and messages. ● **nsr event** enables or disables debugging of IGMP snooping for active/standby switchover events. ● **nsr message** enables or disables debugging of IGMP snooping for active/standby switchover messages.	-
vlan *vlan-id*	Enables or disables debugging of IGMP snooping for a specified VLAN.	The value is an integer ranging from 1 to 4094.
bridge-domain *bd-id*	Enables or disables debugging of IGMP snooping for a specified BD. **NOTE** Only VXLAN-supported devices support this parameter.	The value is an integer that ranges from 1 to 16777215.
interface *interface-type interface-number*	Enables or disables debugging of IGMP snooping for a specified interface.	-
source *ip-address*	Specifies a multicast source address.	The value can be a Class A, Class B, or Class C address, in dotted decimal notation.
group *group-address*	Specifies a multicast group address.	The value ranges from 224.0.1.0 to 239.255.255.255, in dotted decimal notation.
basic-acl-number	Specifies the basic ACL number.	The value is an integer that ranges from 2000 to 2999.

Parameter	Description	Value
advanced-acl-number	Specifies the advanced ACL number.	The value is an integer that ranges from 3000 to 3999.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging igmp snooping** command enables IGMP snooping debugging and diagnosis, and the **undo debugging igmp snooping** command disables IGMP snooping debugging and diagnosis.

Example

# In the user view, enable debugging of all IGMP snooping information.

```
<HUAWEI> debugging igmp snooping all
Apr 10 2012 16:19:39.320.17 huawei 238 SNPG/7/EVENT:Get PW by label success:
VsiIndex=0,InLabel=4125,LspToken=1073758852,ulOutIfIndex=600,TunnelID=1610830182,OutLabel=4096,Tnl
Num=1 (L2MC_SH_VSI1239)
Apr 10 2012 16:19:39.320.19 huawei 238 SNPG/7/QUERY: Proxy receive general query on port(VSI zg1),
reply IGMPv3 report. (L2MC_PROTO_IGMP3024)
Apr 10 2012 16:19:39.320.20 huawei 238 SNPG/7/QUERY:L2MC Proxy recieve general query from VSI zg1.
(L2MCPROXY4304)
Apr 10 2012 16:19:39.320.22 huawei 238 SNPG/7/PACKET:SNPG Forward IPV4 packet, source port is:0
0x101d (L2MC_PROTO_IGMP2457)
Apr 10 2012 16:19:39.320.23 huawei 238 SNPG/7/QUERY: Querier receive IGMP general query on main
board, discard it (L2MC_PROTO_IGMP3209)
Apr 10 2012 16:19:39.330.1 huawei 238 SNPG/7/PACKET:IGMP-snooping receive packet, ulInlabel 0x101d,
type 2, VID 1, CE 0. (L2MC_PKT1445)
Apr 10 2012 16:19:39.330.10 huawei 238 SNPG/7/EVENT: SNPG_ProcessL2EventPacket,ulL2EvtQueNum:3
(L2MC_INIT1373)
```

2.7.6.2 display debugging snpg

Function

The **display debugging snpg** command displays information about current IGMP snooping debugging functions.

Format

display debugging snpg

Parameters

None.

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
snpg	debug

Usage Guidelines

When a large amount of information is output, the **display debugging snpg** command can be used to view information about the enabled IGMP snooping debugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display information about current IGMP snooping debugging functions.

```
<HUAWEI> display debugging snpg
```

2.7.7 MLD Snooping Debugging Commands

NOTE

The CE6880EI, CE6881, CE6881K, CE6820, CE6863, CE6863K, CE6881E, CE5880EI and CE5855EI do not support this feature.

2.7.7.1 debugging mld snooping

Function

The **debugging mld snooping** command enables debugging of MLD snooping.

The **undo debugging mld snooping** command disables debugging of MLD snooping.

By default, debugging of MLD snooping is disabled.

Format

debugging mld snooping all

undo debugging mld snooping all

debugging mld snooping { **query** | **report** | **event** | **timer** | **packet** | **nsr** { **all** | **event** | **message** } } [**vlan** *vlan-id* | **source** *source-address* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging mld snooping done [**vlan** *vlan-id* | **group** *group-address* | **interface** *interface-type interface-number*] *

debugging mld snooping { **done** *basic-acl-number* | { **query** | **report** | **packet** } *advanced-acl-number*}

undo debugging mld snooping { **query** | **report** | **done** | **event** | **timer** | **packet** | **nsr** { **all** | **event** | **message** } }

Parameters

Parameter	Description	Value
all	Enables or disables all debugging functions of MLD snooping.	-
query	Enables or disables debugging of received MLD snooping query packets.	-
report	Enables or disables debugging of received MLD snooping report packets.	-
done	Enables or disables debugging of received MLD snooping done packets.	-
event	Enables or disables debugging of MLD snooping events.	-
timer	Enables or disables debugging of MLD snooping timers.	-
packet	Enables or disables debugging of received MLD snooping packets.	-
vlan *vlan-id*	Enables or disables debugging of MLD snooping for a VLAN.	The value is an integer ranging from 1 to 4094.

Parameter	Description	Value
source *source-address*	Enables or disables debugging of MLD snooping for a specified multicast source address.	The value is a 32-digit hexadecimal number, in the format of X:X:X:X:X:X:X:X.
group *group-address*	Enables or disables debugging of MLD snooping for a specified multicast group address.	The value is a 32-digit hexadecimal number, in the format of X:X:X:X:X:X:X:X. The value ranges from FF00::0 to FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF.
interface *interface-type interface-number*	Enables or disables debugging of MLD snooping for a specified interface.	-
basic-acl-number	Specifies the number of a basic IPv6 ACL.	The value is an integer ranging from 2000 to 2999.
advanced-acl-number	Specifies the number of an advanced IPv6 ACL.	The value is an integer ranging from 3000 to 3999.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging mld snooping** command enables MLD snooping debugging, and the **undo debugging mld snooping** command disables MLD snooping debugging and diagnosis.

Example

# In the user view, enable debugging of all MLD snooping information.

```
<HUAWEI> debugging mld snooping all
Dec 19 2012 03:54:43.646 HUAWEI %%01SNGP/7/DGMP_DEBUG_INFO(d):CID=0x80e22770;(VRFID=0):
Create Instance Event node, VS: 0, AF:2, InstType: 0, InstId: 1, InstEvent: 0.(295)

Dec 19 2012 03:54:43.646 HUAWEI %%01SNGP/7/DGMP_DEBUG_INFO(d):CID=0x80e22770;(VRFID=0):
Remove Instance Event node, VS: 0, AF:2, InstType: 0, InstId: 1, InstEvent: 0.(649)

Dec 19 2012 03:54:43.646 HUAWEI %%01SNGP/7/DGMP_DEBUG_INFO(d):CID=0x80e22770;(VRFID=0): Set
Instance Event node, VS: 0, AF:2, InstType: 0, InstId: 1, InstEvent: 0.(1237)
```

```
Dec 19 2012 03:54:43.646 HUAWEI %%01SNGP/7/DGMP_DEBUG_INFO(d):CID=0x80e22770;(VRFID=0): Set
Instance Event node, VS: 0, AF:2, InstType: 0, InstId: 1, InstEvent: 1.(1237)
```

2.7.8 IP Multicast over VXLAN Debugging Commands

NOTE

Only the CE8868EI, CE8861EI, CE8860EI, CE8850EI, CE7855EI, CE7850EI, CE6875EI, CE6870EI, CE6865EI, CE6860EI, CE6857EI, CE6856HI, CE6855HI, CE6851HI, CE6850U-HI and CE6850HI support this feature.

2.7.8.1 debugging mvpn all

Function

The **debugging mvpn all** command enables all MVPN debuggings.

The **undo debugging mvpn all** command disables all MVPN debuggings.

By default, all MVPN debuggings are disabled.

Format

debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **all**

undo debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **all**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Indicates the debugging of a specified VPN instance. *vpn-instance-name* specifies a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Indicates the debugging of all instances, including public network instances and VPN instances.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To help locate faults occurred during the NG MVPN operation, run the **debugging mvpn all** command to enable all MVPN debuggings.

If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically.

Example

# Enable MVPN debuggings of all instances.

```
<HUAWEI> debugging mvpn all-instance all
```

2.7.8.2 debugging mvpn c-multicast

Function

The **debugging mvpn c-multicast** command enables the debugging of MVPN C-multicast routes.

The **undo debugging mvpn c-multicast** command disables the debugging of MVPN C-multicast routes.

By default, the debugging of MVPN C-multicast routes is disabled.

Format

debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **c-multicast** { **send** | **receive** }

undo debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **c-multicast** { **send** | **receive** }

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Indicates the debugging of C-multicast routes of a specified VPN instance. *vpn-instance-name* specifies a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Indicates the debugging of C-multicast routes of all instances, including public network instances and VPN instances.	-

Parameter	Description	Value
send	Indicates the debugging of sent C-multicast routes.	-
receive	Indicates the debugging of received C-multicast routes.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To help locate C-multicast route faults occurred, run the **debugging mvpn c-multicast** command to enable the debugging of MVPN C-multicast routes.

If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically. If **vpn-instance** *vpn-instance-name* or **all-instance** is not configured, this command enables the debugging of C-multicast routes of public network instances.

Example

# Enable the debugging of sent C-multicast routes of all instances.

```
<HUAWEI> debugging mvpn all-instance c-multicast send
```

2.7.8.3 debugging mvpn event

Function

The **debugging mvpn event** command enables the debugging of MVPN events.

The **undo debugging mvpn event** command disables the debugging of MVPN events.

By default, the debugging of MVPN events is disabled.

Format

debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **event**

undo debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **event**

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Indicates debugging of MVPN events of a specified VPN instance. *vpn-instance-name* specifies a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Indicates the debugging of MVPN events of all instances, including public network instances and VPN instances.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To help locate faults occurred on MVPN, run the **debugging mvpn event** command.

If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically. If **vpn-instance** *vpn-instance-name* or **all-instance** is not configured, this command enables the debugging of MVPN events of public network instances.

Example

# Enable the debugging of MVPN events.

```
<HUAWEI> debugging mvpn all-instance event
```

2.7.8.4 debugging mvpn ipmsi-ad

Function

The **debugging mvpn ipmsi-ad** command enables the debugging of MVPN I-PMSI A-D routes.

The **undo debugging mvpn ipmsi-ad** command disables the debugging of MVPN I-PMSI A-D routes.

By default, the debugging of MVPN I-PMSI A-D routes is disabled.

Format

debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **ipmsi-ad** { **send** | **receive** }

undo debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **ipmsi-ad** { **send** | **receive** }

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Indicates the debugging of I-PMSI A-D routes of a specified VPN instance. *vpn-instance-name* specifies a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Indicates the debugging of I-PMSI A-D routes of all instances, including public network instances and VPN instances.	-
send	Indicates the debugging of sent I-PMSI A-D routes.	-
receive	Indicates the debugging of received I-PMSI A-D routes.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To help locate I-PMSI A-D route faults occurred, run the **debugging mvpn ipmsi-ad** command to enable the debugging of MVPN I-PMSI A-D routes.

If the debugging of all instances is enabled, the debugging of newly added instances is enabled automatically. If **vpn-instance** *vpn-instance-name* or **all-instance** is not configured, this command enables the debugging of I-PMSI A-D routes of public network instances.

Example

# Enable the debugging of sent I-PMSI A-D routes of all instances.

```
<HUAWEI> debugging mvpn all-instance ipmsi-ad send
```

2.7.8.5 debugging mvpn source-active-ad

Function

The **debugging mvpn source-active-ad** command enables the debugging function for Source Active A-D routes on an NG MVPN network.

The **undo debugging mvpn source-active-ad** command disables the debugging function for Source Active A-D routes on an NG MVPN network.

By default, the debugging function is disabled for Source Active A-D routes on an NG MVPN network.

Format

debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **source-active-ad** [**send** | **receive**] [**source** *source-address* | **group** *group-address*] *

undo debugging mvpn { **vpn-instance** *vpn-instance-name* | **all-instance** } **source-active-ad** [**send** | **receive**]

Parameters

Parameter	Description	Value
vpn-instance *vpn-instance-name*	Enables the debugging function for Source Active A-D routes of a specified VPN instance. *vpn-instance-name* specifies the name of a VPN instance.	The value is a string of 1 to 31 case-sensitive characters except spaces. When double quotation marks are used to include the string, spaces are allowed in the string. The value _public_ is reserved and cannot be used as the VPN instance name.
all-instance	Enables the debugging function for Source Active A-D routes of all instances, including public network and VPN instances.	-
send	Enables the debugging function for Source Active A-D routes to be sent.	-

Parameter	Description	Value
receive	Enables the debugging function for received Source Active A-D routes.	-
source *source-address*	Enables the debugging function for Source Active A-D routes with a specified multicast source address.	The value is in dotted decimal notation.
group *group-address*	Enables the debugging function for Source Active A-D routes with a specified multicast group address.	The value ranges from 224.0.1.0 to 239.255.255.255, in dotted decimal notation.

Views

User view

Default Level

3: Management level

Usage Guidelines

If faults occur in Source Active A-D routes on an NG MVPN network, run the **debugging mvpn source-active-ad** command to enable the debugging function for Source Active A-D routes. The command output helps you locate faults.

If the **all-instance** parameter is specified, the debugging function will be enabled automatically for new instances. If the **vpn-instance** *vpn-instance-name* or **all-instance** parameter is not specified, the **debugging mvpn source-active-ad** command enables the debugging function for Source Active A-D routes of the public network by default.

Example

# Enable the debugging function for Source Active A-D routes to be sent in all instances.

```
<HUAWEI> debugging mvpn all-instance source-active-ad send
```

2.7.8.6 display debugging mvpn

Function

The **display debugging mvpn** command displays the NG MVPN debugging status.

Format

display debugging mvpn

Parameters

None

Views

All views

Default Level

3: Management level

Usage Guidelines

When a large amount of information is output, run the **display debugging mvpn** command to check the NG MVPN debugging functions that have been enabled. Then you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# Display the NG MVPN debugging status.

```
<HUAWEI> display debugging mvpn
MVPN(all-instance) ipmsi-ad receive debugging switch is on
MVPN(all-instance) ipmsi-ad send debugging switch is on
MVPN(all-instance) event debugging switch is on
MVPN(all-instance) c-multicast receive debugging switch is on
MVPN(all-instance) c-multicast send debugging switch is on
```

2.8 Security Debugging Commands

2.8.1 AAA Debugging Commands

2.8.1.1 debugging hwtacacs

Function

Using the **debugging hwtacacs** command, you can enable the system debugging function of HWTACACS.

Using the **undo debugging hwtacacs** command, you can disable the system debugging function of HWTACACS.

By default, the debugging of HWTACACS is disabled.

Format

debugging hwtacacs { **all** | **error** | **event** | **message** | **receive-packet** | **send-packet** }

undo debugging hwtacacs { **all** | **error** | **event** | **message** | **receive-packet** | **send-packet** }

Parameters

Parameter	Description	Value
all	Enables or disables debugging of all current modules.	-
error	Indicates debugging of error function.	-
event	Indicates debugging of event function.	-
message	Indicates debugging of message function.	-
receive-packet	Indicates debugging of received packet function.	-
send-packet	Indicates debugging of sent packet function.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

When a HWTACACS module becomes faulty, the network administrator cannot use HWTACACS to get AAA services (Authentication, Authorization and Accounting) on the remote device. You can run this command to start the debugging information on the HWTACACS module and rapidly locate faults based on the obtained information.

Example

# Enable HWTACACS debugging information on a console.

```
<HUAWEI> debugging hwtacacs aaa
```

2.8.1.2 debugging radius

Function

Using the **debugging radius** command, you can enable the system debugging function of RADIUS.

Using the **undo debugging radius** command, you can disable the system debugging function of RADIUS.

By default, the debugging of RADIUS connection is disabled.

Format

debugging radius { **acct-packet** | **acct-verbose** | **all** | **auth-packet** | **auth-server** | **auth-verbose miscellaneous** | **packet** }

undo debugging radius { **acct-packet** | **acct-verbose** | **all** | **auth-packet** | **auth-server** | **auth-verbose** | **miscellaneous** | **packet** }

Parameters

Parameter	Description	Value
acct-packet	Indicates the debugging for RADIUS accounting packets.	-
acct-verbose	Indicates the debugging for RADIUS accounting packets which includes hex dump of packets.	-
all	Enables or disables debugging of all current modules.	-
auth-packet	Indicates the debugging for RADIUS authentication packets.	-
auth-server	Indicates the debugging for RADIUS server probe packets and server state changes.	-
auth-verbose	Indicates the debugging for RADIUS authentication packets that include hex dump of packets.	-
miscellaneous	Indicates the debugging for license and other miscellaneous information.	-
packet	Indicates the debugging for RADIUS packets.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

When a RADIUS module becomes faulty, the network administrator cannot perform local management using RADIUS to get AAA services (Authentication, Authorization and Accounting) on the remote device. You can run this command to start the debugging information on the RADIUS module and rapidly locate faults based on the obtained information.

Example

# Enable RADIUS debugging information on a console.

```
<HUAWEI> debugging radius all
```

2.8.2 ACL Debugging Commands

2.8.2.1 debugging acl

Function

The **debugging acl** command enables ACL debugging for quickly locating faults.

The **undo debugging acl** command disables ACL debugging.

By default, the debugging of ACL is disabled.

Format

debugging acl [**ipv6**] **match-info** { **number** *acl-number* | **name** *acl-name* } [**max-count** *dbg-count*]

undo debugging acl [**ipv6**] **match-info** [**number** *acl-number* | **name** *acl-name*]

Parameters

Parameter	Description	Value
ipv6	Enables the debugging for ACL6.	-
match-info	Debugs matching information about an ACL.	-
number *acl-number*	Debugs the ACL with the specified number.	The value is an integer. ● If **ipv6** is configured, an ACL number ranges from 2000 to 3999. – A basic ACL6 number ranges from 2000 to 2999. – An advanced ACL6 number ranges from 3000 to 3999. ● If **ipv6** is not configured, an ACL number ranges from 2000 to 5999. – A basic ACL number ranges from 2000 to 2999. – An advanced ACL number ranges from 3000 to 3999. – A Layer 2 ACL number ranges from 4000 to 4999. – A user-defined ACL number ranges from 5000 to 5999.

