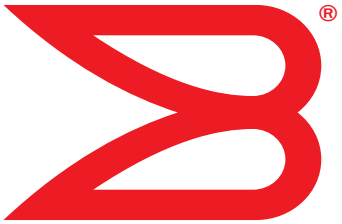
53-1003076-02
30 July 2014



Brocade FastIron SX, FCX, and ICX

Diagnostic Reference

Supporting FastIron Software Release 08.0.10d

BROCADE

Brocade Communications Systems, Incorporated

Corporate and Latin American Headquarters
Brocade Communications Systems, Inc.
130 Holger Way
San Jose, CA 95134
Tel: 1-408-333-8000
Fax: 1-408-333-8101
E-mail: info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems China HK, Ltd.
No. 1 Guanghua Road
Chao Yang District
Units 2718 and 2818
Beijing 100020, China
Tel: +8610 6588 8888
Fax: +8610 6588 9999
E-mail: china-info@brocade.com

European Headquarters
Brocade Communications Switzerland Sàrl
Centre Swissair
Tour B - 4ème étage
29, Route de l'Aéroport
Case Postale 105
CH-1215 Genève 15
Switzerland
Tel: +41 22 799 5640
Fax: +41 22 799 5641
E-mail: emea-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems Co., Ltd. (Shenzhen WFOE)
Citic Plaza
No. 233 Tian He Road North
Unit 1308 – 13th Floor
Guangzhou, China
Tel: +8620 3891 2000
Fax: +8620 3891 2111
E-mail: china-info@brocade.com

Document History

Title	Publication number	Summary of changes	Date
Brocade FastIron, FCX, ICX, and TurboIron Diagnostic Reference	53-1002645-01	Release 07.4.00 document updated with enhancements in Release 08.0.00.	April 2013
Brocade FastIron, FCX, ICX, and TurboIron Diagnostic Reference	53-1002645-02	Release 08.0.00 document updated for Release 08.0.00a.	June 2013
Brocade FastIron, FCX, ICX, and TurboIron Diagnostic Reference	53-1002965-01	Release 08.0.00a document updated for Release 08.0.01.	September 2013
Brocade FastIron SX, FCX, and ICX Diagnostic Reference	53-1003076-01	Release 08.0.01 document updated for Release 08.0.10.	January 2014
Brocade FastIron SX, FCX, and ICX Diagnostic Reference	53-1003076-02	Release 08.0.10 document updated for Release 08.0.10d.	July 2014

Contents

About This Guide

This chapter contains the following sections:

Introduction

This guide describes many common diagnostic processes for the Brocade FastIron devices. Each chapter contains diagnostic information about a specific segment of your network configuration.

Supported Hardware

- FastIron X Series devices (chassis models):
  - Brocade FastIron SX 800
  - FastIron SX 1600
- Brocade FCX Series (Brocade FCX) Stackable Switch
- Brocade ICX 6610 (ICX 6610) Stackable Switch
- Brocade ICX 6430 Series (ICX 6430)
- Brocade ICX 6450 Series (ICX 6450)
- Brocade ICX 6650 Series (ICX 6650)
- Brocade ICX 7750 Series (ICX 7750)

For information about the specific models and modules supported in a product family, refer to the hardware installation guide for that product family. "Related publications" on page x lists the hardware installation guides and software configuration guides.

NOTE
The Brocade ICX 6430-C switch supports the same feature set as the Brocade ICX 6430 switch unless otherwise noted.

NOTE
The Brocade ICX 6450-C12-PD switch supports the same feature set as the Brocade ICX 6450 switch unless otherwise noted.

NOTE
Some debug commands report information about internal hardware settings and registers that is relevant primarily to the Brocade engineering staff. Consequently, this information is not described in this document.

Unsupported features

Features that are not documented in "Related publications" on page x are not supported.

Audience

This document is designed for network engineers with a working knowledge of Layer 2 and Layer 3 switching and routing.

If you are using a Brocade Layer 3 switch, you should be familiar with the following protocols if applicable to your network – IP, RIP, OSPF, BGP, ISIS, IGMP, PIM, DVMRP, and VRRP.

Disclaimer

This manual is provided without any warranty of any kind, expressed or implied. When using this manual to troubleshoot Brocade products, you assume all risk as to the quality and performance of the debug procedures. Brocade assumes no liability for any damages, including general, special, incidental, or consequential damages arising from the use of the procedures in this manual (including, but not limited to any loss of profit or savings, loss of data, or failure to successfully troubleshoot network problems).

Debug information may be changed or updated without notice. You are responsible for obtaining newer versions of this manual when they are made available. The procedures in this document are not intended as a substitute for the expertise of qualified technicians.

Enabling debug commands can seriously degrade system performance. Debug commands are generally intended for use when troubleshooting specific problems while working with qualified service technicians, or in conjunction with calls to Brocade Technical Support. Whenever possible, troubleshoot your system during periods of low network traffic and user activity to preserve system performance.

If you have any questions regarding this disclaimer please contact us at *http://www.brocade.com/products/all/routers/index.page*.

What's new in this document

This document includes the information from IronWare software release 08.0.10d. Table 1 lists the enhancements for FastIron release 08.0.10d.

TABLE 1 Summary of Enhancements in FastIron release 08.0.10d

Feature	Description	Described in
DHCPv6 Relay Agent Prefix Delegation (PD) notification	The **debug ipv6 dhcp** command has been modified to support DHCPv6 PD debugging.	"DHCPv6 debug commands"
VXLAN debug commands	Added new debug commands for debugging the VXLAN Layer 2 tunnel configurations.	"VXLAN debug commands"

Document conventions

This section describes text formatting conventions and important notice formats used in this document.

Text formatting

The narrative-text formatting conventions that are used are as follows:

bold text	Identifies command names
	Identifies the names of user-manipulated GUI elements
	Identifies keywords
	Identifies text to enter at the GUI or CLI
italic text	Provides emphasis
	Identifies variables
	Identifies document titles
code text	Identifies CLI output

For readability, command names in the narrative portions of this guide are presented in mixed lettercase: for example, **switchShow**. In actual examples, command lettercase is all lowercase.

Command syntax conventions

Command syntax in this manual follows these conventions:

command and parameters	Commands and parameters are printed in bold.
[ ]	Optional parameter.
variable	Variables are printed in italics.
...	Repeat the previous element, for example "member[;member...]"
|	Choose from one of the parameters.

Notes and caution notices

The following notices and statements are used in this manual. They are listed below in order of increasing severity of potential hazards.

NOTE
A note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates potential damage to hardware or data.

CAUTION

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

Related publications

The following Brocade documents supplement the information in this guide and can be located at *http://www.brocade.com/ethernetproducts*.

- *FastIron Ethernet Switch Administration Guide, 08.0.10d*
- *FastIron Ethernet Switch Platform and Layer 2 Switching Configuration Guide, 08.0.10d*
- *FastIron Ethernet Switch Layer 3 Routing Configuration Guide, 08.0.10d*
- *FastIron Ethernet Switch IP Multicast Configuration Guide, 08.0.10a*
- *FastIron Ethernet Switch Security Configuration Guide, 08.0.10d*
- *FastIron Ethernet Switch Software Upgrade Guide, 08.0.10a*
- *FastIron Switch Stacking Configuration Guide, 08.0.10a*
- *FastIron Ethernet Switch Traffic Management Guide, 08.0.10b*
- *FastIron Ethernet Switch Software Licensing Guide, 08.0.10*
- *FastIron Feature Support Matrix, 08.0.10d*
- *Brocade ICX 6430-C Compact Switch Hardware Installation Guide, 08.0.10*
- *Brocade ICX 6430 and ICX 6450 Stackable Switches Hardware Installation Guide, 08.0.10*
- *Brocade FCX Series Hardware Installation Guide, 08.0.10*
- *Brocade FastIron ICX 6610 Stackable Switch Hardware Installation Guide, 08.0.10*
- *Brocade ICX 6650 Ethernet Switch Installation Guide, 08.0.10*
- *Brocade ICX 7750 Switch Hardware Installation Guide, 08.0.10*
- *Brocade FastIron SX Series Chassis Hardware Installation Guide, 08.0.10*
- *Brocade ICX 6450-C Compact Switch Hardware Installation Guide, 08.0.10*
- *Unified IP MIB Reference, 08.0.10*

To contact Technical Support, go to *http://www.brocade.com/services-support/index.page* for the latest e-mail and telephone contact information.

Document feedback

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. Forward your feedback to:

documentation@brocade.com

Provide the title and version number of the document and as much detail as possible about your comment, including the topic heading and page number and your suggestions for improvement.

Chapter

1 Using Diagnostic Commands

This chapter contains the following sections:

Using debug commands

This section describes how to use debug commands to monitor and troubleshoot Brocade FastIron, Brocade FCX, and Brocade ICX switch configurations. The debug commands are accessible from the Privileged EXEC mode in the IronWare command line interface (CLI). Most of the debug commands can be configured to send output to a specified destination.

When enabled, the debug commands can noticeably affect system performance. Many debug commands are specifically designed to be used in conjunction with calls to Brocade technical support. If you report a problem, the support engineer may ask you to execute one or more of the debug commands described in this guide.

ATTENTION
Some debug commands report information about internal hardware settings and registers, which is relevant primarily to the Brocade engineering staff. These commands are not described in this document.

Brief and detail debug options

When enabled, many debug commands can significantly impact system performance. Many debug commands provide options for brief or detailed reporting. Generating detailed output places an additional burden on system performance, and in many cases the results may be more difficult to interpret than output generated using the **brief** option. To conserve performance and prevent system disruption, use the **brief** option whenever possible.

Generic debug commands

The following generic debug commands perform functions related to all debugging actions:

- **debug ?**—Generates a list of debug options.
- [**no**] **debug all**—Enables or disables all debug functions.
- **show debug**—Shows all enabled debug settings.
- **debug destination**—Allows you to select an output destination; Telnet, SSH, console, or logging (default).

debug ?

Syntax: debug ?

This command generates a list of available debug variables.

ATTENTION
Many first-level variables have their own variable subsets. When you enter a debug command, the system indicates that there are additional variables available and you have entered an incomplete command. Add a space and a question mark to your original command to view the additional variables.

```
Brocade# debug ip
Incomplete command
Brocade#debug ip ?
  arp             ARP messages
  dhcp_snooping   DHCP snooping
  icmp            ICMP transactions
  igmp            IGMP protocol activity
  pim             PIM/dvmrp protocol activity
  source_guard    Source Guard
  ssh             SSH information
  tcp             TCP information
  udp             UDP based transactionstp
  web             WEB HTTP/HTTPS information
```

debug all

Syntax: [no] debug all

This command enables all the debug functions, and should be used *only* during a troubleshooting session with a Brocade technician.

```
Brocade# debug all
Warning! This may severely impact network performance!
All possible debuggings have been turned on
```

Enter the **no** command to cancel the setting.

```
Brocade#no debug all
Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Disabling ACL log
Don't monitor port
```

CAUTION
The debug all command generates extensive output and can significantly slow device operation. Use this command with caution. Never use this command during periods of peak network activity. Enter no debug all to stop the output.

debug destination

Syntax: [no] debug destination [console | logging | telnet *num* **| ssh** *num* **]**

- **console**—Directs output to the system console.
- **logging**—Directs output to the syslog buffer and to the syslog server (default).
- **telnet** *num*—Directs output to a specified Telnet session (ranges from 1 through 5).
- **ssh** *num*—Directs output to a specified Secure Shell (SSH) session (ranges from 1 through 5).

This command allows you to specify a destination for debugging output. The default destination is the system console, but you can redirect output to a syslog buffer, Telnet session, or SSH session.

To send debug output to a Telnet session, first determine your session number using the **show who** command.

```
Brocade# show who
Console connections (by unit number):
 1      established
        4 minutes 29 seconds in idle
Telnet connections (inbound):
 1      established, client ip address 172.31.0.1
        you are connecting to this session
        2 seconds in idle
 2      closed
 3      closed
 4      closed
 5      closed
Telnet connection (outbound):
 6      closed
SSH connections:
 1      closed
 2      closed
 3      closed
 4      closed
```

This example indicates that you are connected through active Telnet session 1. To redirect the debug output to your Telnet session, enter the following command.

```
Brocade#debug destination telnet 1
```

show debug

Syntax: show debug

This command displays all the enabled debug functions. The output resembles the following example, which shows that ACL log and IPv6 debugging are enabled, with the console as the output destination.

```
Brocade# show debug
Debug message destination:  Console
Enabling ACL log
IPv6 Routing:
ipv6:  icmp debugging is on
```

Disabling debug commands

When activated, most debug commands instruct the system to collect specific information about router configurations and activity. In all cases, adding **no** in front of the command disables the debug function.

Chapter

System Level and Layer 1 Diagnostics 2

This chapter contains the following sections:

Layer 1 debug commands

The following commands deal with the Layer 1 debugging for the Brocade FastIron platforms. In general, Layer 1 issues are related to hardware, the most common being physical connectivity problems.

Hardware backplane debug command

The following command enables the hardware backplane debugging feature in a device.

debug hw

Syntax: [no] **debug hw**

This command enables the hardware backplane debugging feature. To disable this feature, enter the **no** form of the command.

```
Brocade# debug hw
HW BP:  backplane debugging is on
```

Loop detect debug commands

Brocade Port Loop Detection (PLD) protocol allows the Brocade devices to detect loops and disable a port that is on the receiving end of a loop. The loop is detected by sending Bridge Packet Data Unit (BPDU) test packets.

debug loop-detect

Syntax: [no] **debug loop-detect**

This command initiates debugging the loop detection.