Parameter	Description	Value
name *acl-name*	Debugs the ACL with the specified name.	The value is a string of 1 to 32 case-sensitive characters except spaces. The value must start with a letter or digit, and cannot contain only digits.
max-count *dbg-count*	Specifies the maximum number of debugging information outputs.	The value is an integer ranging from 0 to 4294967295.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging acl** command enables ACL debugging and outputs debugging information for locating problems.

Example

# Enable ACL debugging of matching information about ACL rules.

```
<HUAWEI> debugging acl match-info number 2000
Jun 25 2012 04:54:29.364 HUAWEI %%01ACL/7/DEBUG(d):VS=Admin-VS-CID=0x80782754;[LDM match
info:SrcIP(a7508a0a).]

Jun 25 2012 04:54:29.364 HUAWEI %%01ACL/7/DEBUG(d):VS=Admin-VS-CID=0x80782754;[LDM match
rule:RuleID=5, Priority=5, Status=Active, ConditionNum=2, Result=ACL_DENY.]

Jun 25 2012 04:54:29.364 HUAWEI %%01ACL/7/DEBUG(d):VS=Admin-VS-CID=0x80782754;[LDM match
result:uiRet=ACL_DENY, ruleID=5, groupId=1, vsid=0, VpnIndex=0.]
```

2.8.3 DHCP Snooping Debugging Commands

2.8.3.1 debugging dhcp snooping

Function

The **debugging dhcp snooping** command enables DHCP snooping debugging functions.

The **undo debugging dhcp snooping** command disables DHCP snooping debugging functions.

By default, the DHCP snooping debugging functions are disabled.

Format

debugging dhcp snooping packet [**mac-address** *mac-address* | **ip-address** *ip-address* | **interface** *interface-type interface-number*]

undo debugging dhcp snooping { **all** | **error** | **event** | **packet** }

Parameters

Parameter	Description	Value
all	Enables all DHCP snooping debugging functions, including the error debugging, event debugging, and packet debugging.	-
error	Enables the debugging of errors.	-
event	Enables the debugging of events.	-
packet	Enables the debugging of packets.	-
vlan *vlan-id*	Specifies a VLAN ID. If this parameter is specified, the device displays the VLAN ID related debugging information.	The value is an integer ranging from 1 to 4094.
mac-address *mac-address*	Specifies a MAC address. If this parameter is specified, the device displays the MAC address related debugging information.	The value is in the format of H-H-H. Each H stands for one to four hexadecimal digits.
ip-address *ip-address*	Specifies an IP address. If this parameter is specified, the device displays the IP address related debugging information.	The value is in dotted decimal notation.
interface *interface-type interface-number*	Specifies the type and number of an interface. If this parameter is specified, the device displays the interface related debugging information.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

If you want to learn DHCP debugging information, run the **debugging dhcp snooping** command to enable the DHCP snooping debugging functions. All DHCP snooping debugging functions include the error debugging, event debugging, and packet debugging. You can enable specified debugging functions based on MAC addresses, IP addresses, and interfaces.

Example

# Enable all DHCP snooping debugging functions. The processing information about DHCP packets is shown as follows:

```
<HUAWEI> debugging dhcp snooping all
Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
DHCP Snooping process.
DHCP REQUEST, MSGTYPE: DISCOVER, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Receive packet before EUM process.
DHCP REQUEST, MSGTYPE: DISCOVER, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
--- Process dhcp discover ---
DHCP REQUEST, MSGTYPE: DISCOVER, Chaddr: 0001-0101-0101, on IfIndex 8

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
DHCP Snooping protocol process ends with pause.

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
DHCP Snooping process.
DHCP REQUEST, MSGTYPE: DISCOVER, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Receive packet after EUM process.
DHCP REQUEST, MSGTYPE: DISCOVER, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Send packet by trust port.
DHCP REQUEST, MSGTYPE: DISCOVER, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
DHCP Snooping process.
DHCP REPLY, MSGTYPE: OFFER, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Receive packet before EUM process.
DHCP REPLY, MSGTYPE: OFFER, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
--- Process dhcp offer ---
DHCP REPLY, MSGTYPE: OFFER, Chaddr: 0001-0101-0101, on IfIndex 17

Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Send packet by L2If port.
DHCP REPLY, MSGTYPE: OFFER, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000
```

```
Dec  4 2012 17:01:58.326 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
DHCP Snooping protocol process ends with stop.

Dec  4 2012 17:01:58.336 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
DHCP Snooping process.
DHCP REQUEST, MSGTYPE: REQUEST, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.336 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Receive packet before EUM process.
DHCP REQUEST, MSGTYPE: REQUEST, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.336 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
--- Process dhcp request ---
DHCP REQUEST, MSGTYPE: REQUEST, Chaddr: 0001-0101-0101, on IfIndex 8

Dec  4 2012 17:01:58.336 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
Request: Get tmp user info successfully.
DHCP REQUEST, MSGTYPE: REQUEST, Chaddr: 0001-0101-0101, on IfIndex 8

Dec  4 2012 17:01:58.336 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Send packet by trust port.
DHCP REQUEST, MSGTYPE: REQUEST, Chaddr: 0001-0101-0101, on IfIndex 8
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
DHCP Snooping process.
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Receive packet before EUM process.
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
--- Process dhcp ack ---
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
Ack: Update local user info successfully.
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_EVENT(d):CID=0x8053042e;
DHCP Snooping protocol process ends with pause.

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
DHCP Snooping process.
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Receive packet after EUM process.
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000

Dec  4 2012 17:01:58.376 HUAWEI %%01DHCP/7/DHCPSNP_DBG_PACKET(d):CID=0x8053042e;
Send packet by L2If port.
DHCP REPLY, MSGTYPE: ACK, Chaddr: 0001-0101-0101, on IfIndex 17
```

```
Ciaddr: 0.0.0.0, Yiaddr: 10.177.248.238, Giaddr: 0.0.0.0, Siaddr: 0.0.0.0
Htype: 1, Hlen: 6, Hops: 0, Xid: 0x0, Flag: 0x8000
```

2.8.4 IPSec Debugging Commands

NOTE

CE6810LI does not support this command.

2.8.4.1 debugging packet ipsec

Function

The **debugging packet ipsec** command enables debugging of outgoing and incoming IPSec packets.

The **undo debugging packet ipsec** command disables debugging of outgoing and incoming IPSec packets.

By default, the packet IPSec debugging disables.

Format

debugging packet ipsec { **ah** | **esp** } [**verbose**]

undo debugging packet ipsec { **ah** | **esp** } [**verbose**]

Parameters

Parameter	Description	Value
ah	Displays debugging information for AH packets.	-
esp	Displays debugging information for ESP packets.	-
verbose	Displays debugging information in detail.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ipsec	debug

Usage Guidelines

Usage Scenario

When IP Security is applied on the application (ex: OSPFv3 or RIPng), all incoming and outgoing packets will be authenticated. Generally, this command can be used to debug the IPsec packets.

Precautions

Debugging information is displayed on the screen. Do not output too much information for purposes other than debugging so that the system is not affected.

Example

# Enable debugging of IPSec packets.

```
<HUAWEI> debugging packet ipsec esp
LDM:
---------------------------------------------
My Cid          : 0x80782742
Peer Cid        : 0x6503F2
VS              : 0
Handle          : 3
TraceNum        : 1
Direction       : Up
Status          : 0
Interface index : 6
Link type       : ETH
Source mac      : 00 e0 48 06 81 42
Dest mac        : 33 33 00 00 00 05
Link protocol   : 0x86dd
Protocol        : IPV6
Time            : 2011-10-14 13:25:32 81
---------------------------------------------

LDM:
---------------------------------------------
My Cid          : 0x80782742
Peer Cid        : 0x806503F8
VS              : 0
Handle          : 3
TraceNum        : 8
Direction       : Down
Status          : 0
Interface index : 6
Link type       : -
Protocol        : IPV6
Time            : 2011-10-14 13:25:35 451
---------------------------------------------

LDM:
---------------------------------------------
My Cid          : 0x80782742
Peer Cid        : 0x80273C
VS              : 0
Handle          : 3
TraceNum        : 8
Direction       : Down
Status          : 0
Interface index : 6
Link type       : ETH
Source mac      : 38 00 10 03 00 02
Dest mac        : 33 33 00 00 00 05
Link protocol   : 0x86dd
Protocol        : IPV6
Time            : 2011-10-14 13:25:35 451
---------------------------------------------

SOCKET: -
---------------------------------------------
```

```
My Cid          : 0x806503f8
Peer Cid        : 0x80782742
VS             : 0
Handle          : 3
TraceNum        : 1
Direction       : Up
Status          : 0
Data            :
-----------------------------------------------

SOCKET: -
-----------------------------------------------
My Cid          : 0x806503f8
Peer Cid        : 0x803f041a
VS             : 0
Handle          : 3
TraceNum        : 1
Direction       : Up
Status          : 0
Data            :
-----------------------------------------------

IPSEC:
-----------------------------------------------
My Cid            : 0x803F041A
Peer Cid          : 0x806503F8
VS               : 0
Handle            : 3
TraceNum          : 8
Direction         : Up
Status            : 0
IP Packet Version  : 6
Source Addr        : fe8000000000000002e048fffe068142
Destination Addr   : ff020000000000000000000000000005
Packet length      : 64
Protocol          : ESP
SpiIndex          : 300
Time              : 2011-10-14 13:25:32 96
-----------------------------------------------

IPSEC:
-----------------------------------------------
My Cid            : 0x803F041A
Peer Cid          : 0x722714
VS               : 0
Handle            : 3
TraceNum          : 8
Direction         : Up
Status            : 0
IP Packet Version  : 6
Source Addr        : fe8000000000000002e048fffe068142
Destination Addr   : ff020000000000000000000000000005
Packet length      : 40
Protocol          : OSPF
Time              : 2011-10-14 13:25:32 96
-----------------------------------------------

PP6:
-----------------------------------------------
My Cid          : 0x80722719
Peer Cid        : 0x803F041A
VS             : 0
Handle          : 3
TraceNum        : 8
Direction       : Up
Status          : 0
BlockNo         : 0
Time            : 2011-10-14 13:25:32 96
-----------------------------------------------
```

```
IPSEC:
---------------------------------------------
My Cid            : 0x803F041A
Peer Cid          : 0x806503F8
VS                : 0
Handle            : 3
TraceNum          : 8
Direction         : Down
Status            : 0
IP Packet Version : 6
Source Addr       : fe800000000000003a0010fffe030002
Destination Addr  : ff020000000000000000000000000005
Packet length     : 40
Protocol          : OSPF
Time              : 2011-10-14 13:25:35 466
---------------------------------------------

IPSEC:
---------------------------------------------
My Cid            : 0x803F041A
Peer Cid          : 0x806503F8
VS                : 0
Handle            : 3
TraceNum          : 8
Direction         : Down
Status            : 0
IP Packet Version : 6
Source Addr       : fe800000000000003a0010fffe030002
Destination Addr  : ff020000000000000000000000000005
Packet length     : 64
Protocol          : ESP
SpiIndex          : 300
Time              : 2011-10-14 13:25:35 466
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid         : 0x806503f8
Peer Cid       : 0x803f041a
VS             : 0
Handle         : 3
TraceNum       : 8
Direction      : Down
Status         : 0
Data           :
---------------------------------------------

SOCKET: -
---------------------------------------------
My Cid         : 0x806503f8
Peer Cid       : 0x782737
VS             : 0
Handle         : 3
TraceNum       : 8
Direction      : Down
Status         : 0
Data           :
---------------------------------------------
```

Table 2-43 Description of the **debugging packet ipsec** command output

Item	Description
My Cid	Self component id
Peer Cid	Peer component id

Item	Description
VS	Virtual router number
Handle	Handle value
TraceNum	Trace id
Direction	Direction of packet flow : ● Up: Inbound packet that received from neighbor ● Down: Outbound packet that need to Send to neighbor
Status	Status of the packet processing
Interface index	Interface index
Link type	Link type
Source mac	Source address of the packet
Dest mac	Destination address of the packet
Link protocol	Protocol ID
Protocol	Protocol in use
Time	Current system time
IP Packet Version	IP Packe Version, it can be IPv4 or IPv6
Source Addr	Source address of the packet
Destination Addr	Destination address of the packet
Packet length	Length of the packet
SpiIndex	Index of spi
Data	Authenticated data
BlockNo	Number of blocked packets

2.8.5 MACsec Debugging Commands

NOTE

Only the CE6875EI supportes MACsec.

2.8.5.1 debugging mka

Function

The **debugging mka** command enables MKA debugging.

The **undo debugging mka** command disables MKA debugging.

By default, MKA debugging is disabled.

Format

debugging mka module { **all** | **protocol** | **timer** | **smp** | **ssp** } **level** { **error** | **info** | **packet** } [**interface** *interface-type interface-number*]

undo debugging mka module { **all** | **protocol** | **timer** | **smp** | **ssp** } **level** { **error** | **info** | **packet** } [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
all	Enables all MACsec modules.	-
protocol	Enables the MACsec protocol module.	-
timer	Enables the MACsec timer module.	-
ssp	Enables the MACsec SSP module.	-
smp	Enables the MACsec SMP module.	-
level	Specifies the MKA information level.	-
error	Enables debugging for MKA errors.	-
info	Enables debugging for MKA information.	-
packet	Enables debugging for MKA packets.	-
interface *interface-type interface-number*	Specifies the type and number of an interface. If this parameter is not specified, debugging is enabled on all interfaces.	-

Views

Diagnostic view

Default Level

3: Management level

Usage Guidelines

You can run this command to enable MKA debugging on all interfaces or the specified interface.

Example

# Enable debugging of MKA errors for all MACsec modules on 100GE1/0/1.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging mka module all level error interface 100ge 1/0/1
```

2.9 Reliability Debugging Commands

2.9.1 BFD Debugging Commands

2.9.1.1 debugging bfd

Function

The **debugging bfd** command enables BFD debugging functions and outputs debugging information.

The **undo debugging bfd** command disables BFD debugging functions.

Format

debugging bfd { **all** | **packet** | **event** | **session-management** | **error** | **process** | **fsm** | **ha** | **timer** | **sock** | **message-list** }

undo debugging bfd { **all** | **packet** | **event** | **session-management** | **error** | **process** | **fsm** | **ha** | **timer** | **sock** | **message-list** }

Parameters

Parameter	Description	Value
all	Enables all BFD debugging functions.	-
packet	Enables debugging functions of BFD packets.	-
event	Enables event debugging of a BFD interface.	-
session-management	Enables debugging of BFD session control and management.	-
error	Enables debugging of BFD errors.	-

Parameter	Description	Value
process	Enables debugging of BFD processes.	-
fsm	Enables debugging of BFD FSM.	-
ha	Enables debugging of BFD HA.	-
timer	Enables debugging of BFD timers.	-
sock	Enables debugging of the BFD socket SOCK.	-
message-list	Enables debugging of the BFD message list.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging bfd** command enables BFD debugging functions and outputs debugging information.

Example

# Enable debugging of BFD packets.

```
<HUAWEI> debugging bfd packet
2012-04-13 03:55:17 HUAWEI %%01BFD/7/bfd_proctrack(d):CID=7612204;BFD is tra
cked.(
[04-13 03:55:15:821][BFD0]:
[BFD_PKT_SEND]: MD(1)  YD(2)  Diag(0)  State(1)  P(0)  F(0)  C(0)  A(0)  D(0)  M
(0)
DetctMult(50)  Length(24)  TX(2003000)  RX(2003000)  echo_RX(0)  Ver(1)

[04-13 03:55:15:821][BFD0]:call SOCK_SendMbufEx OK!
[04-13 03:55:17:921][BFD0]:
[BFD_PKT_SEND]: MD(1)  YD(2)  Diag(0)  State(1)  P(0)  F(0)  C(0)  A(0)  D(0)  M
(0)
DetctMult(50)  Length(24)  TX(2003000)  RX(2003000)  echo_RX(0)  Ver(1)
)
```

The source end sends a BFD negotiation packet to the remote end. The local ID of the packet is 1 and the peer ID of the packet is 2. There is no diagnosis word. The session at the local end is Down. The session does not require connection acknowledgment or return of a connection acknowledgment packet. BFD packets are processed on the control plane, and they are not used for authentication. BFD Packets do not work in query mode. The detection multiple of sessions is 50, and a packet is 24-byte long. The minimum sending interval is 2003000 microseconds, and the minimum receiving interval is 2003000 microseconds. The echo receiving interval is 0, and the protocol version is 1.

The following table describes the fields in the preceding debugging information:

Field	Description
MD	Local ID of a session
YD	Peer ID of a session
Diag	Reason for last session down: ● 0: No Diagnostic ● 1: Control Detection Time Expired ● 2: Echo Function Failed ● 3: Neighbor Signaled Session Down ● 4: Forwarding Plane Reset ● 5: Path Down ● 6: Concatenated Path Down ● 7: Administratively Down ● 8: Reverse Concatenated Path Down
State	Session status: ● 0: AdminDown ● 1: Down ● 2: Init ● 3: Up
P	Connection acknowledgment or parameter change acknowledgement
F	The **F** field must be set to **1** in response to the packet in which the **P** field is **1**.
C	Whether packets are processed on the control plane
A	Whether the authentication function is provided
D	Work in query mode or not
M	Reserved field. The value is **0**.
DetctMult	Detection multiple
Length	Packet length (unit: byte)
TX	Minimum packet sending interval supported by the local end (unit: microsecond)
RX	Minimum packet receiving interval supported by the local end (unit: microsecond)

Field	Description
echo_RX	Minimum echo packet receiving interval supported by the local end (unit: microsecond)
Ver	Version of a BFD packet

2.9.1.2 display debugging bfd

Function

The **display debugging bfd** command displays the status of BFD debugging functions.

Format

display debugging bfd

Parameters

None

Views

All views

Default Level

3: Management level

Usage Guidelines

To view whether the BFD debugging functions are enabled, run the **display debugging bfd** command.

Example

# Display the status of BFD debugging functions.