NOTE
Execute the command prior to configuring the modes.

Execute the following command to configure loop detection of a single port in strict mode.

```
Brocade# debug loop-detect
Configure loop-detection strict mode on port 1/1/25 :
```

```
Brocade(config-if-e1000-1/1/25)# loop-detection
```

When there is loop detection activity in the switch, the debug information regarding loop detection is displayed on the console. The following example shows the output for the **debug loop-detect** command when the port goes into a disabled state because of loop detection (strict mode).

```
insert_disable primary 1/1/25, vlan=4096
Loop-detection: port 1/1/25 (vlan=1), put into errdisable state
```

Execute the following command to configure loop detection in a VLAN in loose mode.

```
Configure loop-detection loose mode on port vlan 2 :
Brocade(config-vlan-2)# loop-detection
Brocade(config-vlan-2)# insert_disable primary 1/1/31, vlan=2
```

The following output is displayed after the configuration.

```
Loop-detection: port 1/1/31 (vlan=2), put into errdisable state
insert_disable primary 1/1/32, vlan=2
Loop-detection: port 1/1/32 (vlan=2), put into errdisable state
```

debug loop-detect level

Syntax: [**no**] **debug loop-detect level** <*decimal*>

This command debugs loop detection in a particular level.

```
Brocade# debug loop-detect level 1
```

STRICT MODE:

```
Topology:  port 1/1/25 is single port.
Brocade# debug loop-detect level 3
Configure loop-detection on port 1/1/25 :
Brocade(config-if-e1000-1/1/25)# loop-detection
Brocade(config-if-e1000-1/1/25)# Loop-detection: port 1/1/25 (vlan=1), put into
errdisable state
```

LOOSE MODE:

```
Topology:  port 1/1/31 is connected to 1/1/32 in same vlan.

Configure loop-detection on port vlan 2 :
Brocade(config-vlan-2)# loop-detection
Brocade(config-vlan-2)# Loop-detection: port 1/1/31 (vlan=2), put into errdisable
state
Loop-detection: port 1/1/32 (vlan=2), put into errdisable state
```

Port debug commands

The following commands deal with the debugging of port activity.

debug port hw-state

Syntax: [**no**] **debug port hw-state**

This command monitors the hardware status of a port. This helps to enable or disable port debugging in the hardware state.

```
Brocade# debug port hw-state
Topology:     Port 1/1/31 is connected to port 1/1/32.
```

If port 1/1/31 is disabled, an output similar to the following is displayed.

```
Brocade(config)# interface ethernet 1/1/31
Brocade(config-if-e1000-1/1/31)# disable  ( when port is disabled)
Change port 1/1/31 hw_state from PORT_READY(7) to DISABLED(0)
stack: 4040 2050A970 2050AA44 2060D498 20608BF8 2060B5A4 20576A90 20576C74
2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418 2011BBC8
20047698 2011CF7C 5008
Brocade(config-if-e1000-1/1/31)# change port 1/1/32 hw_state from PORT_READY(7)
to SEEQ_INIT(1)
stack: 4040 2050A970 2050AA44 2060D498 20608E18 20609344 20609AFC 2055FA50
2050A214 200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
```

If port 1/1/31 is enabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/32)# interface ethernet 1/1/31
Brocade(config-if-e1000-1/1/31)# enable
Change port 1/1/31 hw_state from DISABLED(0) to SEEQ_INIT(1)
stack: 4040 2050A970 2050AA44 2060D498 20608A34 2060B598 20576BC4 20576C74
2057707C 202A4F84 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418 2011BBC8
20047698 2011CF7C 5008
Brocade(config-if-e1000-1/1/31)# change port 1/1/31 hw_state from SEEQ_INIT(1) to
PORT_READY(7)
stack: 4040 2050A970 2050AA44 2060D498 206093D0 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
Change port 1/1/32 hw_state from SEEQ_INIT(1) to PORT_READY(7)
stack: 4040 2050A970 2050AA44 2060D498 206093D0 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
```

debug port port

Syntax: [no] **debug port port** *stackid/slot/port*

This command monitors a specific physical port and is used with the **debug port up-down** or **debug port hw-state** commands. The *stackid/slot/port* variable refers to the stack ID, slot number, and port number.

```
Brocade# debug port port 1/1/2
Topology:  Port 1/1/31 is connected to port 1/1/2.
```

When the **debug port hw-state** command is enabled after the **debug port port** command, the output similar to the following example is displayed.

```
Brocade# debug port port 1/1/2
monitor port 1/1/2
Brocade# debug port hw-state
```

If port 1/1/2 is disabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/2)# disable
port_disable_cmd: for port 0x1 1/1/2
U1, hal_pp_link_port_enable(1/1/2, 0), is_stby=0, to_shadow=0, parsed = 3,
stack=1
stack: 0116c83c 000de8f0 00accc3c 00acd6ac 00ad21f4 0136c614 0136c850 0136cf00
012b842c 0084b3b0 0084e0a0 00847608 00966c0c 0086c110 0086c154 00243f98 000f0524
00246f54 0159f920 017713c4
pp_link_port_en_dis(port=1/1/2, 0), return 0
stack: 0116ca20 000de8f0 00accc3c 00acd6ac 00ad21f4 0136c614 0136c850 0136cf00
012b842c 0084b3b0 0084e0a0 00847608 00966c0c 0086c110 0086c154 00243f98 000f0524
00246f54 0159f920 017713c4
Change 1/1/2 state from Forward to Disable
```

```
stack: 00acb98c 00ad1e14 00aebd74 0071dddc 0071ebbc 00af0e8c 00af18e8 00ad0e20
00acccf4 00acd6ac 00ad21f4 0136c614 0136c850 0136cf00 012b842c 0084b3b0 0084e0a0
00847608 00966c0c 0086c110 0086c154 00243f98 000f0524 00246f54 0159f920 017713c4
Change port 1/1/2 hw_state from PORT_READY(7) to DISABLED(0)
stack: 00ad5d4c 00accd2c 00acd6ac 00ad21f4 0136c614 0136c850 0136cf00 012b842c
0084b3b0 0084e0a0 00847608 00966c0c 0086c110 0086c154 00243f98 000f0524 00246f54
0159f920 017713c4
Change 1/1/2 state from Disable to Blocked
stack: 00acb98c 00acba3c 00acbce4 00accd38 00acd6ac 00ad21f4 0136c614 0136c850
0136cf00 012b842c 0084b3b0 0084e0a0 00847608 00966c0c 0086c110 0086c154 00243f98
000f0524 00246f54 0159f920 017713c4
Change 1/1/2 state from Blocked to Disable
stack: 00acb98c 00acba3c 00accd48 00acd6ac 00ad21f4 0136c614 0136c850 0136cf00
012b842c 0084b3b0 0084e0a0 00847608 00966c0c 0086c110 0086c154 00243f98 000f0524
00246f54 0159f920 017713c4
```