```
<HUAWEI> display debugging bfd
 BFD event debug switch is on
 BFD packet debug switch is on
 BFD SOCK debug switch is on
 BFD message list debug switch is on
 BFD error debug switch is on
 BFD session cotrol management debug switch is on
 BFD FSM debug switch is on
 BFD HA debug switch is on
 BFD timer debug switch is on
 BFD process debug switch is on
```

Table 2-44 Description of the display debugging bfd command output

Item	Description
BFD event debug switch is on	BFD event debugging was enabled.
BFD packet debug switch is on	BFD packet debugging was enabled.
BFD SOCK debug switch is on	BFD socket debugging was enabled.
BFD message list debug switch is on	BFD message queue debugging was enabled.
BFD error debug switch is on	BFD error debugging was enabled.
BFD session control management debug switch is on	The debugging of BFD session control management was enabled.
BFD FSM debug switch is on	BFD FSM debugging was enabled.
BFD HA debug switch is on	BFD HA debugging was enabled.
BFD timer debug switch is on	BFD timer debugging was enabled.
BFD process debug switch is on	The debugging of BFD processing was enabled.

2.9.2 VRRP Debugging Commands

2.9.2.1 debugging vrrp

Function

The **debugging vrrp** command enables all debugging functions of VRRP and outputs debugging information.

The **undo debugging vrrp** command disables all debugging functions of VRRP.

By default, all debugging functions of VRRP are disabled.

Format

debugging vrrp { **state** | **packet** | **timer** } [**interface** *interface-type interface-number* [**vrid** *virtual-router-id*]]

undo debugging vrrp { **state** | **packet** | **timer** } [**interface** *interface-type interface-number* [**vrid** *virtual-router-id*]]

undo debugging vrrp all

Parameters

Parameter	Description	Value
state	Enables the VRRP status debugging function.	-
packet	Enables the VRRP packet debugging function.	-
timer	Enables the VRRP timer debugging function.	-
interface *interface-type interface-number*	Specifies the type and number of an interface on which a VRRP backup group is configured.	-
vrid *virtual-router-id*	Specifies the number of a VRRP backup group.	The value is an integer ranging from 1 to 255.
all	Disables all VRRP debugging functions.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

- If the master device and backup device become abnormal during switchover, the **debugging vrrp packet** command enables debugging of VRRP packets and displays detailed information about sent and received VRRP packets. This command can be run to locate the fault.
- The **undo debugging vrrp packet** command disables debugging of VRRP packets.
- When status changeover of the VRRP backup group becomes abnormal, the **debugging vrrp state** command enables status debugging of VRRP and displays the status changeover information about the VRRP backup group. This command can be run to locate the fault.
- The **undo debugging vrrp state** command disables status debugging of VRRP.
- When status changeover of the VRRP backup group becomes abnormal, the **debugging vrrp timer** command enables timer debugging of VRRP and displays the timer timeout information about the VRRP backup group. This command can be run to locate the fault.

NOTE

- When the status of the VRRP backup group is master, the timeout information about the ADVER_INTERNAL timer is displayed.
- When the status of the VRRP backup group is backup, the timeout information about the MASTER_DOWN timer is displayed.

- The **undo debugging vrrp timer** command disables debugging of VRRP timers.

Example

# Enable debugging of VRRP packets.

```
<HUAWEI> debugging vrrp packet
Feb 25 2014 10:36:01.369 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_PACKET(d):CID=2149918550; IfIndex: 5 |
Virtual Router 1 | InetType IPv4 : sending from 192.168.17.2, version = 2, priority = 100, timer = 1000 ms,
auth type is no.
```

# The IP address of the Ethernet interface whose index is 5 is 1.1.1.1 and VRRP backup group 1 is configured on the interface. VRRP backup group 1 sends a notification packet whose priority is 100 and version number is 2 every 1s. The packet does not need to be authenticated.

```
Feb 25 2014 11:36:01.369 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_PACKET(d):CID=2149918550; IfIndex: 5 |
Virtual Router 2 | InetType IPv4 : receiving from 192.168.18.2, version = 2, priority = 100, timer = 1000 ms,
auth type is no.
```

2.9.2.2 debugging vrrp event

Function

The **debugging vrrp event** command enables debugging of information about interaction between the VRRP module and external modules.

The **undo debugging vrrp event** command disables debugging of information about interaction between the VRRP module and external modules.

By default, debugging of information about interaction between the VRRP module and external modules is disabled.

Format

debugging vrrp event

undo debugging vrrp event

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging vrrp event** command enables debugging of information about interaction between the VRRP module and external modules so that you can analyze and locate faults.

Example

# Enable the debugging function for the exchange between VRRP and external modules.

```
<HUAWEI> debugging vrrp event
Jun 25 2014 08:42:07.073 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP
Notify ARP to send gratuitous arp packets due to VRRP change to master event.

Jun 25 2014 08:42:07.073 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP
Notify ARP to send gratuitous arp packets due to GRA/NA timer expired event.

Jun 25 2014 08:42:07.073 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP
Notify ARP to send gratuitous arp packets due to LBRG VRRP delete event.

Jun 25 2014 08:42:07.073 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP
Notify ARP to send gratuitous arp packets due to LBRG-MEMBER VRRP delete event.
```

2.9.2.3 debugging vrrp6

Function

The **debugging vrrp6** command enables VRRP6 debugging functions and displays debugging information.

The **undo debugging vrrp6** command disables VRRP6 debugging functions.

By default, VRRP6 debugging functions are disabled.

Format

debugging vrrp6 { **state** | **packet** | **timer** } [**interface** *interface-type interface-number* [**vrid** *virtual-router-id*]]

undo debugging vrrp6 { **state** | **packet** | **timer** } [**interface** *interface-type interface-number* [**vrid** *virtual-router-id*]]

undo debugging vrrp6 all

Parameters

Parameter	Description	Value
state	Enables the VRRP6 status debugging function.	-
packet	Enables the VRRP6 packet debugging function.	-
timer	Enables the VRRP6 timer debugging function.	-

Parameter	Description	Value
interface-type interface-number	Specifies the type and number of an interface on which a VRRP6 backup group is configured.	-
vrid *virtual-router-id*	Specifies the ID of a VRRP6 backup group.	The value is an integer ranging from 1 to 255.
all	Disables all VRRP6 debugging functions.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

If an exception occurs during a master/backup VRRP6 switchover:

- Run the **debugging vrrp6 packet** command to enable the VRRP6 packet debugging function and display detailed information about sent and received VRRP6 packets.
- To disable the VRRP6 packet debugging function, run the **undo debugging vrrp6 packet** command.
- Run the **debugging vrrp6 state** command to enable the VRRP6 status debugging function and display VRRP6 status change information.
- To disable the VRRP6 status debugging function, run the **undo debugging vrrp6 state** command.
- Run the **debugging vrrp6 timer** command to enable the VRRP6 timer debugging function and display timer expiration information.

  NOTE

  - If the status of the VRRP6 backup group is Master, ADVER_INTERVAL timer expiration information is displayed.
  - If the status of the VRRP6 backup group is Backup, MASTER_DOWN timer expiration information is displayed.
- To disable the VRRP6 timer debugging function, run the **undo debugging vrrp6 timer** command.

You can use the information to find out the reason of the exception.

Example

# Enable the VRRP6 packet debugging function.

```
<HUAWEI> debugging vrrp6 packet
Aug 28 2013 08:53:15.354 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_PACKET(d):CID=0x80bc04b0; IfIndex:
108 | Virtual Router 5 | InetType IPv6 : receiving from FE80::225:9EFF:FE01:203, version = 3, priority = 120,
timer = 1000 ms, auth type is no.
```

The command output shows the following information:

- VRRP6 backup group 5 was configured on the interface with an index of 108.
- The backup group received a VRRP6 Advertisement packet carrying a version number of 3, a priority of 120, an interval of 1000 milliseconds, and an authentication type of no.

```
Aug 28 2013 08:53:15.854 HUAWEI %%01VRRP/7/VRRP_DEBUG_ID_PACKET(d):CID=0x80bc04b0; IfIndex:
108 | Virtual Router 4 | InetType IPv6 : sending from FE80::219:74FF:FE59:3305, version = 3, priority = 120,
timer = 1200 ms, auth type is no.
```

The command output shows the following information:

- VRRP6 backup group 4 was configured on the interface with an index of 108.
- The backup group sent a VRRP6 Advertisement packet carrying a priority of 120, an interval of 1200 milliseconds, and an authentication type of no.

2.9.2.4 debugging vrrp6 event

Function

The **debugging vrrp6 event** command enables the debugging function for the exchange between VRRP6 and external modules.

The **undo debugging vrrp6 event** command disables the debugging function for the exchange between VRRP6 and external modules.

By default, the debugging function is disabled for the exchange between VRRP6 and external modules.

Format

debugging vrrp6 event

undo debugging vrrp6 event

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable the debugging function for the exchange between VRRP6 and external modules for fault analysis and locating, run the **debugging vrrp6 event** command.

Example

# Enable the debugging function for the exchange between VRRP6 and external modules.

```
<HUAWEI> debugging vrrp6 event
Jun 25 2014 09:41:36.187 HUAWEI %%01VRRP6/7/VRRP6_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP6
Notify ND to send NA packets due to VRRP change to master event.

Jun 25 2014 09:41:36.187 HUAWEI %%01VRRP6/7/VRRP6_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP6
Notify ND to send NA packets due to GRA/NA timer expired event.

Jun 25 2014 09:41:36.187 HUAWEI %%01VRRP6/7/VRRP6_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP6
Notify ND to send NA packets due to LBRG VRRP delete event.

Jun 25 2014 09:41:36.187 HUAWEI %%01VRRP6/7/VRRP6_DEBUG_ID_NOTIFY(d):CID=0x80b00419; VRRP6
Notify ND to send NA packets LBRG-MEMBER VRRP delete event.
```

2.9.3 DLDP Debugging Commands

2.9.3.1 debugging dldp

Function

The **debugging dldp** command enables DLDP debugging functions and displays debugging information.

The **undo debugging dldp** command disables DLDP debugging functions.

By default, DLDP debugging functions are disabled.

Format

debugging dldp { **packet** | **state** | **packet** { **receive** | **send** } } [**interface** *interface-type interface-number*]

debugging dldp { **all** | **error** }

undo debugging dldp { **all** | **error** }

undo debugging dldp { **packet** | **state** | **packet** { **receive** | **send** } } [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
all	Enables all debugging functions.	-
error	Enables the debugging functions for error packets or messages.	-
packet	Enables the debugging functions for sent and received DLDPDUs.	-
receive	Enables the debugging functions for received DLDPDUs.	-

Parameter	Description	Value
send	Enables the debugging functions for sent DLDPDUs.	-
state	Enables the debugging functions for the DLDP status.	-
interface-type interface-number	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

When the DLDP status fails to be negotiated or is abnormal, run the **debugging dldp** command to enable all DLDP debugging functions for fault locating.

Example

# Enable all DLDP debugging functions.

```
<HUAWEI>debugging dldp all
Mar  5 2013 14:14:41.726 HUAWEI %%01DLDP/7/DLDP_DEBUG_PACKET_SEND(d):CID=0x80a4048f;
IfIndex(0x2b) send packet, pdu type[advertisement], pdu operation code[0x0], data:
00010101    00000000    00000000    00000000    00000000
00000005    0000000A    0B0C0D01    002B0000    00000000
00000000    00000000    00000000    00000000    00000000
00000000    0000
```

Table 2-45 Description of the **debugging dldp** command output

Item	Description
IfIndex	Index of the interface
send packet	Flag indicating that DLDPDUs have been sent
pdu type	Type of the DLDPDUs
pdu operation code	Operation code of the DLDPDUs
data	Data at the transmit end

2.9.4 Smart Link Debugging Commands

2.9.4.1 debugging smart-link

Function

The **debugging smart-link** command enables all the debugging of the Smart Link module.

The **undo debugging smart-link** command disables all the debugging of the Smart Link module.

By default, all the debugging of the Smart Link module is disabled.

Format

debugging smart-link

undo debugging smart-link

debugging smart-link { **all** | **error** | **event** | **fsm-machine** } [**group** *group-id*]

undo debugging smart-link { **all** | **error** | **event** | **fsm-machine** } [**group** *group-id*]

debugging smart-link flush [**all** | **receive** | **send**]

undo debugging smart-link flush [**all** | **receive** | **send**]

Parameters

Parameter	Description	Value
all	Indicates all the debugging.	-
error	Indicates the debugging of errors.	-
event	Indicates the debugging of events.	-
fsm-machine	Indicates the debugging of the state machine.	-
group *group-id*	Specifies the ID of a Smart Link group.	The value is an integer ranging from 1 to 48.
flush	Indicates the debugging of Flush packets.	-
receive	Indicates the debugging of the reception of Flush packets.	-
send	Indicates the debugging of the transmission of Flush packets.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable all the debugging of the Smart Link module, run the **debugging smart-link** command. This helps locate faults and maintain the device.

Example

# Enable all the debugging of the Smart Link module and display all the debugging information.

```
<HUAWEI> debugging smart-link
Apr 24 2013 12:38:57.313 HUAWEI %%01SMLK/7/SMLK_DEBUG_FSM(d):CID=0x80b1273e; group 1,Refresh
PI information successful
Apr 24 2013 12:38:57.313 HUAWEI %%01SMLK/7/SMLK_DEBUG_VR_FSM(d):CID=0x80b1273e;Process
customer subscribe event, event = 2
```

2.9.5 EFM Debugging Commands

2.9.5.1 debugging efm

Function

The **debugging efm** command enables all debugging EFM functions and outputs debugging information.

The **undo debugging efm** command disables all EFM debugging functions.

By default, all efm debugging functions are disabled.

Format

debugging efm { **all** | **error** | **message** | **packet** { **all** | **receive** | **send** } | **state** } [**interface** *interface-type interface-num*]

undo debugging efm { **all** | **error** | **message** | **packet** { **all** | **receive** | **send** } | **state** } [**interface** *interface-type interface-num*]

Parameters

Parameter	Description	Value
all	Enables all debugging functions.	-
error	Enables debugging of error packets or error messages.	-

Parameter	Description	Value
message	Enables debugging of message exchanged between components.	-
packet	Enables debugging of packets sent and received by the EFM.	-
receive	Enables debugging of packets received by the EFM.	-
send	Enables debugging of packets sent by the EFM.	-
state	Enables debugging of the EFM state.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

- When an EFM session fails to be negotiated, the **debugging efm packet** command displays the sent and received packets and checks whether packet sending and receiving are normal and whether the packet format is correct and the **debugging efm state** command displays changes of the state machine and checks whether stateful switchover is normal.
- When the EFM fails to interwork with other components, the **debugging efm message** command displays the messages exchanged between the EFM and other components and checks whether message sending and receiving of the components are normal.
- When the EFM component becomes abnormal, the **debugging efm error** command checks whether an error occurs in packet or message processing.

Example

# Enable EFM debugging of packets.

```
<HUAWEI> debugging efm packet all
```

# Enable EFM debugging of message.

```
<HUAWEI> debugging efm message
```

# Enable EFM debugging of state.

```
<HUAWEI> debugging efm state
```

# Enable EFM debugging of error.

```
<HUAWEI> debugging efm error
```

2.9.6 CFM Debugging Command

2.9.6.1 debugging eoam-y1731

Function

The **debugging eoam-y1731** command enables a specified Y.1731 debugging function or all Y.1731 debugging functions and outputs debug information.

The **undo debugging eoam-y1731** command disables the Y.1731 debugging functions.

Format

debugging eoam-y1731 { **error** | **all** | **info** | **debug** | **warning** }

undo debugging eoam-y1731 { **error** | **all** | **info** | **debug** | **warning** }

Parameters

Parameter	Description	Value
all	Enables all debugging functions.	-
debug	Enables the debugging function of detailed information in the processing procedure.	-
error	Enables the debugging function in the case when an error occurs.	-
info	Enables the debugging function of key information in the processing procedure.	-
warning	Enables the debugging function of insufficient resources.	-

Views

User view

Default Level

3: Monitoring level

Usage Guidelines

To enable a specified Y.1731 debugging function or all Y.1731 debugging functions, run the **debugging eoam-y1731** command. The output debug information helps locate faults in the Y.1731 module.

Example

# Enable all Y.1731 debugging functions.

```
<Switch> debugging eoam-y1731 all
Jan  1 2010 10:09:09.236 HUAWEI %%01Y1731/7/Y1731_DEBUG_DEBUG(d):CID=0x80a70423;
```

```
[a9f8d6f7.433] # Event happen(94)

Jan  1 2010 10:09:09.236 HUAWEI %%01Y1731/7/Y1731_DEBUG_INFO(d):CID=0x80a70423;
[a9f96940.1011] # cailiang:Result update(0)

Jan  1 2010 10:09:09.236 HUAWEI %%01Y1731/7/Y1731_DEBUG_DEBUG(d):CID=0x80a70423;
[a9f8d6f7.433] # Event happen(94)

Jan  1 2010 10:09:09.236 HUAWEI %%01Y1731/7/Y1731_DEBUG_INFO(d):CID=0x80a70423;
[a9f96940.1011] # cailiang:Result update(0)
und
Jan  1 2010 10:09:10.244 HUAWEI %%01Y1731/7/Y1731_DEBUG_DEBUG(d):CID=0x80a70423;
[a9f8ddf8.319] # Receive SLM stat info from FEI >>
 uiFlowIndex:       142
 uiSeq:             0
 uiTxFCf:           0
 uiTxFCb:           0
 uiRxFCf:           1275
 uiRxFCl:           16125
```

Table 2-46 Description of the **debugging eoam-y1731** command output

Item	Description
uiFlowIndex	Index of a detected flow
uiSeq	Sequence number
uiTxFCf	Value of the local counter TxFCl at the time of CCM transmission
uiTxFCb	Value of the TxFCf field in the last received CCM
uiRxFCf	Value of the local counter RxFCl at the time of the reception of the last CCM
uiRxFCl	Number of CCMs received by the local end

2.10 Device Management Debugging Commands

2.10.1 1588v2 (PTP) Debugging Commands

NOTE

The 1588v2 function is supported only by the CE8850-64CQ-EI, CE6865EI, CE6880-24S4Q2CQ-EI and CE6880-48S4Q2CQ-EI.

2.10.1.1 debugging ptp

Function

The **debugging ptp** command enables 1588v2 protocol debugging.

The **undo debugging ptp** command disables 1588v2 protocol debugging.

By default, 1588v2 protocol debugging is disabled.

Format

debugging ptp { **protocol** | **event** | **packet** | **sync** | **all** }

undo debugging ptp { **protocol** | **event** | **packet** | **sync** | **all** }

Parameters

Parameter	Description	Value
protocol	Enables 1588v2 protocol module debugging.	-
event	Enables 1588v2 event debugging.	-
packet	Enables 1588v2 packet debugging.	-
sync	Displays debugging information about communication and data synchronization between MPUs and LPUs.	-
all	Displays all debugging information.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

If 1588v2 faults occur, run the **debugging ptp** command for fault location.

Configuration Impact

The **debugging ptp** command affects the system performance. Disable the debugging function after you complete troubleshooting.

Example

# Enable 1588v2 event debugging.