If port 1/1/2 is enabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/2)# enable
Brocade(config-if-e1000-1/1/2)# change port 1/1/2 hw_state from SEEQ_INIT(1) to
PORT_READY(7)
stack: 4040 2050A970 2050AA44 2060D498 206093D0 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
Change 1/1/2 state from Disable to Blocked
stack: 4040 2050A970 2050AA44 206082E0 2060B354 206215F8 203F9F10 203FB4D8
203FA2B4 2062268C 20623404 2060A624 20609520 20609AFC 2055FA50 2050A214 200478DC
2011BBCC 20047698 2011CF7C
Change 1/1/2 state from Blocked to Listen
stack: 4040 2050A970 2050AA44 206082E0 2060B354 20621734 203F9F10 203FD490
203FD1C4 203FA2BC 2062268C 20623404 2060A624 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698
Change 1/1/2 state from Listen to Learn
stack: 4040 2050A970 2050AA44 206082E0 2060B354 2062178C 203F9F10 20400F38
20400C84 20400A1C 203F9980 2050A214 200478DC 2011BBCC 20047698 2011CF7C 5008 135C8
18524
Change 1/1/2 state from Learn to Blocked
stack: 4040 2050A970 2050AA44 206082E0 2060B354 206215F8 203F9F10 203FD70C
203FD248 203FDFD8 204F3C8C 20616AB8 20612D30 200DC33C 200DC600 200DC6C0 200DC780
200444C4 2011BC28 20047698
```

The **debug port up-down** disables or enables a port to get output information of the specified port. When the **debug port up-down** command is enabled after the **debug port port** command, the output similar to the following example is displayed.

```
Brocade# debug port port 1/1/32
monitor port 1/1/32
Brocade# debug port up-down
```

If port 1/1/32 is disabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/32)# disable
port_down_indication. port=1/1/32, UNTAG, vlan-idx=1
stack: 4040 2050A970 2050AA44 2060A8FC 20608BEC 2060B5A4 20576A90 20576C74
2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418 2011BBC8
20047698 2011CF7C 5008
Change 1/1/32 state from Blocked to Disable
stack: 4040 2050A970 2050AA44 206082E0 2060B354 206214F0 203F9F10 203FA3A0
20622F60 20623404 2060AA38 20608BEC 2060B5A4 20576A90 20576C74 2057707C 202A50A8
20491E9C 20492DC0 204906D8
Change 1/1/32 state from Disable to Blocked
```

```
stack: 4040 2050A970 2050AA44 206082E0 20608330 206084C4 20608C00 2060B5A4
20576A90 20576C74 2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 204A03C8
204A0418 2011BBC8 20047698
Change 1/1/32 state from Blocked to Disable
stack: 4040 2050A970 2050AA44 206082E0 20608330 20608C0C 2060B5A4 20576A90
20576C74 2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418
2011BBC8 20047698 2011CF7C
```

If port 1/1/32 is enabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/32)# enable
Change 1/1/32 state from Disable to Blocked
stack: 4040 2050A970 2050AA44 206082E0 206089A0 2060B598 20576BC4 20576C74
2057707C 202A4F84 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418 2011BBC8
20047698 2011CF7C 5008
Brocade(config-if-e1000-1/1/32)# pp_link_change_final. port=1/1/32, up=1
port_up_indication. port=1/1/32, UNTAG, vlan-idx=1
stack: 4040 2050A970 2050AA44 2060A2C8 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
Change 1/1/32 state from Blocked to Listen
stack: 4040 2050A970 2050AA44 206082E0 2060B354 20621734 203F9F10 203FD490
203FD1C4 203FA2BC 2062268C 20623404 2060A624 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698
Change 1/1/32 state from Listen to Blocked
stack: 4040 2050A970 2050AA44 206082E0 2060B354 206215F8 203F9F10 203FD70C
203FD248 203FDFD8 204F3C8C 20616AB8 20612D30 200DC33C 200DC600 200DC6C0 200DC780
200444C4 2011BC28 20047698
```

debug port ten-gig

Syntax: [no] **debug port ten-gig**

This command monitors all 10 Gigabit ports in the Brocade FastIron devices and is used in conjunction with the **debug port hw-state** command.

```
Brocade# debug port ten-gig
Topology:  Port 2/3/1 is the 10 gig port which is Up and forwarding on stack.
```

```
Brocade# debug port ten-gig
Brocade# debug port hw-state
  hw-state       monitor hw_state change
Brocade# debug port hw-state
```

If port 1/1/32 is disabled, an output similar to the following is displayed.

```
Brocade(config-if-e10000-2/3/1)# disable
Change port 2/3/1 hw_state from PORT_READY(7) to DISABLED(0)
stack: 4040 2050A970 2050AA44 2060D498 20608BF8 2060B5A4 20576A90 20576C74
2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 2027FD8C 20280888 202809F0
2015FCDC 2015FF44 20160B80
```

If port 1/1/32 is enabled, an output similar to the following is displayed.

```
Brocade(config-if-e10000-2/3/1)# enable
Change port 2/3/1 hw_state from DISABLED(0) to SEEQ_INIT(1)
stack: 4040 2050A970 2050AA44 2060D498 20608A34 2060B598 20576BC4 20576C74
2057707C 202A4F84 20491E9C 20492DC0 204906D8 2050BF48 2027FD8C 20280888 202809F0
2015FCDC 2015FF44 20160B80
Brocade(config-if-e10000-2/3/1)# change port 2/3/1 hw_state from SEEQ_INIT(1) to
PORT_READY(7)
stack: 4040 2050A970 2050AA44 2060D498 206093D0 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
```

debug port up-down

Syntax: [no] **debug port up-down** *decimal*

This command monitors the status of the ports. The port status up, down, and up/down are indicated by 1, 2, and 3 respectively. The *decimal* variable refers to the status of the port.

```
Brocade# debug port up-down 1
Monitor port up
Topology:  Port 1/1/31 is connected to port 1/1/32.
```

The **debug port up-down** command disables or enables a port to get output information of the specified port.

If port 1/1/31 is disabled, an output similar to the following is displayed.