```
<HUAWEI> debugging ptp event
```

# Enable 1588v2 packet debugging.

```
<HUAWEI> debugging ptp packet
```

# Display debugging information about communication and data synchronization between MPUs and LPUs.

```
<HUAWEI> debugging ptp syn
```

# Enable 1588v2 protocol module debugging.

```
<HUAWEI> debugging ptp protocol
```

2.10.2 Fault Management Debugging Commands

2.10.2.1 debugging alarm

Function

The **debugging alarm** command enables the debugging of the alarm management module.

The **undo debugging alarm** command disables the debugging of the alarm management module.

By default, the debugging of the alarm management module is disabled.

Format

debugging alarm { **all** | **error** | **trace** | **fsm** | **corre** }

undo debugging alarm { **all** | **error** | **trace** | **fsm** | **corre** }

Parameters

Parameter	Description	Value
all	Enables the debugging of all functions of the alarm management module.	-
error	Enables the debugging of errors generated during the running of the alarm management module.	-
trace	Enables the debugging of procedures during the running of the alarm management module.	-
fsm	Enables the debugging of the alarm state machine.	-
corre	Enables the debugging of the alarm correlation.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

If you want to enable the debugging of the alarm management module or check the required debugging information by specifying parameters to help locate faults, run the **debugging alarm** command.

Example

# Enable the debugging of the alarm correlation.

```
<HUAWEI> debugging alarm error
```

2.10.3 Information Management Debugging Commands

2.10.3.1 undo debugging all

Function

The **undo debugging all** command disables all debugging of the device.

Format

undo debugging all

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

This command is used to disable all the debugging during the normal operation of a device.

Configuration Impact

Enabling debugging affects the system performance. It is recommended to enable debugging only when necessary.

Example

# Disable all the debugging on the device.

```
<HUAWEI> undo debugging all
```

2.10.4 PM Debugging Commands

2.10.4.1 debugging pm

Function

The **debugging pm** command enables performance management debugging.

The **undo debugging pm** command disables performance management debugging.

By default, performance management debugging is disabled.

Format

debugging pm { { { **statistics** | **sample** | **alarm** | **distribute** | **trace** | **error** | **config** | **packet** } [**pmocid** *cid* [**pmoindex** *index*]] } | **file** | **all** }

undo debugging pm { { { **statistics** | **sample** | **alarm** | **distribute** | **trace** | **error** | **config** | **packet** } [**pmocid** *cid* [**pmoindex** *index*]] } | **file** | **all** }

Parameters

Parameter	Description	Value
statistics	Specifies the statistic debugging.	-
sample	Enables sample statistics debugging.	-
file	Specifies the statistic of file debugging.	-
alarm	Specifies the statistic of alarm debugging.	-
all	Specifies all the debugging.	-
distribute	Enables distributed statistics debugging.	-
trace	Enables trace statistics debugging.	-
error	Enables error statistics debugging.	-
config	Enables configuration statistics debugging.	-
packet	Enables packet statistics debugging.	-
pmocid *cid*	Specifies a type of a statistic object.	The value is an integer ranging from 0 to 4294967295.
pmoindex *index*	Specifies an index of a statistic object.	The value is an integer ranging from 0 to 4294967295.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging pm** command enables performance management debugging, and the **undo debugging pm** command disables performance management debugging.

Example

# Enable performance management debugging.

```
<HUAWEI> debugging pm statistics
```

2.10.4.2 debugging pmlib

Function

The **debugging pmlib** command enables debugging of the pmlib module used for performance management.

The **undo debugging pmlib** command disables debugging of the pmlib module used for performance management.

By default, the debugging function is disabled for the pmlib module.

Format

debugging pmlib { **sample** | **trace** | **all** | **config** | **error** } **compid** *compid*

undo debugging pmlib { **sample** | **trace** | **all** | **config** | **error** } [**compid** *compid*]

Parameters

Parameter	Description	Value
sample	Enables the debugging function for sampling information.	-
trace	Enables the debugging function for packet tracing.	-
all	Enables the debugging function for all information.	-
config	Enables the debugging function for configuration information.	-
error	Enables the debugging function for error information.	-

Parameter	Description	Value
compid *compid*	Enables the debugging function for the component with a specified ID.	The value is an integer ranging from 0 to 4294967295.

Views

User view

Default Level

3: Management level

Usage Guidelines

To locate and analyze performance management problems, run the **debugging pmlib** command to enable debugging of the pmlib module.

Example

# Enable the debugging function for the sampling information of the pmlib module, with the component ID being 10945613.

```
<HUAWEI> debugging pmlib sample compid 10945613
```

2.11 Network Management Debugging Commands

2.11.1 LLDP Debugging Commands

2.11.1.1 debugging lldp

Function

The **debugging lldp** command enables the debugging function of the LLDP module.

The **undo lldp enable** command enables the debugging function of the LLDP module.

By default, the debugging function of the LLDP module is disabled.

Format

debugging lldp [**event** | **packet** | **all**] [**interface** *interface-type interface-number*]

undo debugging lldp [**event** | **packet** | **all**] [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
event	Enables debugging function of event.	-
packet	Enables debugging function of packet.	-
all	Enables all the debugging function.	-
interface *interface-type interface-number*	Enables debugging function of a specified interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging lldp** command displays the debugging information about the LLDP module. Different debugging information can be obtained by selecting different key words, which facilitates fault location and equipment maintenance.

Example

# Enable the event debugging of the LLDP module on the 10GE1/0/1 interface.

```
<HUAWEI> debugging lldp event interface 10ge 1/0/1
Dec 22 2012 16:29:48.216 CE6K_S_53.57 %%01LLDP/7/LLDP_DEBUG_EVENT(d):CID=0x80a50441; interface 
4,
Encode LLDP information frame successfully, length=334.
```

Table 2-47 Description of the **debugging lldp** command output

Item	Description
Dec 22 2012 16:29:48.216	Time when a debugging record is output.
LLDP	Module of the debugging information.
LLDP_DEBUG_EVENT(d)	The name and type of the debugging information.
CID=0x80a50441	The ID 0x80a50441
interface 4	The debugging information was generated on the interface with the index 4.
Encode LLDP information frame successfully, length=334	Debugging information.

2.11.2 NETCONF Debugging Command

2.11.2.1 debugging netconf

Function

Using the **debugging netconf** command, you can enable the debugging flag of NETCONF module.

Using the **undo debugging netconf** command, you can disable the debugging flag of NETCONF module.

By default, The debugging flag of NETCONF module is disabled.

Format

debugging netconf { **all** | **cfg-message** | **error** | **rpc** | **state-transition** | **tree** | **message** | **authorization** } [**session** *session-id*]

undo debugging netconf { **all** | **cfg-message** | **error** | **rpc** | **state-transition** | **tree** | **message** | **authorization** }

Parameters

Parameter	Description	Value
all	Enables or disables all debugging functions of NETCONF module.	-
cfg-message	Displays debugging information between NETCONF and CFG module..	-
error	Displays debugging information about error of NETCONF module.	-
rpc	Displays debugging information about RPC request and response of NETCONF module.	-
state-transition	Displays debugging information about state transition of NETCONF module.	-
tree	Displays debugging information about tree traversal of NETCONF module.	-
message	Displays debugging information about message exchange of NETCONF module.	-
authorization	Displays debugging information about NETCONF authorization.	-
session *session-id*	Displays debugging information about the NETCONF session with the specified ID.	It is an integer data type. The value range is from 1 to 65535.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
netconf	debug

Usage Guidelines

When a NETCONF module becomes faulty, the network administrator cannot perform local management using NETCONF to start, modify or delete configurations on the remote device. You can run this command to start the debugging information on the NETCONF module and rapidly locate faults based on the obtained information.

Example

# Enable NETCONF module message exchange information.

```
<HUAWEI> debugging netconf message
Sent Control Message of Type %lu with Trans No = %lu and Length = %lu to Interface %lu + Sub Interface
%lu
Aug  9 2011 03:47:26 HUAWEI %%01NETCONF/7/NCA_CTRL_MSG_SND(d):VR=0-CID=0x972727;Sent
Control Message of Type 8 with Trans No = 1 and Length = 180 to Interface 1 + Sub Interface 6

Sent Data Message of Type %lu with Packet Id = %lu and Ack Id = %lu and Length = %lu to Interface %lu
+ Sub Interface %lu
Aug  9 2011 03:47:26 HUAWEI %%01NETCONF/7/NCA_DATA_MSG_SND(d):VR=0-CID=0x972727;Sent Data
Message of Type 2 with Packet Id = 0 and Ack Id = 0 and Length = 1074 to Interface 1 + Sub Interface 7

Sent Message of Length %lu to PID %lu
Aug  8 2011 08:41:59 HUAWEI %%01NETCONF/7/NCA_NON_REL_MSG_SND(d):VR=0-CID=0x972727;Sent
Message of Length 20 to PID 9643811
```

# Enable NETCONF-CFG message interaction information.

```
<HUAWEI> debugging netconf cfg-message
CFG Infocode Block: Return Code = [0x%x], Atom Seq Num = [0x%x], Info Code Num = [0x%x]
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/INFOCODE_BLOCK_INFO(d):VR=0-CID=0x972727;CFG
Infocode Block: Return Code = [0x0], Atom Seq Num = [0x0], Info Code Num = [0x0]

CFG Infocode Item: Info code = [%u], Item Size = [%u], Info Type = [%u], Field Num = [%u]
Aug  9 2011 05:30:02 HUAWEI %%01NETCONF/7/INFOCODE_ITEM(d):VR=0-CID=0x972727;CFG Infocode
Item: Info code = [17], Item Size = [18], Info Type = [6], Field Num = [1]

CFG Infocode Item Field: Field Type = [%u], Field Length = [%u], Field Data = [%s]
Aug  9 2011 05:30:02 HUAWEI %%01NETCONF/7/INFOCODE_ITEM_FIELD(d):VR=0-CID=0x972727;CFG
Infocode Item Field: Field Type = [25], Field Length = [14], Field Data = [NtpAuthKeyCfg]

CFG Block Header: Block Index = [%u], Operation Type = [%s], Operation Length = [%u]
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/CFG_BLOCK_HEADER(d):VR=0-CID=0x972727;CFG Block
Header: Block Index = [1], Operation Type = [Unknown], Operation Length = [16]

CFG Object Block: Class Id. = [0x%x], Object Sequence Num = [%u], Field Num = [%u]
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/EDIT_CFG_OBJ_BLOCK(d):VR=0-CID=0x972727;CFG
```

```
Object Block: Class Id. = [0x8770195], Object Sequence Num = [1], Field Num = [2]

CFG Conditional Field Info: Field Id = [%u], Field Relation = [%u], Field Condition = [%u], Field Type =
[%u], Field Length = [%u]
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/CONDITION_FIELD_INFO(d):VR=0-CID=0x972727;CFG
Conditional Field Info: Field Id = [6], Field Relation = [2], Field Condition = [0], Field Type = [25], Field
Length = [8]

CFG Target Info: Source Target Id = [%u], Destination Target Id = [%u], Query Type = [%s]
Aug  9 2011 05:37:24 HUAWEI %%01NETCONF/7/QUERY_TARGET_INFO(d):VR=0-CID=0x972727;CFG
Target Info: Source Target Id = [14], Destination Target Id = [0], Query Type =
[CFG_QUERY_TYPE_CFGID_DIFF]

CFG sync-increment OR preview start OR end response: DB Id = [%u]
Aug  9 2011 05:37:24 HUAWEI %%01NETCONF/7/SYNC_INC_PREVIEW_START_END_RSP(d):VR=0-
CID=0x972727;CFG sync-increment OR preview start OR end response: DB Id = 0

CFG Query Field Info: Query Field Id = [%u]
Aug  9 2011 03:54:48 HUAWEI %%01NETCONF/7/QUERY_REQ_FIELD_INFO(d):VR=0-CID=0x972727;CFG
Query Field Info: Query Field Id = 2

CFG Query Response Block: Class Id = [0x%x], Display Template Id = [%u], Query Type = [%u], Diff Code =
[%u], Response Field Num = [%u], Record Num = [%u], Record Length = [%u]
Aug  9 2011 03:54:48 HUAWEI %%01NETCONF/7/QUERY_RSP_BLOCK_INFO(d):VR=0-CID=0x972727;CFG
Query Response Block: Class Id = [0x8770195], Display Template Id = [142049280], Query Type = [5], Diff
Code = [0], Response Field Num = [11], Record Num = [1], Record Length = [141]

CFG Query Response Field Info: Field Id = [%u], Field Type = [%u], Field Length = [%u]
Aug  9 2011 03:54:48 HUAWEI %%01NETCONF/7/QUERY_RSP_FIELD_INFO(d):VR=0-CID=0x972727;CFG
Query Response Field Info: Field Id = [1], Field Type = [32], Field Length = [4]

CFG Commit Request: Commit Type. = [%s], Time-out Value = [%u], Commit Mode = [%s]
Aug  9 2011 05:37:43 HUAWEI %%01NETCONF/7/COMIT_REQ_INFO(d):VR=0-CID=0x972727;CFG Commit
Request: Commit Type. = [CFG_COMMIT_TIME_IMMEDIATE], Time-out Value = [0], Commit Mode =
[CFG_COMMIT_MODE_ALLORNONE]

CFG Header: Transaction Num = [%u], Session Id = [%u], Sender Id = [0x%x], Block Num = [%u], Message
Type = [%s], Trans Flag = [%s], HA Flag = [%s], Fragment Flag = [%s], Edit Mode Flag = [%s], Sub
Sequence Num = [%u], Flow Id = [%u], Command Flag = [0x%x], Data Size = [%u]
Aug  9 2011 03:54:48 HUAWEI %%01NETCONF/7/CFG_MSG_HEADER(d):VR=0-CID=0x972727;CFG Header:
Transaction Num = [13], Session Id = [66], Sender Id = [0x972727], Block Num = [1], Message Type =
[MSG_CFGI_QUERY_DATA], Trans Flag = [CFG_MSGTRANS_TYPE_MID_REQ], HA Flag =
[CFG_HA_FLAG_ACTIVE], Fragment Flag
= [CFG_FRAG_FLAG_FALSE], Edit Mode Flag = [Unknown], Sub Sequence Num = [0], Flow Id = [0],
Command Flag = [0x0], Data Size = [4]

CFG Query Block: Class Id = [0x%x], Condition Field Num = [%u], Query Field Num = [%u]
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/QUERY_REQ_BLOCK_INFO(d):VR=0-CID=0x972727;CFG
Query Block: Class Id = [0x8770195], Condition Field Num = [2], Query Field Num = [1]
```

# Enable RPC request and response.

```
<HUAWEI> debugging netconf rpc
Aug  9 2011 03:52:25 HUAWEI HUAWEI %%01NETCONF/7/NCA_DEBUG_LOG_GENERIC(d):VR=0-
CID=0x972727;<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">
  <get-config>
    <source>
      <running/>
    </source>
    <filter type="subtree">
      <ifm xmlns="http://www.huawei.com/netconf/vrp" content-version="1.0" format-version="1.0">
        <interfaces>
          <interface/>
        </interfaces>
      </ifm>
    </filter>
  </get-config>
</rpc>
```

```
Aug  9 2011 03:52:26 HUAWEI %%01NETCONF/7/NCA_DEBUG_LOG_GENERIC(d):VR=0-CID=0x972727;<?
xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data/>
</rpc-reply>
```

# Enable NETCONF tree traversal information.

```
<HUAWEI> debugging netconf tree
Node Type = %s, Node Name = %s
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/NCA_TRAVERSE_NODE_INFO(d):VR=0-
CID=0x972727;Node Type = Service Node, Node Name = radius

Traversal Type = %s
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/NCA_TRAVERSE_TYPE(d):VR=0-CID=0x972727;Traversal
Type = EN_NCA_TRAVERSE_CHILD
```

# Enable NETCONF state transition information.

```
<HUAWEI> debugging netconf state-transition
Session state changed.(session-id=%hu, current-state=%s, next-state=%s.)
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/NCA_SESS_STATE_TRANSITION(d):VR=0-
CID=0x972727;Session state changed.(session-id=66, current-state=READY, next-state=WAIT_OP_RESP.)

Transaction state changed.(session-id=%hu, current-state=%s, next-state=%s.)
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/NCA_TRANS_STATE_TRANSITION(d):VR=0-
CID=0x972727;Transaction state changed.(session-id=66, current-state=NEUTRAL, next-
state=WAIT_CREATE_TRANS_RSP.)

Session query state changed.(session-id=%hu, current-state=%s, next-state=%s.)
Aug  9 2011 03:54:31 HUAWEI %%01NETCONF/7/NCA_SESS_QUERY_STATE_TRANSIT(d):VR=0-
CID=0x972727;Session query state changed.(session-id=66, current-state=QUERY, next-
state=QUERY_AUTO_KEY.)

Pipeline query state changed.(session-id=%hu, current-state=%s, next-state=%s.)
Aug  9 2011 03:47:29 HUAWEI %%01NETCONF/7/NCA_OP_QUERY_STATE_TRANSIT(d):VR=0-
CID=0x972727;Pipeline query state changed.(session-id=66, current-state=QUERY, next-state=SYNC_START.)
```

# Enable NETCONF internal error information.