```
Brocade(config)# interface ethernet 1/1/31
Brocade(config-if-e1000-1/1/31)# disable
port_down_indication. port=1/1/31, UNTAG, vlan-idx=1
stack: 4040 2050A970 2050AA44 2060A8FC 20608BEC 2060B5A4 20576A90 20576C74
2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418 2011BBC8
20047698 2011CF7C 5008
Brocade(config-if-e1000-1/1/31)# port_down_indication. port=1/1/32, UNTAG,
vlan-idx=3
stack: 4040 2050A970 2050AA44 2060A8FC 20608DA0 20609344 20609AFC 2055FA50
2050A214 200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
pp_link_change_final. port=1/1/32, up=0
```

If port 1/1/31 is enabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/31)# enable
Brocade(config-if-e1000-1/1/31)# pp_link_change_final. port=1/1/31, up=1
port_up_indication. port=1/1/31, UNTAG, vlan-idx=1
stack: 4040 2050A970 2050AA44 2060A2C8 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
pp_link_change_final. port=1/1/32, up=1
port_up_indication. port=1/1/32, UNTAG, vlan-idx=3
stack: 4040 2050A970 2050AA44 2060A2C8 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
```

debug port vlan

Syntax: [no] **debug port vlan** *decimal*

This command monitors a specific VLAN and is used with the **debug port up-down** and **debug port** commands. The *decimal* variable refers to the number of the VLAN.

```
Brocade# debug port vlan 1
monitor vlan 1

Topology:  Port 1/1/31 is connected to port 1/1/32.

Brocade# debug port vlan 2
monitor vlan 2
Brocade# debug port up-down
  up-down        monitor port up/down event. 1: up, 2: down, 3: up/down

Brocade# debug port up-down 3
Monitor both port up and down
```

If port 1/1/31 is disabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/31)# disable
port_down_indication. port=1/1/31, TAG, vlan-idx=4096
stack: 4040 2050A970 2050AA44 2060A8FC 20608BEC 2060B5A4 20576A90 20576C74
2057707C 202A50A8 20491E9C 20492DC0 204906D8 2050BF48 204A03C8 204A0418 2011BBC8
20047698 2011CF7C 5008
Brocade(config-if-e1000-1/1/31)# port_down_indication. port=1/1/32, TAG,
vlan-idx=4096
stack: 4040 2050A970 2050AA44 2060A8FC 20608DA0 20609344 20609AFC 2055FA50
2050A214 200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
pp_link_change_final. port=1/1/32, up=0
```

If port 1/1/31 is enabled, an output similar to the following is displayed.

```
Brocade(config-if-e1000-1/1/31)# enable
Brocade(config-if-e1000-1/1/31)# pp_link_change_final. port=1/1/31, up=1
port_up_indication. port=1/1/31, TAG, vlan-idx=4096
stack: 4040 2050A970 2050AA44 2060A2C8 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
pp_link_change_final. port=1/1/32, up=1
port_up_indication. port=1/1/32, TAG, vlan-idx=4096
stack: 4040 2050A970 2050AA44 2060A2C8 20609520 20609AFC 2055FA50 2050A214
200478DC 2011BBCC 20047698 2011CF7C 5008 135C8 18524
```

System debug commands

The following system debug commands enable debugging of the system services and device drivers.

debug system campram

Syntax: [no] **debug system campram**

This command traces Content Addressable Memory (CAM) or Parallel Random Access Machine (PRAM) operations.

```
Brocade# debug system campram
cam/pram:  Trace debugging is on
```

debug system optics

Syntax: [no] **debug system optics**

This command activates optical monitor debugging.

```
Brocade# debug system optics
optics:  Trace debugging is on
```

debug system poll

Syntax: [no] **debug system poll**

This command is used for backplane polling.

```
Brocade# debug system poll
backplane-poll:  Trace debugging is on
```

Software licensing debug command

This section describes the debug command that generates software licensing information.

debug license

Syntax: [no] **debug license**

This command is used to display the package information on which the license has been loaded. The license information is encoded as Hexa decimal values and can be displayed only when the **show license** command is used with the license index; for example, show license unit 1 index 1.

Before enabling debugging::

```
Brocade# show license unit 1 index 1
License information for unit 1 license <1>:
        +license name:        FCX-ADV-LIC-SW
        +lid:                 deyHHGLhGvz
        +license type:        trial
        +status:              active
        +license period:      30 days
Trial license information:
        +days used:          0
        +hours used:         0
        +days left:          30
        +hours left:         0
```

After enabling debugging::

```
Brocade# debug license
License all debugging ON
Brocade(config)# show license unit 1 index 1
License information for unit 1 license <1>:
        +license name:        FCX-ADV-LIC-SW
        +lid:                 deyHHGLhGvz
        +license type:        trial
        +status:              active
        +license period:      30 days
Trial license information:
        +license precedence: 1
        +days used:          0
        +hours used:         0
        +days left:          30
        +hours left:         0
Brocade license information:
        +pkg info:     0X200
```

Chapter

Layer 2 Diagnostics 3

This chapter contains the following sections:

STP debug commands

The Spanning Tree Protocol (STP) eliminates Layer 2 loops in networks, by selectively blocking some ports and allowing other ports to forward traffic, based on the global (bridge) and local (port) parameters.

STP-related features, such as Rapid Spanning Tree Protocol (RSTP) and Per VLAN Spanning Tree (PVST), extend the operation of standard STP, enabling you to fine-tune standard STP and avoid some of its limitations.

You can enable or disable STP on a global basis (for the entire device), a port-based Virtual Local Area Network (VLAN) basis (for the individual Layer 2 broadcast domain), or an individual port basis. For more information on configuring STP, refer to the *FastIron Configuration Guide*.

A control protocol, such as STP, can block one or more ports in a protocol-based VLAN that uses a virtual routing interface to route to other VLANs. For IP VLANs and IP subnet VLANs, even though some of the physical ports of the virtual routing interface are blocked, the virtual routing interface can still route as long as at least one port in the protocol-based VLAN is not blocked by STP.

The following are the commands used to enable the STP debugging.

debug span all_802_1d_events

Syntax: [no] **debug span all_802_1d_events vlan** *decimal*

This command monitors information about all the events, timers, and packets on a specific VLAN. The *decimal* variable refers to the number of the VLAN.

The output from this command resembles the following example.

```
Brocade# debug span all_802_1d_events vlan 2
STP  Enabling All events Debugging for VLAN 2
```

```
Brocade# STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
```

```
Disabling the debug span all_802_1d_events on FCX:
Brocade# no debug span all_802_1d_events  vlan  2
STP : Disabling All 802.1w Debugging for VLAN 2
```

debug span config

Syntax: [**no**] **debug span config vlan** *decimal*

This command monitors information about STP Bridge Protocol Data Unit (BPDU) configuration on a specific VLAN. The *decimal* variable refers to the number of the VLAN.

If the configuration of VLAN 2 on the Brocade FCX is enabled, the output similar to the following is displayed.

```
Brocade# debug span config vlan 2
STP Enabling packets Debugging for VLAN 2
```

```
Brocade# STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/5
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 05 0100 1400 0200 0f00
STP: Transmitting Config BPDU - VLAN 2 - Port 1/1/6
0000 00 00 00 80000012f2dbfd80 00000004
8000002438154580 08 06 0100 1400 0200 0f00
```

If the configuration of VLAN 2 on the Brocade FCX is disabled, the output similar to the following is displayed.