```
<HUAWEI> debugging netconf error
Error Location: File: %s, Line: %u, Error Code: 0x%x.
Aug  9 2011 03:54:48 HUAWEI %%01NETCONF/7/NCA_INTERNAL_ERROR(d):VR=0-CID=0x972727;Error
Location: File: nca_bsc_op_data.c, Line: 3524, Error Code: 0x800a62.
```

Table 2-48 Description of the debugging netconf command output

Item	Description
CFG Infocode Block	Indicates the configuration infocode block that includes return code, atom sequence number and info code number information.
CFG Infocode Item	Indicates the configuration infocode item that includes info code, item size, info type and field number information.
CFG Infocode Item Field	Indicates the configuration infocode item field that includes field type, length and data information.
CFG Block Header	Indicates the configuration block header that includes block index, operation type and length information.

Item	Description
CFG Object Block	Indicates the configuration object block that includes class ID, object sequence number and field number information.
CFG Conditional Field Info	Indicates the configuration conditional field information that includes field ID, relation, condition, type and length information.
CFG Target Info	Indicates the configuration target information that includes source target, destination target and query type information.
CFG sync-increment OR preview start OR end response	Indicates the configuration sync-increment, preview start or end response information.
CFG Query Field Info	Indicates the configuration query field ID information.
CFG Query Response Block	Indicates the configuration query response block that includes class ID, display template ID, query type, diff code, response field number, record number and record length information.
CFG Query Response Field Info	Indicates the configuration query response field information that includes field ID, type and length information.
CFG Commit Request	Indicates the configuration commit request that includes commit type, time-out value and commit mode information.
CFG Header	Indicates the configuration header that includes transaction number, session ID, sender ID, block number, message type, HA flag, fragment flag, edit mode flag, sub sequence number, flow ID, command flag and data size information.
CFG Query Block	Indicates the configuration query that includes class ID, condition field number and query field number information.
Node Type	Indicates the type of node.
Node Name	Indicates the name of node.
Transaction Type	Indicates the type of transaction.
Error Location	Indicates the error location that includes file, line and error code information.

2.11.3 OpenFlow Debugging Commands

2.11.3.1 debugging sdn fp-data

Function

The **debugging sdn fp-data** command enables the Forwarding Point Client (FPC) debugging on the forwarder.

The **undo debugging sdn fp-data** command disables the FPC debugging on the forwarder.

By default, the FPC debugging is disabled on the forwarder.

Format

debugging sdn fp-data { **all** | **error** | **process** }

undo debugging sdn fp-data { **all** | **error** | **process** }

Parameters

Parameter	Description	Value
all	Specifies all FPC debugging functions.	-
error	Specifies error debugging.	-
process	Specifies process debugging.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

To view the FPC debugging information on a forwarder, run this command. You can view different types of debugging information by specifying different keywords. The debugging information helps you locate faults and maintain devices.

Prerequisites

The OpenFlow Agent function has been enabled on the forwarder.

Example

# Enable all FPC debugging functions.

```
<HUAWEI> debugging sdn fp-data all
<HUAWEI>
Oct 29 2014 02:51:13.936 HUAWEI %%01SDNCCOMM/7/
NORMALDEBUGOUT(d):CID=0x80ff0439;
[APC] <14:00655>: fpc CMF:CMF Add Op Info Area 0,
0

Oct 29 2014 02:51:13.936 HUAWEI %%01SDNCCOMM/7/
NORMALDEBUGOUT(d):CID=0x80ff0439;
[APC] <14:00667>: fpc CMF:CMF Send Rsp
Msg

Oct 29 2014 02:51:13.937 HUAWEI %%01SDNCCOMM/7/
NORMALDEBUGOUT(d):CID=0x80ff0439;
[APC] <21:00129>: Receive message : MsgLen[72] Intf[1]
SubIntf[0]

Oct 29 2014 02:51:13.938 HUAWEI %%01SDNCCOMM/7/
NORMALDEBUGOUT(d):CID=0x80ff0439;
[APC] <22:00222>: Receive APPCFG MSG!

Oct 29 2014 02:51:13.938 HUAWEI %%01SDNCCOMM/7/
NORMALDEBUGOUT(d):CID=0x80ff0439;
[APC] <14:00303>: Get SMP Msg(0x2)!

Oct 29 2014 02:51:13.938 HUAWEI %%01SDNCCOMM/7/
NORMALDEBUGOUT(d):CID=0x80ff0439;
[APC] <14:00330>: Process ENUM MSG APPCFGI ACTION.
```

2.11.3.2 debugging sdn openflow

Function

The **debugging sdn openflow** command enables OpenFlow session debugging on the forwarder.

The **undo debugging sdn openflow** command disables OpenFlow session debugging on the forwarder.

By default, the OpenFlow session debugging is disabled on the forwarder.

Format

debugging sdn openflow { **all** | **flow-mod** [**table** *table-id*] | **port_status** | **session** }

undo debugging sdn openflow { **all** | **flow-mod** [**table** *table-id*] | **port-status** | **session** }

Parameters

Parameter	Description	Value
all	Specifies all debugging functions of the OpenFlow session module on the forwarder.	-

Parameter	Description	Value
flow-mod	Specifies the debugging of OpenFlow flow table delivery.	-
table *table-id*	Specifies the flow table ID.	The value is an integer that ranges from 0 to 4294967295.
port-status	Specifies the debugging of OpenFlow interface information reporting.	-
session	Specifies the debugging of OpenFlow session process.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

To view the OpenFlow session debugging information on a forwarder, run this command. You can view different types of debugging information by specifying different keywords. The debugging information helps you locate faults and maintain devices.

Prerequisites

The OpenFlow Agent function has been enabled on the forwarder.

Example

# Enable all debugging functions of the OpenFlow session module on the forwarder.

```
<HUAWEI> debugging sdn openflow all
<HUAWEI>
Nov 28 2011 07:37:13.221 HUAWEI %%01TEBASE/7/NORMALDEBUGOUT(d):=Admin--CID=0
x80180440;
[LSC] <301900296>: Create log timer fail, report fail to ssp
```

2.11.3.3 display debugging sdn fp-data

Function

The **display debugging sdn fp-data** command disables the status of the FPC debugging on the forwarder.

Format

display debugging sdn fp-data

Parameters

None

Views

All views

Default Level

1: Monitoring level

Usage Guidelines

Usage Scenario

To check whether the FPC debugging is enabled on a forwarder, run the **display debugging sdn fp-data** command. The FPC debugging functions include:

- SDN fp-data error debug
- SDN fp-data event debug
- SDN fp-data process debug

Prerequisites

The FPC debugging has been enabled.

Example

# View the status of the FPC debugging on the forwarder.

```
<HUAWEI> display debugging sdn fp-data
Sdn FP client process debugging switch is on
Sdn FP client error debugging switch is on
Sdn FP client event debugging switch is on
```

2.11.3.4 display debugging sdn openflow

Function

The **display debugging sdn openflow** command displays the status of the OpenFlow session debugging on the forwarder.

Format

display debugging sdn openflow

Parameters

None

Views

All views

Default Level

1: Monitoring level

Usage Guidelines

Usage Scenario

To check whether the OpenFlow session debugging is enabled on a forwarder, run the **display debugging sdn openflow** command. The OpenFlow session debugging functions include:

- SDN OpenFlow flow_mod debug
- SDN OpenFlow port_status debug
- SDN OpenFlow session debug

Prerequisites

The OpenFlow session debugging has been enabled.

Example

# View the status of the OpenFlow session debugging on the forwarder.

```
<HUAWEI>display debugging sdn openflow
Sdn Openflow Port Status debugging switch is on
Sdn Openflow Flow_mod debugging switch is
on
Sdn Openflow Session debugging switch is on
```

2.11.4 OVSDB Debugging Commands

2.11.4.1 debugging ovsdb client

Function

The **debugging ovsdb client** command enables the debugging function for an open vSwitch database (OVSDB) client component.

The **undo debugging ovsdb client** command disables the debugging function for the OVSDB client component.

By default, the debugging function for the OVSDB client component is disabled.

Format

debugging ovsdb client [**cfgi** | **rpc** | **ha** | **mac** | **monitor** | **transact**] { **error** | **info** }

undo debugging ovsdb client [**cfgi** | **rpc** | **ha** | **mac** | **monitor** | **transact**]

Parameters

Parameter	Description	Value
cfgi	Indicates the debugging function for CFGI information.	-
rpc	Indicates the debugging function for RPC information.	-
ha	Indicates the debugging function for HA information.	-
mac	Indicates the debugging function for MAC address information.	-
transact	Indicates the debugging function for Transact information about the OVSDB database.	-
error	Sets the information printing level to **ERROR**.	-
info	Sets the information printing level to **INFO** and **ERROR**.	-

Views

Diagnose view

Default Level

3: Management level

Usage Guidelines

By enabling the debugging function for the OVSDB client component, you can query the debugging information in code for easy fault locating and troubleshooting.

Example

# Enable all debugging functions of the OVSDB client component and set the information printing level to **ERROR**.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging ovsdb client error
```

# Enable the debugging function for MAC address information about the OVSDB client component and set the information printing level to **INFO** and **ERROR**.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging ovsdb client mac info
```

# Disable all debugging functions of the OVSDB client component.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] undo debugging ovsdb client
```

# Disable the debugging function for MAC address information about the OVSDB client component.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] undo debugging ovsdb client mac
```

2.11.4.2 debugging ovsdb server

Function

The **debugging ovsdb server** command enables the debugging function for an open vSwitch database (OVSDB) server component.

The **undo debugging ovsdb server** command disables the debugging function for the OVSDB server component.

By default, the debugging function for the OVSDB server component is disabled.

Format

debugging ovsdb server [**rpc**] { **error** | **info** }

undo debugging ovsdb server [**rpc**]

Parameters

Parameter	Description	Value
rpc	Indicates the debugging function for RPC information.	-
error	Sets the information printing level to **ERROR**.	-
info	Sets the information printing level to **INFO** and **ERROR**.	-

Views

Diagnose view

Default Level

3: Management level

Usage Guidelines

By enabling the debugging function for the OVSDB server component, you can query the debugging information in code for easy fault locating and troubleshooting.

Example

# Enable the debugging function for RPC information about the OVSDB server component and set the information printing level to **ERROR**.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging ovsdb server rpc error
```

# Disable the debugging function for the OVSDB server component.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] undo debugging ovsdb server
```

2.11.5 Netstream Debugging Commands

2.11.5.1 debugging netstream

Function

The **debugging netstream** command enables debugging of NetStream.

The **undo debugging netstream** command disables debugging of NetStream.

By default, NetStream debugging is disabled.

Format

debugging netstream { **event** | **packet** } { **ip** | **ipv6** | **ethernet** | **vxlan inner-ip** }

undo debugging netstream { **event** | **packet** } { **ip** | **ipv6** | **ethernet** | **vxlan inner-ip** }

Parameters

Parameter	Description	Value
event	Enables event debugging.	-
packet	Enables packet debugging.	-
ip	Enables event or packet debugging about IP flow.	-
ipv6	Enables event or packet debugging about IPv6 flow.	-
ethernet	Enables event or packet debugging about ethernet flow.	-
vxlan inner-ip	Enables event or packet debugging about VXLAN flexible flow.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
netstream	debug

Usage Guidelines

- The **event** command displays the messages received by the local NetStream component.
- The **packet** command displays whether packets of the NetStream component are sent normally.

Example

# Enable the debugging of NetStream events related to IP flows.

```
<HUAWEI> debugging netstream event ip
```

# Enable the debugging of NetStream packets related to IP flows.

```

<HUAWEI> debugging netstream packet ip
```

2.11.6 NTP Debugging Commands

2.11.6.1 debugging ntp

Function

The **debugging ntp** command enables NTP debugging.

The **undo debugging ntp** command disables NTP debugging.

By default, NTP debugging is disabled.

Format

debugging ntp [**access** | **adjustment** | **authentication** | **event** | **filter** | **packet** [**ipv6**] [**receive** | **send**] | **parameter** | **refclock** | **selection** | **synchronization** | **validity** | **all**]

undo debugging ntp [**access** | **adjustment** | **authentication** | **event** | **filter** | **packet** [**ipv6**] [**receive** | **send**] | **parameter** | **refclock** | **selection** | **synchronization** | **validity** | **all**]

Parameters

Parameter	Description	Value
access	Enables debugging of access control.	-
adjustment	Enables debugging of clock adjustment.	-
authentication	Enables authentication debugging.	-
event	Enables event debugging.	-
filter	Enables debugging of loopback filter information.	-
packet [**ipv6**] **receive**	Enables received packet debugging.	-

Parameter	Description	Value
packet [**ipv6**] **send**	Enables sended packet debugging.	-
parameter	Enables debugging of clock parameters.	-
refclock	Enables debugging of reference clocks.	-
selection	Enables debugging of clock selection information.	-
synchronization	Enables debugging of clock synchronization information.	-
validity	Enables debugging of validity of the remote host.	-
all	Enables all debugging functions.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
ntp	debug

Usage Guidelines

The **debugging ntp**command displays all types of NTP debugging information, or the specified type of NTP debugging information when you configure a key word so that you can determine whether the currently received and sent packets are correct.

Example

# Enable debugging of NTP authentication.

```
<HUAWEI> debugging ntp authentication
```

2.11.7 SNMP Debugging Commands

2.11.7.1 debugging snmp-agent

Function

The **debugging snmp-agent** command enables SNMP debugging functions.

The **undo debugging snmp-agent** command disables SNMP debugging functions.

By default, SNMP debugging is disabled.

Format

debugging snmp-agent { **agent** | **all** | **dispatch** | **misc** | **msgproc** | **notify** | **packet** [**peer-ip** *ip-address* | [**query** | **set** | **notification**] **mib-node** *node-oid*] | **proxy** | **security** | **set-cache** | **shell** | **vacm** }

undo debugging snmp-agent { **agent** | **all** | **dispatch** | **misc** | **msgproc** | **notify** | **packet** | **proxy** | **security** | **set-cache** | **shell** | **vacm** }

Parameters

Parameter	Description	Value
agent	Enables debugging of the SNMP module agent.	-
all	Enables debugging of all SNMP modules.	-
dispatch	Enables debugging of the SNMP module dispatch.	-
msgproc	Enables debugging of the SNMP module msgproc.	-
misc	Enables debugging of the SNMP module misc.	-
notify	Enables debugging of the SNMP module notify.	-
packet	Enables packet debugging of SNMP modules.	-
peer-ip *ip-address*	Enables debugging information about the specified peer IP address. **NOTE** The parameter has been available since V100R005C10.	The value is in dotted decimal notation.
query	Enables debugging information about the query operation. **NOTE** The parameter has been available since V100R005C10.	-

Parameter	Description	Value
set	Enables debugging information about the SET operation. **NOTE** The parameter has been available since V100R005C10.	-
notification	Enables debugging information about the Trap or Inform operation. **NOTE** The parameter has been available since V100R005C10.	-
mib-node *node-oid*	Enables debugging information about the MIB node of a specified OID. **NOTE** The parameter has been available since V100R005C10.	The value is a string of 1 to 255 characters.
security	Enables debugging of the SNMP module security.	-
shell	Enables debugging of the SNMP module shell.	-
set-cache	Enables debugging of the SNMP set message caching function.	-
proxy	Enables debugging of the SNMP module proxy.	-
vacm	Enables debugging of the SNMP module vacm.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
snmp	debug

Usage Guidelines

Usage Scenario

This command displays SNMP debugging information of all types or of the specified type. The debugging information shows whether the packets are successfully sent and received.

Precautions

The debugging information is displayed on screen. Too much debugging information displayed will degrade device performance, so disable the debugging functions if you do not want to debug the device.

Example

# Enable all debugging functions of SNMP.

```
<HUAWEI> debugging snmp-agent all
2011-03-08 02:15:49 HUAWEI %%01snmp/3/SHELL(d):VS=0-CID=2161452883;[IPS MESSAGE RECEIVED]
Interface = 1, Sub Interface = 1 Type = 16 Sender Id = 0x602712
```

# If a message is received, the SNMP message receiving function is normal and the component ID can be obtained.

```
2011-03-08 02:15:49 HUAWEI %%01snmp/3/SHELL(d):VS=0-CID=2161452883;SNMP has sended the FM
ack msg !
```

# SNMP sends a response message to the FM.

```
2011-03-08 02:15:49 HUAWEI %%01snmp/3/SHELL(d):VS=0-CID=2161452883;[IPS MESSAGE RECEIVED]
Interface = 1, Sub Interface = 1 Type = 16 Sender Id = 0x602712
```

2.11.8 IP FPM Debugging Commands

2.11.8.1 display debugging ipfpm-dcp

Function

The **display debugging ipfpm-dcp** command displays the DCP debugging status.

Format

display debugging ipfpm-dcp

Parameters

None

Views

All views

Default Level

3: Management level

Usage Guidelines

To check whether debugging is enabled or disabled for a DCP in the IP FPM model, run the **display debugging ipfpm-dcp** command in all views.

Example

# Display the DCP debugging status.

```
<HUAWEI> system-view
[~HUAWEI] display debugging ipfpm-dcp
IPFPM DCP packet debugging switch is on
```

2.11.8.2 display debugging ipfpm-mcp

Function

The **display debugging ipfpm-mcp** command displays the MCP debugging status.

Format

display debugging ipfpm-mcp

Parameters

None

Views

All views

Default Level

3: Management level

Usage Guidelines

To check whether debugging is enabled or disabled for an MCP in the IP FPM model, run the **display debugging ipfpm-mcp** command in all views.

Example

# Display the MCP debugging status.

```
<HUAWEI> system-view
[~HUAWEI] display debugging ipfpm-mcp
IPFPM MCP packet debugging switch is on
```

2.11.9 Telemetry Debugging Commands

2.11.9.1 debugging telemetry grpc packet

Function

The **debugging telemetry grpc packet** command enables the gRPC module to output packet information.

The **undo debugging telemetry grpc packet** command disables the gRPC module from outputting packet information.

By default, the gRPC module is disabled from outputting packet information.

Format

debugging telemetry grpc packet

undo debugging telemetry grpc packet

Parameters

None

Views

Diagnostic view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
grpc	debug

Usage Guidelines

If you want to view the related content of the gRPC packet, such as the packet sending rate, you can run this command to check the gRPC packet sending information.

Example

# Enable the gRPC module to output packet information

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging telemetry grpc packet
```

2.11.10 Twamp Debugging Commands

2.11.10.1 debugging twamp

Function

The **debugging twamp** command enables debugging for TWAMP.

The **undo debugging twamp** command disables debugging for TWAMP.

By default, NetStream debugging is disabled.