```
Brocade# no debug  span config vlan 2
STP :   Disabling Packets Debugging for VLAN 2
```

debug span timers

Syntax: [no] **debug span timers vlan** *decimal*

This command displays information about the specific STP timer events. The *decimal* variable refers to the number of the VLAN.

If the STP timer event on VLAN 2 of the Brocade FastIron device is enabled, the output similar to the following is displayed.

```
Brocade# debug span timers vlan 2
STP Enabling Timer Debugging for VLAN 2
```

```
Sample output:
STP: Timer Alert - Forward Delay Timer expired On port 1/1/6(5) , VLAN 2
STP: Timer Alert - Forward Delay Timer expired On port 1/1/1(0) , VLAN 2
STP: Timer Alert - Forward Delay Timer expired On port 1/1/6(5) , VLAN 2
STP: Timer Alert - Message Age Timer expired On port 1/1/1(0) , VLAN 2
STP: Timer Alert - Message Age Timer expired On port 1/1/2(1) , VLAN 2
STP: Timer Alert - Forward Delay Timer expired On port 1/1/2(1) , VLAN 2
STP: Timer Alert - Forward Delay Timer expired On port 1/1/2(1) , VLAN 2
```

If the STP timer event on VLAN 2 of the Brocade FastIron device is disabled, the output similar to the following is displayed.

```
Brocade# no debug span timers vlan 2
STP  Disabling Timer Debugging for VLAN 2
```

RSTP debug commands

RSTP provides rapid traffic reconvergence for point-to-point links within a few milliseconds (less than 500 milliseconds) following the failure of a bridge or bridge port. This reconvergence occurs more rapidly than that provided by STP because convergence in RSTP bridges is based on the explicit handshakes between designated ports and their connected root ports rather than on timer values. The **debug 802.1w** command displays some information about RSTP.

NOTE
For RSTP debug commands, enter 4094 if STP is enabled. Enter the VLAN ID for Multiple Spanning Tree Protocol (MSTP). The CLI alerts the user if the VLAN ID does not exist.

debug 802.1w all_802_1w_events

Syntax: [no] **debug 802.1w all_802_1w_events vlan** *decimal*

This command debugs all the RSTP transactions, timers, and packets on a specific VLAN. The *decimal* variable refers to the number of the VLAN.

If the events are enabled, the output similar to the following is displayed.

```
Brocade# debug 802.1w all_802_1w_events vlan 2
RSTP  Enabling All events Debugging for VLAN 2
```

```
Brocade# RSTP[daa69]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) ,
VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/2(1) VLAN 2
RSTP: PTX =>ROLE is ALTERNATE or BACKUP , port 1/1/2(1), VLAN 2
RSTP[daa69]: Timer Alert - helloWhen timer_expired On port 1/1/9(8) , VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/9(8) VLAN 2
RSTP[daa69]: Tx RST Config BPDU Port 1/1/9(8) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00004e20
8000002438154580 08 09 0100 1400 0200 0f00
RSTP: Rcvd RST Config BPDU: Port 1/1/1(0) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 01 0000 0000 0000 0000
RSTP: PRT =>no valid transition found ,no error, port 1/1/1(0) VLAN 2
RSTP: Rcvd RST Config BPDU: Port 1/1/2(1) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 02 0000 0000 0000 0000
RSTP[daa7d]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) , VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/2(1) VLAN 2
RSTP: PTX =>ROLE is ALTERNATE or BACKUP , port 1/1/2(1), VLAN 2
RSTP[daa7d]: Timer Alert - helloWhen timer_expired On port 1/1/9(8) , VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/9(8) VLAN 2
RSTP[daa7d]: Tx RST Config BPDU Port 1/1/9(8) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00004e20
8000002438154580 08 09 0100 1400 0200 0f00
RSTP: Rcvd RST Config BPDU: Port 1/1/1(0) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 01 0000 0000 0000 0000
RSTP: PRT =>no valid transition found ,no error, port 1/1/1(0) VLAN 2
RSTP: Rcvd RST Config BPDU: Port 1/1/2(1) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 02 0000 0000 0000 0000
RSTP[daa91]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) , VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/2(1) VLAN 2
RSTP: PTX =>ROLE is ALTERNATE or BACKUP , port 1/1/2(1), VLAN 2
RSTP[daa91]: Timer Alert - helloWhen timer_expired On port 1/1/9(8) , VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/9(8) VLAN 2
RSTP[daa91]: Tx RST Config BPDU Port 1/1/9(8) VLAN 2
```

If the events are disabled, the output similar to the following is displayed.

```
Brocade# no debug 802.1w all_802_1w_events vlan 2
RSTP  Disabling All 802.1w Debugging for VLAN 2
```

debug 802.1w messages

Syntax: [no] **debug 802.1w messages vlan** *decimal*

This command displays BPDU information on a VLAN. The *decimal* variable refers to the number of the VLAN.

If the 802.1w messages are enabled, an output similar to the following is displayed.

```
Brocade# debug 802.1w messages vlan 2
RSTP  Enabling packets Debugging for VLAN 2
```

```
Brocade# RSTP: Rcvd RST Config BPDU: Port 1/1/1(0) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 01 0000 0000 0000 0000
RSTP: Rcvd RST Config BPDU: Port 1/1/2(1) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
```

```
80000012f2dbfd80 08 02 0000 0000 0000 0000
RSTP[db06d]: Tx RST Config BPDU Port 1/1/9(8) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00004e20
8000002438154580 08 09 0100 1400 0200 0f00
RSTP: Rcvd RST Config BPDU: Port 1/1/1(0) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 01 0000 0000 0000 0000
RSTP: Rcvd RST Config BPDU: Port 1/1/2(1) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 02 0000 0000 0000 0000
RSTP[db081]: Tx RST Config BPDU Port 1/1/9(8) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00004e20
8000002438154580 08 09 0100 1400 0200 0f00
RSTP: Rcvd RST Config BPDU: Port 1/1/1(0) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 01 0000 0000 0000 0000
RSTP: Rcvd RST Config BPDU: Port 1/1/2(1) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 02 0000 0000 0000 0000
RSTP[db095]: Tx RST Config BPDU Port 1/1/9(8) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00004e20
8000002438154580 08 09 0100 1400 0200 0f00

Brocade# RSTP: Rcvd RST Config BPDU: Port 1/1/1(0) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 01 0000 0000 0000 0000
RSTP: Rcvd RST Config BPDU: Port 1/1/2(1) VLAN 2
0000 02 02 7e 80000012f2dbfd80 00000000
80000012f2dbfd80 08 02 0000 0000 0000 0000
```

If the 802.1w messages are disabled, the output similar to the following is displayed.

```
Brocade# no debug 802.1w messages valn 2
RSTP  Disabling Packets Debugging for VLAN 2
```

debug 802.1w timer

Syntax: [no] **debug 802.1w timer vlan** *decimal*

This command debugs the RSTP (802.1w) timer expiration. The *decimal* variable refers to the number of the VLAN.

If the timer is enabled, an output similar to the following is displayed.

```
Brocade# debug 802.1w timer vlan 2

Brocade# RSTP[db6fd]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) ,
VLAN 2
RSTP[db6fd]: Timer Alert - helloWhen timer_expired On port 1/1/9(8) , VLAN 2
RSTP[db711]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) , VLAN 2
RSTP[db711]: Timer Alert - helloWhen timer_expired On port 1/1/9(8) , VLAN 2
RSTP[db725]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) , VLAN 2
RSTP[db725]: Timer Alert - helloWhen timer_expired On port 1/1/9(8) , VLAN 2
RSTP[db739]: Timer Alert - helloWhen timer_expired On port 1/1/2(1) , VLAN 2
```

If the timer is disabled, the output similar to the following is displayed.

```
Brocade# no debug 802.1w timer vlan 2
RSTP  Disabling Timer Debugging for VLAN 2
```

debug 802.1w transitions

Syntax: [**no**] **debug 802.1w transitions vlan** *decimal*

This command debugs the RSTP state machine transitions. The *decimal* variable refers to the number of the VLAN.

If the 802.1w transtitions are enabled, an output similar to the following is displayed.

```
Brocade# debug 802.1w transitions vlan 2
RSTP  Enabling Events Debugging for VLAN 2

Brocade# RSTP: PRT =>no valid transition found ,no error, port 1/1/1(0) VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/2(1) VLAN 2
RSTP: PTX =>ROLE is ALTERNATE or BACKUP , port 1/1/2(1), VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/9(8) VLAN 2
RSTP: PRT =>no valid transition found ,no error, port 1/1/1(0) VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/2(1) VLAN 2
RSTP: PTX =>ROLE is ALTERNATE or BACKUP , port 1/1/2(1), VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/9(8) VLAN 2
RSTP: PRT =>no valid transition found ,no error, port 1/1/1(0) VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/2(1) VLAN 2
RSTP: PTX =>ROLE is ALTERNATE or BACKUP , port 1/1/2(1), VLAN 2
RSTP: PTX entering the Periodic state  , port 1/1/9(8) VLAN 2
```

If the 802.1w transtitions are disabled, the output similar to the following is displayed.

```
Brocade# no debug 802.1w transitions vlan 2
RSTP  Disabling Events Debugging for VLAN 2
```

MSTP debug commands

With Multiple Spanning Tree Protocol (MSTP), the entire network runs a common instance of RSTP. Within the common instance, one or more VLANs can be individually configured into distinct regions. The entire network runs the Common Spanning Tree (CST) instance and the regions run a local instance, or Internal Spanning Tree (IST). Because the CST treats each IST as a single bridge, ports are blocked to prevent loops that might occur within an IST and also throughout the CST. In addition, MSTP can coexist with individual devices running STP or RSTP in the Common and Internal Spanning Tree instance (CIST). With the exception of the provisions for multiple instances, MSTP operates exactly like RSTP.

The following MSTP debug commands are used for debugging information.

debug mstp bpdu

Syntax: [**no**] **debug mstp bpdu**

This command records and displays information in conjunction with the **debug mstp enable** and **debug mstp events** commands. When the **debug mstp bpdu** command is enabled, it monitors MSTP BPDUs.

If this command is enabled, an output similar to the following is displayed.

```
Brocade# debug mstp bpdu
MSTP Bpdu debugging ON

Brocade# debug mstp enable

Brocade# MSTP[0xeb217]: PRX RECEIVE->RECEIVE - Port 1/1/1
```

```
MST 0, Port 1/1/1 - received BPDU

(802.1s) 0000 03 02 7c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8001 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
MSTP[0xeb217]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0xeb217]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0xeb217]: PRX RECEIVE->RECEIVE - Port 1/1/2
MST 0, Port 1/1/2 - received BPDU
        [0xeb217] (802.1s) 0000 03 02 3c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8002 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
MSTP[0xeb217]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0xeb217]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0xeb21b]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0xeb21b]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
MSTP[0xeb22b]: PRX RECEIVE->RECEIVE - Port 1/1/1
MST 0, Port 1/1/1 - received BPDU
        [0xeb22b] (802.1s) 0000 03 02 7c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8001 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
MSTP[0xeb22b]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0xeb22b]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0xeb22b]: PRX RECEIVE->RECEIVE - Port 1/1/2
MST 0, Port 1/1/2 - received BPDU
        [0xeb22b] (802.1s) 0000 03 02 3c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8002 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
MSTP[0xeb22b]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0xeb22b]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0xeb22f]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0xeb22f]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
MSTP[0xeb23f]: PRX RECEIVE->RECEIVE - Port 1/1/1
MST 0, Port 1/1/1 - received BPDU
        [0xeb23f] (802.1s) 0000 03 02 7c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8001 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
MSTP[0xeb23f]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0xeb23f]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0xeb23f]: PRX RECEIVE->RECEIVE - Port 1/1/2
MST 0, Port 1/1/2 - received BPDU
        [0xeb23f] (802.1s) 0000 03 02 3c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8002 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
MSTP[0xeb23f]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0xeb23f]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0xeb243]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0xeb243]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
MSTP[0xeb253]: PRX RECEIVE->RECEIVE - Port 1/1/1
MST 0, Port 1/1/1 - received BPDU
        [0xeb253] (802.1s) 0000 03 02 7c 80000012f2dbfd80 00000000
        80000012f2dbfd80 00000000 8001 0000 0014 0002 000f 80000012f2dbfd80 14
        {0xac36177f50283cd4b83821d8ab26de62}
```

debug mstp events

Syntax: [no] **debug mstp events**

This command displays MSTP state machine events. It monitors any MSTP events that take place.

```
Brocade# debug mstp events
MSTP Event debugging ON
```

If MSTP events are enabled, an output similar to the following is displayed.

```
Brocade# debug mstp enable
Brocade# MSTP[0xeda7f]: PRX RECEIVE->RECEIVE - Port 1/1/1
MSTP[0xeda7f]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0xeda7f]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0xeda7f]: PRX RECEIVE->RECEIVE - Port 1/1/2
MSTP[0xeda7f]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0xeda7f]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0xeda89]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0xeda89]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
MSTP[0xeda93]: PRX RECEIVE->RECEIVE - Port 1/1/1
MSTP[0xeda93]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0xeda93]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
```

If MSTP events are disabled, an output similar to the following is displayed.

```
Brocade# no debug mstp events
MSTP Event debugging OFF
```

debug mstp level

Syntax: [no] **debug mstp level** *decimal*

This command monitors MSTP. Level 3 gives more information than level 2, and level 0 gives no information. The *decimal* variable refers to the level of MSTP.