Format

debugging twamp { **error** | **process** } [**client-ip** *set-client-ip* **client-port** *set-client-port* [**vpn-instance** *vpn-instance-name*]]

undo debugging twamp

Parameters

Parameter	Description	Value
error	Enables the TWAMP error debugging.	-
process	Enables the TWAMP process debugging.	-
client-ip *set-client-ip*	Specifies the IP address of the Client.	The value is in dotted decimal notation.
client-port *set-client-port*	Specifies the port of the Client.	The value is an integer ranging from 0 to 65535.
vpn-instance *vpn-instance-name*	Specifies a VPN instance.	The value is a string of 1 to 31 case-sensitive characters, spaces not supported. In addition, the VPN instance name must not be _public_. When double quotation marks are used around the string, spaces are allowed in the string.

Views

User view

Default Level

3: Management level

Usage Guidelines

To debug the entire TWAMP process, you can run the **debugging twamp process** command to enable TWAMP debugging.

Example

# Configure the TWAMP error debugging for the control-client with an IP address 1.1.1.1.

```
<HUAWEI> debugging twamp error client-ip 1.1.1.1 client-port 862
```

2.12 MPLS Debugging Commands

NOTE

Only the CE8850EI, CE8860EI, CE8861EI, CE8868EI, CE7850EI, CE7855EI, CE6870EI, CE6875EI, CE6860EI, CE6865EI, CE6857EI, CE6856HI, CE6855HI, CE6850U-HI, CE6851HI and CE6850HI switches support MPLS.

2.12.1 LDP Debugging Commands

2.12.1.1 debugging packet ldp interface

Function

The **debugging packet ldp interface** command enables a device to trace Hello messages of a local LDP session.

The **undo debugging packet ldp interface** command disables a device from tracing Hello messages of a local LDP session.

The Hello message trace function is disabled by default.

Format

debugging packet ldp interface *interface-type interface-number* [**verbose**]

undo debugging packet ldp interface *interface-type interface-number* [**verbose**]

Parameters

Parameter	Description	Value
interface-type interface-number	Specifies the type and number of an interface.	-
verbose	Displays detailed information.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging packet ldp interface** command checks whether protocol packets are lost between components.

Example

# Trace Hello messages sent by a local LDP session on VLANIF20.

```
<HUAWEI> debugging packet ldp interface interface vlanif 20
<HUAWEI> terminal debugging
LDM:
---------------------------------------------
My Cid          : 0x8078273F
Peer Cid        : 0x802738
VS            : 0
Handle          : 1
TraceNum          : 5
Direction       : Down
Status         : 0
Interface index : 12
Link type       : ETH
Source mac      : 38 00 10 03 00 07
Dest mac        : 01 00 5e 00 00 02
Link protocol   : 0x0800
Protocol        : IPV4
Time           : 2012-12-22 12:12:9 329
Data           :
0x01005E00000238001003000708045C00046380C0000FF1197D60A010101E000000202860286003 2BD050
00100260101010900000100001C000040FE04000004
---------------------------------------------
```

Table 2-49 Description of the **debugging packet ldp interface** command output

Item	Description
My Cid	ID of the LDM Component.
Peer Cid	ID of the SOCK component.
VS	Virtual router number.
Handle	Handle for tracing protocol packets.
TraceNum	Number of protocol packet tracing.
Direction	Receiving or sending direction of a protocol packet. The sending direction is Down, and the receiving direction is Up.

Item	Description
Status	Tracing status of a protocol packet: ● **0**: success ● **1**: parameter error ● **2**: discard due to backpressure ● **3**: system error
Interface index	Interface index
Link type	Link type
Source mac	Source MAC address carried in the packet header
Dest mac	Destination MAC address carried in the packet header
Link protocol	Link layer protocol
Protocol	Protocol address family
Time	Time when the packet is received
Data	Packet information

2.12.1.2 debugging packet ldp peer

Function

The **debugging packet ldp peer** command enables TCP packet tracing on an LDP peer.

By default, TCP packet tracing on LDP peers is disabled.

Format

debugging packet ldp peer *lsr-id* [**verbose**]

undo debugging packet ldp peer *lsr-id*

Parameters

Parameter	Description	Value
lsr-id	Specifies the LSR ID of an LDP neighbor.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

Usage Scenario

The **debugging packet ldp peer** command checks whether TCP packets are lost between components.

Example

# Trace TCP packets of the LDP neighbor whose address is 10.1.1.1.

```
<HUAWEI> debugging packet ldp peer 10.1.1.1
<HUAWEI> terminal debugging

LDC:
---------------------------------------------
My Cid         : 0x801c0455
Peer Cid       : 0x80650406
VS           : 0
Handle         : 6
TraceNum         : 110
Direction       : Down
Status         : 0
Time           : 2012-12-22 16:22:26 747
Data           :
---------------------------------------------
```

Table 2-50 Description of the **debugging packet ldp peer** command output

Item	Description
My Cid	ID of the LDC component.
Peer Cid	ID of the SOCK component.
VS	ID of a virtual system.
Handle	Handle for tracing protocol packets.
TraceNum	Number of protocol packet tracing.
Direction	Receiving or sending direction of a protocol packet. The sending direction is Down, and the receiving direction is Up.
Status	Tracing status of a protocol packet: ● **0**: success ● **1**: parameter error ● **2**: discard due to backpressure ● **3**: system error

Item	Description
Time	Time when protocol packets are generated. The time format is YYYY-MM-DD HH:MM:SS MS.
Data	Packet information.

2.12.1.3 debugging packet ldp remote-peer

Function

The **debugging packet ldp remote-peer** command enables packet tracing of LDP remote peers, namely unicast packet tracing.

By default, unicast packet tracing is disabled.

Format

debugging packet ldp remote-peer [**verbose**]

undo debugging packet ldp remote-peer [**verbose**]

Parameters

None.

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging packet ldp remote-peer** command checks whether protocol packets are lost between components.

Example

# Trace packets of LDP remote peers.

```
<HUAWEI> debugging packet ldp remote-peer
<HUAWEI> terminal debugging

LNM:
----------------------------------------------
My Cid          : 0x801b0439
Peer Cid        : 0x80650406
VS              : 0
Handle          : 4
TraceNum        : 3
```

```
Direction        : Up
Status          : 0
Time            : 2011-12-22 16:45:39 977
Data            :
---------------------------------------------
```

Table 2-51 Description of the **debugging packet ldp remote-peer** command output

Item	Description
My Cid	ID of the LNM Component.
Peer Cid	ID of the SOCK component.
VS	ID of a virtual system.
Handle	Handle for tracing protocol packets.
TraceNum	Number of protocol packet tracing.
Direction	Receiving or sending direction of a protocol packet. The sending direction is Down, and the receiving direction is Up.
Status	Tracing status of a protocol packet: ● **0**: success ● **1**: parameter error ● **2**: discard due to backpressure ● **3**: system error
Time	Time when protocol packets are generated. The time format is YYYY-MM-DD HH:MM:SS MS.
Data	Packet information.

2.12.1.4 display debugging mpls ldp

Function

The **display debugging mpls ldp** command displays information about all configured LDP debugging functions.

Format

display debugging mpls ldp

Parameters

None

Views

User view

Default Level

1: Monitoring level

Usage Guidelines

After you enable the LDP debugging functions and configure filtering policies, you can run the **display debugging mpls ldp** command to view information about all configured LDP debugging functions. The command output helps you check whether the debugging functions and the filtering policies are configured successfully.

Example

# Display information about all configured LDP debugging functions.

```
<HUAWEI> display debugging mpls ldp
LDP discovery debugging switch is on
LDP session debugging switch is on
LDP socket debugging switch is on
```

Table 2-52 Description of the **display debugging mpls ldp** command output

Item	Description
LDP session debugging switch is on	Debugging of LDP sessions
LDP socket debugging switch is on	Debugging of LDP Socket
LDP discovery debugging switch is on	Debugging of LDP peers

2.12.1.5 debugging mpls ldp discovery

Function

The **debugging mpls ldp discovery** command enables all LDP peer discovery debugging functions and displays debugging information.

The **undo debugging mpls ldp discovery** command disables the LDP peer discovery debugging functions.

By default, the debugging function is disabled.

Format

debugging mpls ldp discovery { **error** | **events** | **interface** [*interface-type interface-number*] | **job** | **partner** [**packet**] | **peer** [*peer-id*] | **socket** [*peer-id*] [**receive** | **send**] [**packet**] }

undo debugging mpls ldp discovery { **all** | **error** | **events** | **interface** [*interface-type interface-number*] | **job** | **partner** [**packet**] | **peer** [*peer-id*] | **socket** [*peer-id*] [**receive** | **send**] [**packet**] }

debugging mpls ldp discovery partner session-manager control-information [**packet**]

undo debugging mpls ldp discovery partner session-manager control-information [**packet**]

Parameters

Parameter	Description	Value
error	Displays error debugging information.	-
events	Displays event debugging information.	-
interface *interface-type interface-number*	Displays debugging information about a specified interface.	-
job	Displays job debugging information.	-
partner	Displays partner debugging information.	-
packet	Enables or disables the packet debugging.	-
peer	Enables or disables the LDP peer debugging.	-
peer-id	Displays debugging information about an LDP peer with a specified LSR ID.	The value is in dotted decimal notation.
socket	Displays socket debugging information.	-
receive	Enables or disables the socket function for received packets.	-
send	Enables or disables the socket function for sent packets.	-
session-manager control-information	Enables the debugging function for communication between the LDP peer management module and LDP session management module.	-
all	Disables all debugging functions.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mpls-ldp	debug

Usage Guidelines

To view information about LDP peer discovery packets, run the **debugging mpls ldp discovery** command. The debugging information helps check whether the packet format is correct and whether the packets are transmitted properly if LDP session negotiation fails.

Example

# Enable event debugging of the LDP peer discovery module.

```
<HUAWEI> debugging mpls ldp discovery events
```

# Enable interface debugging of the LDP peer discovery module.

```
<HUAWEI> debugging mpls ldp discovery interface 10GE 3/0/8
```

# Enable partner debugging of the LDP peer discovery module.

```
<HUAWEI> debugging mpls ldp discovery partner packet
```

# Enable socket debugging of the LDP peer discovery module.

```
<HUAWEI> debugging mpls ldp discovery socket packet
```

2.12.1.6 debugging mpls ldp lsp

Function

The **debugging mpls ldp lsp** command enables the debugging of the LDP LSP manager.

The **undo debugging mpls ldp lsp** command disables the debugging of the LDP LSP manager.

By default, the debugging function is disabled.

Format

debugging mpls ldp lsp { **error** | **events** | **fec** [*ip-address mask*] | **job** | **partner** [**packet**] }

undo debugging mpls ldp lsp { **all** | **error** | **events** | **fec** [*ip-address mask*] | **job** | **partner** [**packet**] }

debugging mpls ldp lsp partner route-manager [{ **fec** [*ip-address mask*] | **nexthop** [*ip-address*] | **control-information** } [**packet**]]

undo debugging mpls ldp lsp partner route-manager [{ **fec** [*ip-address mask*] | **nexthop** [*ip-address*] | **control-information** } [**packet**]]

debugging mpls ldp lsp partner { **forward-manager** | **tunnel-manager** } [{ **fec** [*ip-address mask*] | **control-information** } [**packet**]]

undo debugging mpls ldp lsp partner { **forward-manager** | **tunnel-manager** } [{ **fec** [*ip-address mask*] | **control-information** } [**packet**]]

debugging mpls ldp lsp partner session-manager control-information [**packet**]

undo debugging mpls ldp lsp partner session-manager control-information [**packet**]

Parameters

Parameter	Description	Value
error	Displays error debugging information.	-
events	Displays event debugging information.	-
fec [*ip-address mask*]	Specifies the destination address mapped to a FEC.	The value is in dotted decimal notation.
job	Displays job debugging information.	-
partner [**packet**]	Displays partner debugging information.	-
all	Disables all debugging functions.	-
partner route-manager	Enables the debugging function for communication between the LDP LSP manager and route management module.	-
partner forward-manager	Enables the debugging function for communication between the LDP LSP manager and forwarding engine management module.	-
partner tunnel-manager	Enables the debugging function for communication between the LDP LSP manager and tunnel manager.	-
nexthop [*ip-address*]	Specifies a next hop IP address.	-
control-information	Enables the control information debugging.	-
partner session-manager	Enables the debugging function for communication between the LDP manager and LDP session management module.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mpls-ldp	debug

Usage Guidelines

To view LDP LSP manager debugging information, run the **debugging mpls ldp lsp** command. The debugging information helps check whether the packet format is correct and whether the packets are transmitted properly if an LDP LSP fails.

Example

# Enable the error debugging of the LDP LSP manager.

```
<HUAWEI> debugging mpls ldp lsp error
```

# Enable event debugging of the LDP LSP manager.

```
<HUAWEI> debugging mpls ldp lsp events
```

# Enable partner debugging of the LDP LSP manager.

```
<HUAWEI> debugging mpls ldp lsp partner
```

2.12.1.7 debugging mpls ldp session

Function

The **debugging mpls ldp session** command enables debugging functions of the MPLS LDP session management module and displays debugging information.

The **undo debugging mpls ldp session** command disables the MPLS LDP session debugging functions.

The debugging is disabled by default.

Format

debugging mpls ldp session { **error** | **events** | **job** }

undo debugging mpls ldp session { **error** | **events** | **job** }

debugging mpls ldp session peer [*peer-id*]

undo debugging mpls ldp session peer [*peer-id*]

debugging mpls ldp session partner [**packet**]

undo debugging mpls ldp session partner [**packet**]

debugging mpls ldp session socket [*peer-id*] [**init** | **keepalive** | **label**] [**receive** | **send**] [**packet**]

undo debugging mpls ldp session socket [*peer-id*] [**init** | **keepalive** | **label**] [**receive** | **send**] [**packet**]

debugging mpls ldp session partner { **lsp-manager** | **neighbor-manager** } **control-information** [**packet**]

undo debugging mpls ldp session partner { **lsp-manager** | **neighbor-manager** } **control-information** [**packet**]

undo debugging mpls ldp session all

Parameters

Parameter	Description	Value
error	Enables or disables the error debugging.	-
events	Enables or disables the event debugging.	-
job	Enables or disables the job debugging.	-
peer	Enables or disables the LDP peer debugging.	-
peer-id	Displays debugging information about an LDP peer with a specified LSR ID.	This value is in dotted decimal notation.
partner	Enables or disables the partner debugging.	-
packet	Enables or disables the packet debugging.	-
socket	Enables or disables the socket debugging.	-
init	Enables or disables the socket function for initialization packets.	-
keepalive	Enables or disables the socket function for Keepalive packets.	-
label	Enables or disables the socket function for label packets.	-
receive	Enables or disables the socket function for received packets.	-
send	Enables or disables the socket function for sent packets.	-
lsp-manager control-information	Enables the debugging function for control information exchanged between the LDP session management module and LDP LSP manager.	-
neighbor-manager control-information	Enables the debugging function for control information exchanged between the LDP session management module and LDP peer management module.	-
all	Disables all debugging functions.	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
mpls-ldp	debug

Usage Guidelines

To view LDP session debugging information, run the **debugging mpls ldp session** command. The debugging information helps check whether the packet format is correct and whether the packets are transmitted properly if LDP session negotiation fails.

Example

# Enable event debugging of the LDP session module.

```
<HUAWEI> debugging mpls ldp session events
```

# Enable partner debugging of the LDP session module.

```
<HUAWEI> debugging mpls ldp session partner
```

# Enable peer debugging of the LDP session module.

```
<HUAWEI> debugging mpls ldp session peer
```

# Enable the socket debugging of the LDP session module.

```
<HUAWEI> debugging mpls ldp session socket packet
```

2.13 VPN Debugging Commands

2.13.1 GRE Debugging Commands

2.13.1.1 debugging gre

Function

The **debugging gre** command enables debugging of a GRE tunnel.

The **undo debugging gre** command disables debugging of a GRE tunnel.

By default, the debugging of a GRE tunnel is disabled.

Format

debugging gre { **all** | **keepalive** | **packet** } [**interface** *interface-type interface-number*]

undo debugging gre { **all** | **keepalive** | **packet** } [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
all	Enables all debugging functions for a GRE tunnel.	-
packet	Enables debugging of GRE sent and received packets.	-
interface *interface-type interface-number*	Specifies the type and number of an interface.	The tunnel interface must already exist.

Views

User view

Default Level

3: Management Level

Task Name and Operations

Task Name	Operations
gre	debug

Usage Guidelines

To help identify a GRE tunnel problem, run the **debugging gre** command to enable debugging of a GRE tunnel.

Example

# Enable debugging of a GRE tunnel.

```
<HUAWEI> debugging gre packet interface tunnel 1
```

2.13.1.2 display debugging gre

Function

The **display debugging gre** command displays information about the enabled GRE debugging functions.

Format

display debugging gre

Parameters

None

Views

All views

Default Level

1: Monitoring level

Task Name and Operations

Task Name	Operations
gre	read

Usage Guidelines

When a large amount of information is output, the **display debugging gre** command can be used to display information about the enabled GREdebugging functions. Based on the command output, you can disable some unnecessary debugging functions to minimize the debugging information output.

Example

# View information about the enabled debugging functions.

```
<HUAWEI> display debugging gre
GRE packet debugging switch is on
GRE keepalive debugging switch is on
```

Table 2-53 Description of the **display debugging gre** command output

Items	Description
GRE packet debugging switch is on	The debugging function has been enabled for GRE packets.

Items	Description
GRE keepalive debugging switch is on	The debugging function has been enabled for the GRE Keepalive function.

2.13.2 Tunnel Management Debugging Commands

2.13.2.1 debugging tnlm

Function

The **debugging tnlm** command enables debugging of tunnel management.

The **undo debugging tnlm** command disables debugging of tunnel management.

Format

debugging tnlm { **all** | **backup** | **consumer** | **download** | **event** | **producer** }

undo debugging tnlm { **all** | **backup** | **consumer** | **download** | **event** | **producer** }

Parameters

Parameter	Description	Value
all	Outputs information about all processes.	-
backup	Outputs information about the process related to backup.	-
consumer	Outputs information about the process related to tunnel consumer.	-
download	Outputs information about the process related to tunnel advertisement.	-
event	Outputs information about the process related to the events defined in the tunnel management component.	-
producer	Outputs information about the process related to tunnel producer.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging tnlm** command to enable debugging of tunnel management so that you can view the internal processes of the tunnel management component. The process information is output on the screen.

Different debugging functions are associated with different processes. The available debugging options are as follows:

(1) backup: outputs information about the process related to backup. When backup cannot be completed or data is inconsistent, enable the option to perform analysis.

(2) consumer: outputs information about the process related to tunnel consumer. When a tunnel consumer cannot subscribe to tunnels correctly, enable the option to perform analysis.

(3) download: outputs information about the process related to tunnel advertisement. When the status of the tunnel interface is incorrect, enable the option to perform analysis.

(4) event: outputs information about the process related to the events defined in the tunnel management component. This option is used with other options. Event debugging can be enabled when other debugging functions are enabled.

(5) producer: outputs information about the process related to tunnel producer. When a tunnel exists but the tunnel information does not exist in the tunnel management component, enable the option to perform analysis.

(6) all: outputs information about all processes. When you cannot determine which debugging option will be enabled or the problem cause cannot be determined, enable this debugging option to perform analysis.

Example

# Enable all debugging functions of tunnel management.

```
<HUAWEI> debugging tnlm all
```

2.13.3 VPLS Debugging Commands

NOTE

Only the CE8868EI, CE8861EI, CE8850EI, CE8860EI, CE7850EI, CE7855EI, CE6870EI, CE6875EI, CE6860EI, CE6865EI, CE6857EI, CE6856HI, CE6855HI, CE6850U-HI, CE6851HI and CE6850HI switches support VPLS.

2.13.3.1 debugging mpls l2vpn service

Function

The **debugging mpls l2vpn service** command enables debugging for L2VPN services.

Format

debugging mpls l2vpn service vsi [*vsi-name* [*peer-address vc-id*

undo debugging mpls l2vpn service vsi [*vsi-name* [*peer-address vc-id*

Parameters

Parameter	Description	Value
vsi	Indicates VPLS debugging functions of the current module.	-
vsi-name	Specifies the name of a VSI.	The name of a VSI must already exist.
peer-address	Specifies the peer IP address of the PW.	The value is in dotted decimal notation.
vc-id	Specifies the LSR ID of the remote device on the PW.	The value is an integer ranging from 1 to 4294967295, in bytes.

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
vpws	debug

Usage Guidelines

Usage Scenario

You can enable debugging to check the information about L2VPN services.

Precautions

The debugging information is output on the screen. For less impact on performance, do not ouput too much non-debugging information.

Example

# Enable debugging of VPLS services.

```
<HUAWEI> debugging mpls l2vpn service vsi
```

2.13.3.2 debugging mpls l2vpn

Function

The **debugging mpls l2vpn** command enables debugging for L2VPN components.

The **undo debugging mpls l2vpn** command disables debugging for L2VPN components.

By default, L2VPN components debugging is disabled.

Format

debugging mpls l2vpn { **all** | **download** | **error** | **event** | **ha** | **signaling** | **timer** | **message-merge** }

undo debugging mpls l2vpn { **all** | **download** | **error** | **event** | **ha** | **signaling** | **timer** | **message-merge** }

Parameters

Parameter	Description	Value
all	All debugging functions of the current module	-
download	Debugging of a two-layer VC based on MPLS	-
error	Debugging for reporting faults between modules	-
event	Debugging for event reports between modules	-
ha	Debugging for HA	-
signaling	Debugging for signaling information	-
timer	Debugging for reports from a timer on a module	-
message-merge	Debugging for message merging	-

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
vpws	debug

Usage Guidelines

Usage Scenario

You can enable debugging to check the information about L2VPN components.

Precautions

The debugging information is output on the screen. For less impact on performance, do not ouput too much non-debugging information.

Example

# Enable all debugging of L2VPN components.

```
<HUAWEI> debugging mpls l2vpn all

2011-08-11 07:10:55 HUAWEI %%01L2VPNCOMM/3/DEBUG_INFO(d):CID=2155751297;PWE3 [LDP TNL
TIMER] Processing for LDP VC(vcid=1000,vctype=5)...
2011-08-11 07:23:54 HUAWEI %%01L2VPNCOMM/3/DEBUG_INFO(d):CID=2155751297;Recv VRP message,
Intf:3, SubIntf:2, MsgType:18(MSG_TNLMI_SUBSCRIBE_UPDATE), MsgLen:100
```

Table 2-54 Description of the **debugging mpls l2vpn all** Command Output

Item	Description
Intf	Indicates the interface type.
SubIntf	Indicates the layer 2 sub-interface type.
MsgType	Indicates the message type.
MsgLen	Indicates the length of a message.

2.14 DCN and Server Management Debugging Commands

2.14.1 TRILL Debugging commands

Function

NOTE

CE8861EI, CE8868EI, CE6880EI, CE6863, CE6863K, CE6881E, CE6820, CE6881, CE6881K, CE6865EI, CE6857EI, CE5880EI, CE5810EI, and CE6810LI does not support TRILL feature.

2.14.1.1 debugging trill adjacency

Function

The **debugging trill adjacency** command enables debugging of TRILL adjacency information.

The **undo debugging trill adjacency** command disables debugging of TRILL adjacency information.

By default, debugging of TRILL adjacency information is disabled.

Format

debugging trill adjacency [**interface** *interface-type interface-number*]

undo debugging trill adjacency [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill adjacency** command enables debugging of TRILL adjacency information, which helps you locate the fault.

Example

# Enables debugging of TRILL adjacency information for 10GE 3/0/1.

```
<HUAWEI> debugging trill adjacency interface 10ge3/0/1
May 28 2012 06:27:58.086 HUAWEI %%01TRILL/6/RX_LAN_IIH_TRILL(d):CID=0x8089041a;TRILL-ADJ:
Received Lan Level-1 IIH. (IfName=10GE3/0/1, RemoteSnpa=36.03.a6.21.12.20)
May 28 2012 06:27:59.056 HUAWEI %%01TRILL/6/TX_LAN_IIH_TRILL(d):CID=0x8089041a;TRILL-ADJ:
Sending Lan Level-1 IIH. (IfName=10GE3/0/1, LocalSnpa=36.03.a6.11.12.20)
```

Table 2-55 Description of the **debugging trill adjacency** command output

Item	Description
IfName	Interface name
LocalSnpa	MAC address of the local interface
RemoteSnpa	MAC address of the remote interface

2.14.1.2 debugging trill appointed-forwarder

Function

The **debugging trill appointed-forwarder** command enables debugging of appointed forwarder (AF) information.

The **debugging trill appointed-forwarder** command disables debugging of AF information.

By default, debugging of AF information is disabled.

Format

debugging trill appointed-forwarder [**interface** *interface-type interface-number*]

undo debugging trill appointed-forwarder [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill appointed-forwarder** command enables debugging of AF information, which helps you locate the fault.

Example

# Enable debugging of AF information.

```
<HUAWEI> debugging trill appointed-forwarder
May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_DESC_TRILL(d):CID=0x8089041a;TRILL-AF:
update AF information of 10GE3/0/1

May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_TRILL(d):CID=0x8089041a;TRILL-AF: 0020
0021 0022 0023

May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_TRILL(d):CID=0x8089041a;TRILL-AF: 0024
0025 0026 0027 0028 0029 0030 0031

May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_TRILL(d):CID=0x8089041a;TRILL-AF: 0032
0033 0034 0035 0036 0037 0038 0039

May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_TRILL(d):CID=0x8089041a;TRILL-AF: 0040
0041 0042 0043 0044 0045 0046 0047

May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_TRILL(d):CID=0x8089041a;TRILL-AF: 0048
0049 0050 0051 0052 0053 0054 0055

May 28 2012 06:54:43.367 HUAWEI %%01TRILL/6/INM_AF_UPT_TRILL(d):CID=0x8089041a;TRILL-AF: 0056
0057 0058 0059 0060 0061 0062 0063
```

2.14.1.3 debugging trill cmt-event

Function

The **debugging trill cmt-event** command enables coordinated multicast tree (CMT) debugging in a TRILL dual-homing access scenario.

The **undo debugging trill cmt-event** command disables CMT debugging in a TRILL dual-homing access scenario.

By default, CMT debugging is disabled in a TRILL dual-homing access scenario.

Format

debugging trill cmt-event

undo debugging trill cmt-event

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
trill	debug

Usage Guidelines

To enable CMT debugging in a TRILL dual-homing access scenario, run the **debugging trill cmt-event** command. The command output helps locate faults.

Example

# Enable CMT debugging in a TRILL dual-homing access scenario.

```
<HUAWEI> debugging trill cmt-event
Jan 6 2014 20:05:34.496 HUAWEI %%01TRILL/6/ENTER_ACTIVE_STATE_TRILL(d):CID=0x8086043f;TRILL-
ACTIVE: Trill enter active-active state. (PeerNickname=200, PseudoNickname=300, Priority=128, TreeId=0)
Jan 6 2014 20:05:34.496 HUAWEI %%01TRILL/6/LEAVE_ACTIVE_STATE_TRILL(d):CID=0x8086043f;TRILL-
ACTIVE: Trill leave active-active state. (PeerNickname=200, LeaveReason=6)
```

2.14.1.4 debugging trill circuit-information

Function

The **debugging trill circuit-information** command enables debugging of TRILL-capable interface information.

The **undo debugging trill circuit-information** command disables debugging of TRILL-capable interface information.

By default, debugging of TRILL-capable interface information is disabled.

Format

debugging trill circuit-information [**interface** *interface-type interface-number*]

undo debugging trill circuit-information [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill circuit-information** command enables debugging of TRILL-capable interface information, which helps you locate the fault.

Example

# Enable the debugging of TRILL-capable interface information.

```
<HUAWEI> debugging trill circuit-information
May 28 2012 06:41:55.151 HUAWEI %%01TRILL/7/DESTROY_SOCKET_TRILL(d):CID=0x8089041a;TRILL-
CIRC: Destroy socket. (IfName=10GE3/0/1, SocketID=1)
May 28 2012 06:41:55.171 HUAWEI %%01TRILL/7/CIRC_STATE_UP_TRILL(d):CID=0x8086041b;TRILL-CIRC:
The state of circuit is up. (IfName=10GE3/0/1, AddrType=TRILL)
May 28 2012 06:41:55.171 HUAWEI %%01TRILL/6/CIRC_LINK_UP_TRILL(d):CID=0x8086041b;TRILL-CIRC:
Circuit TRILL  link state change from down to up. (IfName=10GE3/0/1, OldCircState=84, NewCircState=127)
May 28 2012 06:41:55.171 HUAWEI %%01TRILL/7/CIRC_CHANGE_STATE_TRILL(d):CID=0x8089041a;TRILL-
CIRC: Circ change state. (oldState=0, newState=2)
```

Table 2-56 Description of the **debugging trill circuit-information** command output

Item	Description
IfName	Interface name
SocketID	Socket ID
AddrType	IP address type
OldCircState	Original state of link attributes on the interface
NewCircState	New state of link attributes on the interface
oldState	Original state of the logic on the interface
newState	New state of the logic on the interface

2.14.1.5 debugging trill mspf-event

Function

The **debugging trill mspf-event** command enables debugging of multicast SPF events.

The **undo debugging trill mspf-event** command disables debugging of multicast SPF events.

By default, debugging of multicast SPF events is disabled.

Format

debugging trill mspf-event

undo debugging trill mspf-event

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill mspf-event** command enables debugging of multicast route calculation events, which helps you locate the fault.

Example

# Enable debugging of multicast route calculation events.

```
<HUAWEI> debugging trill mspf-event
May 28 2012 07:48:03.176 HUAWEI %%01TRILL/6/MSPF_DTREE_NUM_TRILL(d):CID=0x809f0419;TRILL-
MSPF: number of dtrees to be calculated is 2
May 28 2012 07:48:03.176 HUAWEI %%01TRILL/6/MSPF_PREP_DTREE_TRILL(d):CID=0x809f0419;TRILL-
MSPF: prepared dtree to compute.(nickname=222)
May 28 2012 07:48:03.176 HUAWEI %%01TRILL/6/MSPF_DTREE_CAL_TRILL(d):CID=0x809f0419;TRILL-
MSPF: start calculate dtree.(TreeNickname=222, TreeNum=1)
May 28 2012 07:48:03.176 HUAWEI %%01TRILL/6/
MSPF_DTREE_ADD_PARENT_TRILL(d):CID=0x809f0419;TRILL-MSPF: dtree parent node added.
(parent=3603.a621.1220.00, child=3603.a611.1220.00)
May 28 2012 07:48:07.177 HUAWEI %%01TRILL/6/MSPF_CRT_MCRT_TRILL(d):CID=0x809f0419;TRILL-MSPF:
create multicast route entry.(tree=222, vlan=127)
May 28 2012 07:48:07.177 HUAWEI %%01TRILL/6/MSPF_ADD_MCRT_NH_TRILL(d):CID=0x809f0419;TRILL-
MSPF: add nexthop of multicast route entry.(tree=222, vlan=127, ifindex=5, OutVlan=11)
```

Table 2-57 Description of the **debugging trill mspf-event** command output

Item	Description
nickname	RB nickname
TreeNickname	Multicast root nickname
TreeNum	Number of the MDT
parent	Parent node
child	Child node
tree	MDT
vlan	VLAN to which a node belongs
ifindex	Interface index
OutVlan	Outer VLAN ID

2.14.1.6 debugging trill receiving-packet-content

Function

The **debugging trill receiving-packet-content** command enables debugging of received hexadecimal TRILL packets.

The **undo debugging trill receiving-packet-content** command disables debugging of received hexadecimal TRILL packets.

By default, debugging of received hexadecimal TRILL packets is disabled.

Format

debugging trill receiving-packet-content [**interface** *interface-type interface-number*]

undo debugging trill receiving-packet-content [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill receiving-packet-content** command enables debugging of received hexadecimal Hello packets or LSPs, which helps you locate the fault.

The **debugging trill receiving-packet-content** command must be run with the **debugging trill adjacency** or **debugging trill update-packet** command. If you run the **debugging trill receiving-packet-content** and **debugging trill adjacency** commands, debugging information about received Hello packets is displayed. If you run the **debugging trill receiving-packet-content** and **debugging trill update-packet** commands, debugging information about received LSPs and SNPs is displayed.

Example

# Enable debugging of received hexadecimal TRILL packets and adjacency information.

```
<HUAWEI> debugging trill receiving-packet-content
<HUAWEI> debugging trill adjacency
May 28 2012 06:47:00.096 HUAWEI %%01TRILL/6/RX_LAN_IIH_TRILL(d):CID=0x8089041a;TRILL-ADJ:
Received Lan Level-1 IIH. (IfName=Ethernet3/0/1, RemoteSnpa=36.03.a6.21.12.20)
May 28 2012 06:47:00.096 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0010 :03 83 1b
01 06 0f 01 00 01 01 36 03 a6 21 12 20
May 28 2012 06:47:00.096 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0020 :00 0a 02
00 40 36 03 a6 21 12 20 01 01 02 01 00
May 28 2012 06:47:00.096 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0030 :81 01 c0
8f 0c 00 00 01 08 00 01 00 de 10 0b 00
May 28 2012 06:47:00.096 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0040 :0b 8f ff
00 00 02 fb 00 64 ff ff ff ff ff ff ff
May 28 2012 06:47:00.096 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0050 :ff ff ff ff
ff ff ff ff ff ff ff ff ff ff ff ff
```

Table 2-58 Description of the **debugging trill receiving-packet-content** and **debugging trill adjacency** command output

Item	Description
IfName	Interface name
RemoteSnpa	MAC address of the remote end

2.14.1.7 debugging trill self-originate-update

Function

The **debugging trill self-originate-update** command enables debugging of locally generated update TRILL LSPs.

The **undo debugging trill self-originate-updatee** command disables debugging of locally generated update TRILL LSPs.

By default, debugging of locally generated update TRILL LSPs is disabled.

Format

debugging trill self-originate-update

undo debugging trill self-originate-update

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill self-originate-update** command enables debugging of locally generated update TRILL LSPs, which helps you locate the fault in information advertising and LSP synchronizing processes.

Example

# Enable the debugging of locally generated update TRILL LSPs.

```
<HUAWEI> debugging trill self-originate-update
May 28 2012 07:53:40.009 HUAWEI %%01TRILL/7/
DEL_NBR_OPTION_FROM_LSP_TRILL(d):CID=0x8086041b;TRILL-UPDT: Delete neighbour option from LSP.
(TlvType=22, Level=1, NbrId=3603.A621.1220.00)
```

```
May 28 2012 07:53:40.559 HUAWEI %%01TRILL/7/
ADD_NBR_OPTION_IN_LSP_TRILL(d):CID=0x8086041b;TRILL-UPDT: Add neighbour option in LSP.
(TlvType=22, Level=1, NbrId=3603.A621.1220.00)
May 28 2012 07:53:41.489 HUAWEI %%01TRILL/7/
SELF_LSP_TIMER_EXPIRE_TRILL(d):CID=0x8086041b;TRILL-UPDT: Lsp generation Intelligent timer expired.
(Level=1)
```

Table 2-59 Description of the **debugging trill self-originate-update** command output

Item	Description
TlvType	TLV type
Level	TRILL level
NbrId	System ID of the neighbor

2.14.1.8 debugging trill sending-packet-content

Function

The **debugging trill sending-packet-content** command enables debugging of sent hexadecimal TRILL packets.

The **undo debugging trill sending-packet-content** command disables debugging of sent hexadecimal TRILL packets.

By default, debugging of sent hexadecimal TRILL packets is disabled.

Format

debugging trill sending-packet-content [**interface** *interface-type interface-number*]

undo debugging trill sending-packet-content [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill sending-packet-content** command enables debugging of sent hexadecimal TRILL packets. If you run the **debugging trill sending-packet-content** and **debugging trill adjacency** commands, debugging information about sent Hello packets is displayed. If you run the **debugging trill update-packet** and **debugging trill sending-packet-content** commands, debugging information about sent LSPs and SNPs is displayed.

Example

# Enable debugging of sent hexadecimal TRILL packets and adjacency information.

```
<HUAWEI> debugging trill sending-packet-content
<HUAWEI> debugging trill adjacency
```

# Enable debugging of sent hexadecimal Hello packet on broadcast networks.

```
May 28 2012 07:57:28.827 HUAWEI %%01TRILL/6/TX_LAN_IIH_TRILL(d):CID=0x8089041a;TRILL-ADJ:
Sending Lan Level-1 IIH. (IfName=Ethernet3/0/1, LocalSnpa=36.03.a6.11.12.20)
May 28 2012 07:57:28.827 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0010 :03 83 1b
01 06 0f 01 00 01 01 36 03 a6 11 12 20
May 28 2012 07:57:28.827 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0020 :00 1e 01
65 40 36 03 a6 21 12 20 01 01 02 01 00
May 28 2012 07:57:28.827 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0030 :81 01 c0
8f 0c 00 00 01 08 00 01 00 6f 10 0b 00
May 28 2012 07:57:28.827 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0040 :0b 8f fe
00 00 02 fa 00 14 ff ff ff ff ff ff ff
May 28 2012 07:57:28.827 HUAWEI %%01TRILL/6/IS_PDU_TRILL(d):CID=0x8089041a;TRILL-0050 :ff ff ff ff
ff ff ff ff ff ff ff ff ff ff ff ff
```

Table 2-60 Description of the **debugging trill sending-packet-content** and **debugging trill adjacency** command output

Item	Description
IfName	Interface name
LocalSnpa	Local SNPA address

2.14.1.9 debugging trill snp-packet

Function

The **debugging trill snp-packet** command enables debugging of TRILL SNPs.

The **undo debugging trill snp-packet** command disables debugging of TRILL SNPs.

By default, debugging of TRILL SNPs is disabled.

Format

debugging trill snp-packet [**interface** *interface-type interface-number*]

undo debugging trill snp-packet [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill snp-packet** command enables debugging of TRILL SNPs, including information about receiving and sending CSNPs and PSNPs, which helps you locate the fault in TRILL LSDB synchronization.

Example

# Enable debugging of TRILL SNPs.

```
<HUAWEI> debugging trill snp-packet
```

The local interface received PSNPs.

```
May 28 2012 08:01:04.077 HUAWEI %%01TRILL/7/RECV_SNP_FROM_CIRC_TRILL(d):CID=0x8086041b;TRILL-
RECV: Receive CSNP from circuit. (IfName=10GE3/0/1, Level=1)
May 28 2012 08:01:04.887 HUAWEI %%01TRILL/6/SEND_SNP_OK_TRILL(d):CID=0x8086041b;TRILL-SNP:
Succeed to send PSNP on circuit. (IfName=10GE3/0/1, Level=1)
```

Table 2-61 Description of the **debugging trill snp-packet** command output

Item	Description
IfName	Interface name
Level	Level of received and sent packets

2.14.1.10 debugging trill spf-event

Function

The **debugging trill spf-event** command enables debugging of TRILL SPF events.

The **undo debugging trill spf-event** command disables debugging of TRILL SPF events.

By default, debugging of TRILL SPF events is disabled.

Format

debugging trill spf-event

undo debugging trill spf-event

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill spf-event** command enables debugging of TRILL SPF events, which helps you locate the fault in unicast route calculation.

Example

# Enable debugging of TRILL SPF events.

```
<HUAWEI> debugging trill spf-event
May 28 2012 08:03:47.081 HUAWEI %%01TRILL/6/SPF_CAL_INFO_TRILL(d):CID=0x809f0419;TRILL-
CALCULATE-PHASE: Full SPF calculation started.
May 28 2012 08:03:47.081 HUAWEI %%01TRILL/6/SPF_CREATE_NODE_TRILL(d):CID=0x809f0419;TRILL-
CAL: Create a node 3603.a611.1220.00.
May 28 2012 08:03:47.081 HUAWEI %%01TRILL/6/SPF_CREATE_NODE_TRILL(d):CID=0x809f0419;TRILL-
CAL: Create a node 3603.a621.1220.00.
May 28 2012 08:03:47.081 HUAWEI %%01TRILL/6/SPF_CAL_NH_TRILL(d):CID=0x809f0419;TRILL-SPF:
calculate nexthop for 3603.a621.1220.00.
May 28 2012 08:03:47.081 HUAWEI %%01TRILL/6/SPF_CRT_NH_TRILL(d):CID=0x809f0419;TRILL-SPF: create
nexthop for 3603.a621.1220.00 success.(ifindex=5, neighbor=3603.a621.1220)
May 28 2012 08:03:47.081 HUAWEI %%01TRILL/6/SPF_ADD_TENT_NODE_TRILL(d):CID=0x809f0419;TRILL-
SPF: node added to tentlist.(node=3603.a621.1220.00, distance=200000)
```

Table 2-62 Description of the **debugging trill spf-event** command output

Item	Description
ifindex	Interface index
neighbor	System ID of the neighbor
node	System ID of the RB that is performing the SPF calculation
distance	Cost of the route from the RB that is performing the SPF calculation to the local RB

2.14.1.11 debugging trill trace

Function

The **debugging trill trace** command enables debugging of TRILL trace information.

The **undo debugging trill trace** command disables debugging of TRILL trace information.

By default, debugging of TRILL trace information is disabled.

Format

debugging trill trace

undo debugging trill trace

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

To enable debugging of TRILL trace information, run the **debugging trill trace** command. The command output helps you check link connectivity and locate faults on a TRILL network.

Example

# Enable debugging of TRILL trace information for 10GE 3/0/1.

```
<HUAWEI> debugging trill trace
[2014-11-14 10:56:35:886][TRILLOAM]: TRILLOAM Utrace Start TransNumber:0X800054AB)

Nov 14 2014 10:56:35.901 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:35:901][TRILLOAM]: TRILLOAM Utrace Start
Para:Count=5,MaxTtl=63,Interval=2000,Timeout=2000,NickName=1003,SourceMac=00-00-00,DestMac=00-00
-00,SourceIp=0x0,DestIp=0x0,SourcePort=0,DstPort=0,CeVlan=0,EthType=0x0,Protocol=0,OutIfIndex=0,LogicO
utIfIndex=0,SrcIfIndex=0,LogicSrcIfIndex=0,uiVrId=0,CmdType=0x0,PortType=5,LocalNickName=1001,
TestId=0, BitMap=0x0)

Nov 14 2014 10:56:35.949 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:35:949][TRILLOAM]: TRILLOAM PKT Send OK.)

Nov 14 2014 10:56:35.949 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:35:949][TRILLOAM]: TRILLOAM UTrace Send Pkt, sndnum:1)
```

```
Nov 14 2014 10:56:35.949 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:35:949][TRILLOAM]: Create Timer(0) Success Id:0Xb3a09bc0)

Nov 14 2014 10:56:35.949 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:35:949][TRILLOAM]: TRILLOAM UTrace Create Trace Instance Success)

Nov 14 2014 10:56:36.593 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:36:593][TRILLOAM]: TRILLOAM Utrace Timeout Timer Waked)

Nov 14 2014 10:56:37.403 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:37:403][TRILLOAM]: TRILLOAM PKT Rcv Success)

Nov 14 2014 10:56:37.403 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:37:403][TRILLOAM]: TRILLOAM Utrace Recv Pkt, LoopbackTransId:0x1,
IfName:Ethernet3/0/0, ReplyNickName:1002, PreviousNickName:1001)

Nov 14 2014 10:56:37.403 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:37:403][TRILLOAM]: TRILLOAM UTrace Proc Rcv Pkt, rcvnum:1)

Nov 14 2014 10:56:37.403 HUAWEI %%01TRILLOAM/7/TRILLOAM_DEBUG(d):CID=0x80f304a1;TRILLOAM is
on.(
[2014-11-14 10:56:37:403][TRILLOAM]: TRILLOAM Utrace Check Test End, sendcount:1, recvcount:1,
timeoutcount:0, pkttl:1, cmdttl:63)
```

Table 2-63 Description of the **debugging trill trace** command output

Item	Description
Count	TTL value of a sent TRILL packet
MaxTtl	Maximum TTL value
Interval	Interval at which Echo Request packets are sent
Timeout	Timer for a device to receive an Echo Reply packet in response to an Echo Request
NickName	Nickname of the destination node
SourceMac	Source MAC address carried in a data packet
DestMac	Destination MAC address carried in a data packet
SourceIp	Source IP address carried in a data packet
DestIp	Destination IP address carried in a data packet
SourcePort	Source port number carried in a data packet

Item	Description
DstPort	Destination port number carried in a data packet
CeVlan	CE VLAN ID carried in a data packet
EthType	Ethernet type carried in a data packet
Protocol	Protocol type carried in a data packet
OutIfIndex	Physical outbound interface that sends a TRILL packet
LogicOutIfIndex	Logical outbound interface that sends a TRILL packet
SrcIfIndex	Index of the source physical interface carried in a TRILL packet
LogicSrcIfIndex	Index of the logical interface carried in a TRILL packet
uiVrId	Local VS ID
CmdType	Command type
PortType	Port type of a TRILL interface
LocalNickName	Local nickname
TestId	ID of a test instance
BitMap	Bit map
LoopbackTransId	Transmission ID of a TRILL packet
IfName	Name of the inbound interface of a TRILL packet
ReplyNickName	Nickname replied with by the remote end
PreviousNickName	Nickname of the previous hop
sendcount	Number of sent TRILL packets
recvcount	Number of received TRILL packets
timeoutcount	Number of times a device fails to receive an Echo Reply packet in response to an Echo Request packet within a timer
pkttl	TTL value carried in an Echo Reply packet
cmdttl	TTL value specified in a command

2.14.1.12 debugging trill update-packet

Function

The **debugging trill update-packet** command enables debugging of TRILL update LSPs and SNPs.

The **undo debugging trill update-packet** command disables debugging of TRILL update LSPs and SNPs.

By default, debugging of TRILL update LSPs and SNPs is disabled.

Format

debugging trill update-packet [**interface** *interface-type interface-number*]

undo debugging trill update-packet [**interface** *interface-type interface-number*]

Parameters

Parameter	Description	Value
interface *interface-type interface-number*	Specifies the type and number of an interface.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill update-packet** command enables debugging of TRILL update LSPs and SNPs, including information about the processing of received LSPs and SNPs, which helps you locate the fault in TRILL LSDB synchronization.

Example

# Enable debugging of TRILL update LSPs and SNPs.

```
<HUAWEI> debugging trill update-packet
May 28 2012 08:09:15.920 HUAWEI %%01TRILL/6/LSP_CLEAR_TRILL(d):CID=0x8086041b;TRILL-LSP:
Succeed to clear LSP information. (LspId=3603.a621.1220.00-01, Level=1)
May 28 2012 08:09:15.920 HUAWEI %%01TRILL/7/LSP_ADD_DESC_TRILL(d):CID=0x8086041b;TRILL-LSP:
Add LSP desc to LSP set. (LspId=3603.a611.1220.00-00)
May 28 2012 08:09:15.920 HUAWEI %%01TRILL/7/LSP_ADD_OPT_GROUP_TRILL(d):CID=0x8086041b;TRILL-
LSP: Add option group to LSP desc. (TlvType=1)
May 28 2012 08:09:15.920 HUAWEI %%01TRILL/7/LSP_ADD_OPT_TRILL(d):CID=0x8086041b;TRILL-LSP: Add
option to option list in option group. (TlvType=1)
May 28 2012 08:09:16.430 HUAWEI %%01TRILL/7/LSP_ADD_OPT_GROUP_TRILL(d):CID=0x8086041b;TRILL-
LSP: Add option group to LSP desc. (TlvType=22)
May 28 2012 08:09:16.430 HUAWEI %%01TRILL/7/LSP_ADD_OPT_TRILL(d):CID=0x8086041b;TRILL-LSP: Add
option to option list in option group. (TlvType=22)
```

```
May 28 2012 08:09:17.380 HUAWEI %%01TRILL/7/LSP_SET_SRM_TRILL(d):CID=0x8086041b;TRILL-LSP: Set
SRM flag. (LspId=3603.a611.1220.00-00, IfName=10GE3/0/1)
```

Table 2-64 Description of the **debugging trill update-packet** command output

Item	Description
LspId	LSP ID
Level	Level of TRILL packets
TlvType	TLV type
IfName	Interface name

2.14.1.13 debugging trill update-process

Function

The **debugging trill update-process** command enables debugging of the processing of received TRILL update LSPs.

The **undo debugging trill update-process** command disables debugging of the processing of received TRILL update LSPs.

By default, debugging of the processing of received TRILL update LSPs is disabled.

Format

debugging trill update-process

undo debugging trill update-process

Parameters

None

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging trill update-process** command enables debugging of the processing of received TRILL update LSPs, which helps you locate the fault in TRILL LSDB synchronization.

Example

# Enable debugging of the processing of received TRILL update LSPs.

```
<HUAWEI> debugging trill update-process
May 28 2012 08:11:35.851 HUAWEI %%01TRILL/7/
MINLSPGEN_TMR_NOT_EXPIRED_TRILL(d):CID=0x8086041b;TRILL-UPDT: MinLspGen Timer has not expired.
Not generating LSP.
May 28 2012 08:11:36.509 HUAWEI %%01TRILL/7/GEN_NEWER_LSP_TRILL(d):CID=0x8086041b;TRILL-
UPDT: Prev incarnation LSP, generating newer LSP.
May 28 2012 08:11:36.639 HUAWEI %%01TRILL/6/LSP_AREA_PRS_OK_TRILL(d):CID=0x8086041b;TRILL-
UPDT: Succeed to parse area address 00.(LspId=3603.a621.1220.00-00)
May 28 2012 08:11:36.639 HUAWEI %%01TRILL/6/LSP_EXTNBR_OK_TRILL(d):CID=0x8086041b;TRILL-UPDT:
Succeed to parse LSP  extended neighbor. (LspId=3603.a621.1220.00-00, Neighbor=3603.A611.1220.00)
May 28 2012 08:11:36.639 HUAWEI %%01TRILL/7/
CREATE_SYSTEM_INFO_NODE_TRILL(d):CID=0x8086041b;TRILL-UPDT: System Info Node 3603.A621.1220.00
is created.
May 28 2012 08:11:36.639 HUAWEI %%01TRILL/7/INS_LSP_TO_SYS_NODE_TRILL(d):CID=0x8086041b;TRILL-
UPDT: LSP is installed to system info node. (SysId=3603.A621.1220, PseudoId=0, FragNum=0)
May 28 2012 08:11:36.639 HUAWEI %%01TRILL/6/LSP_NBR_UPDT_TRILL(d):CID=0x8086041b;TRILL-UPDT:
Update Neighbor 3603.A611.1220.00 to SPF.
May 28 2012 08:11:40.656 HUAWEI %%01TRILL/6/LSP_AREA_PRS_OK_TRILL(d):CID=0x8086041b;TRILL-
UPDT: Succeed to parse area address 00.(LspId=3603.a621.1220.00-00)
May 28 2012 08:11:40.656 HUAWEI %%01TRILL/6/LSP_EXTNBR_OK_TRILL(d):CID=0x8086041b;TRILL-UPDT:
Succeed to parse LSP  extended neighbor. (LspId=3603.a621.1220.00-00, Neighbor=3603.A611.1220.00)
May 28 2012 08:11:40.656 HUAWEI %%01TRILL/7/INS_LSP_TO_SYS_NODE_TRILL(d):CID=0x8086041b;TRILL-
UPDT: LSP is installed to system info node. (SysId=3603.A621.1220, PseudoId=0, FragNum=1)
```

Table 2-65 Description of the **debugging trill update-process** command output

Item	Description
LspId	ID of the received LSP
Neighbor	System ID of the neighbor
SysId	System ID of the local RB
PseudoId	Pseudonode ID
FragNum	LSP fragment number

2.14.1.14 undo debugging trill all

Function

The **undo debugging trill all** command disables all debugging functions of TRILL.

Format

undo debugging trill all

Parameters

None

Views

User view

Default Level

3: Management level

Task Name and Operations

Task Name	Operations
trill	debug

Usage Guidelines

You can run the **undo debugging trill all** command to disable all debugging functions at a time, instead of disabling these functions one by one.

Example

# Disable all debugging functions of TRILL.

```
<HUAWEI> undo debugging trill all
```

2.14.2 FCoE Debugging Commands

NOTE

CE6810LI does not support FCoE forwarder (FCF) or NPort Virtualization (NPV) function. CE6870EI and CE6875EI do not support NPV function. Only CE8860EI, CE8861EI, CE8861P, and CE6850U-HI support FC interfaces. CE6880EI, CE6863, CE6863K, CE6881E, CE6820, CE6881, CE6881K, and CE5800 do not support this feature.

2.14.2.1 debugging fcoe

Function

The **debugging fcoe** command enables FCoE debugging.

The **undo debugging fcoe** command disables FCoE debugging.

By default, FCoE debugging is disabled.

Format

debugging fcoe packet

undo debugging fcoe packet

Parameters

Parameter	Description	Value
packet	Enables debugging of FIP packets.	-

Views

User view

Default Level

3: Management level

Usage Guidelines

The **debugging fcoe** command enables FCoE debugging. You can use this command to view debugging information about all FC instances, including source and destination IP addresses, VLAN IDs, and types of FIP packets. The command output helps you locate FCoE faults.

Example

# Enable FCoE debugging for all FC instances.

```
<HUAWEI> debugging fcoe packet
```

2.15 SFC Debugging Commands

2.15.1 debugging service-chain

Function

The **debugging service-chain** command enables key SFC process debugging.

The **undo debugging service-chain** command disables key SFC process debugging.

By default, key SFC process debugging is disabled.

Format

debugging service-chain { **all** | **rm** | **fes** | **info** | **error** | **event** }

undo debugging service-chain { **all** | **rm** | **fes** | **info** | **error** | **event** }

Parameters

Parameter	Description	Value
all	Indicates debugging for all SFC components.	-
rm	Indicates RM debugging.	-
fes	Indicates FES debugging.	-

Parameter	Description	Value
info	Indicates info debugging.	-
error	Indicates error debugging.	-
event	Indicates event debugging.	-

Views

Diagnostic view

Default Level

3: Management level

Usage Guidelines

You can run the **debugging service-chain** to enable SFC debugging for fault location.

Example

# Enable RM debugging.

```
<HUAWEI> system-view
[~HUAWEI] diagnose
[~HUAWEI-diagnose] debugging service-chain rm
Dec 30 2016 16:07:00.379 HUAWEI %%01SFC/7/SFC_DEBUG(d):CID=0x810b0444; [SFC]:  Send Apply Vrf
Info, Vrf:0, send:0, ack:0
```