```
Brocade# debug mstp level 2
Configuration :
```

```
Brocade# [ebd95] MSTP_RX[pid=#0]  ***
[ebd95] MSTP_RX[pid=#0] done (region) ***
[ebd95] MSTP_RX[pid=#1]  ***
[ebd95] MSTP_RX[pid=#1] done (region) ***
[ebda9] MSTP_RX[pid=#0]  ***
[ebda9] MSTP_RX[pid=#0] done (region) ***
[ebda9] MSTP_RX[pid=#1]  ***
[ebda9] MSTP_RX[pid=#1] done (region) ***
[ebdbd] MSTP_RX[pid=#0]  ***
[ebdbd] MSTP_RX[pid=#0] done (region) ***
[ebdbd] MSTP_RX[pid=#1]  ***
[ebdbd] MSTP_RX[pid=#1] done (region) ***
[ebdd1] MSTP_RX[pid=#0]  ***
[ebdd1] MSTP_RX[pid=#0] done (region) ***
[ebdd1] MSTP_RX[pid=#1]  ***
[ebdd1] MSTP_RX[pid=#1] done (region) ***
[ebde5] MSTP_RX[pid=#0]  ***
[ebde5] MSTP_RX[pid=#0] done (region) ***
[ebde5] MSTP_RX[pid=#1]  ***
```

If the MSTP level 3 is enabled, an output similar to the following is displayed.

```
Brocade# debug mstp level 3

Brocade# [ebf4d] MSTP_RX[pid=#0]  ***
cist_Rx[ebf4d] CIST, Port(#0)
[ebf4d] MSTP_RX[pid=#0] done (region) ***
[ebf4d] MSTP_RX[pid=#1]  ***
cist_Rx[ebf4d] CIST, Port(#1)
[ebf4d] MSTP_RX[pid=#1] done (region) ***
[ebf61] MSTP_RX[pid=#0]  ***
cist_Rx[ebf61] CIST, Port(#0)
[ebf61] MSTP_RX[pid=#0] done (region) ***
[ebf61] MSTP_RX[pid=#1]  ***
cist_Rx[ebf61] CIST, Port(#1)
[ebf61] MSTP_RX[pid=#1] done (region) ***
n[ebf75] MSTP_RX[pid=#0]  ***
cist_Rx[ebf75] CIST, Port(#0)
[ebf75] MSTP_RX[pid=#0] done (region) ***
[ebf75] MSTP_RX[pid=#1]  ***
cist_Rx[ebf75] CIST, Port(#1)
[ebf75] MSTP_RX[pid=#1] done (region) ***
```

debug mstp msti

Syntax: [no] **debug mstp msti** *decimal*

This command displays information for a specific MSTP instance. The *decimal* variable specifies the value 0 for CIST and from 1 through 4094 for the Multiple Spanning Tree Instance (MSTI).

```
Brocade# debug mstp msti 0
MSTP debugging turned on for instances 0
```

If this command is enabled, an output similar to the following is displayed.

```
Brocade# debug mstp msti 2
MSTP debugging turned on for instances 2 ,0

Brocade# debug mstp enable
Brocade# debug mstp events
MSTP Event debugging ON

Brocade# MSTP[0x175f15]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0x175f15]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
MSTP[0x175f21]: PRX RECEIVE->RECEIVE - Port 1/1/1
MSTP[0x175f21]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0x175f21]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0x175f21]: PIM CURRENT->RECEIVE - MST 2, Port 1/1/1
MSTP[0x175f21]: PIM RECEIVE->REPEATED_DESIGNATED - MST 2, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileMsti mst=2, Port 1/1/1, rcvdInfoWhile 4->7
MSTP[0x175f21]: PRX RECEIVE->RECEIVE - Port 1/1/2
MSTP[0x175f21]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0x175f21]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0x175f21]: PIM CURRENT->RECEIVE - MST 2, Port 1/1/2
MSTP[0x175f21]: PIM RECEIVE->REPEATED_DESIGNATED - MST 2, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileMsti mst=2, Port 1/1/2, rcvdInfoWhile 4->7
MSTP[0x175f29]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0x175f29]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
noMSTP[0x175f35]: PRX RECEIVE->RECEIVE - Port 1/1/1
MSTP[0x175f35]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
```

```
MSTP[0x175f35]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0x175f35]: PIM CURRENT->RECEIVE - MST 2, Port 1/1/1
MSTP[0x175f35]: PIM RECEIVE->REPEATED_DESIGNATED - MST 2, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileMsti mst=2, Port 1/1/1, rcvdInfoWhile 4->7
MSTP[0x175f35]: PRX RECEIVE->RECEIVE - Port 1/1/2
MSTP[0x175f35]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0x175f35]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0x175f35]: PIM CURRENT->RECEIVE - MST 2, Port 1/1/2
MSTP[0x175f35]: PIM RECEIVE->REPEATED_DESIGNATED - MST 2, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileMsti mst=2, Port 1/1/2, rcvdInfoWhile 4->7
 deMSTP[0x175f3d]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0x175f3d]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
bug msMSTP[0x175f49]: PRX RECEIVE->RECEIVE - Port 1/1/1
MSTP[0x175f49]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/1
MSTP[0x175f49]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/1, rcvdInfoWhile 5->7
MSTP[0x175f49]: PIM CURRENT->RECEIVE - MST 2, Port 1/1/1
MSTP[0x175f49]: PIM RECEIVE->REPEATED_DESIGNATED - MST 2, Port 1/1/1
MSTP: mstp_updtRcvdInfoWhileMsti mst=2, Port 1/1/1, rcvdInfoWhile 4->7
MSTP[0x175f49]: PRX RECEIVE->RECEIVE - Port 1/1/2
MSTP[0x175f49]: PIM CURRENT->RECEIVE - MST 0, Port 1/1/2
MSTP[0x175f49]: PIM RECEIVE->REPEATED_DESIGNATED - MST 0, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileCist mst=0, Port 1/1/2, rcvdInfoWhile 5->7
MSTP[0x175f49]: PIM CURRENT->RECEIVE - MST 2, Port 1/1/2
MSTP[0x175f49]: PIM RECEIVE->REPEATED_DESIGNATED - MST 2, Port 1/1/2
MSTP: mstp_updtRcvdInfoWhileMsti mst=2, Port 1/1/2, rcvdInfoWhile 4->7
tpMSTP[0x175f51]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/1
MSTP[0x175f51]: PTX IDLE->TRANSMIT_PERIODIC - Port 1/1/2
 eventMSTP[0x175f5d]: PRX RECEIVE->RECEIVE - Port 1/1/1
```

debug mstp port-level

Syntax: [no] **debug mstp port-level** *decimal*

This command displays the information about the hardware level of the port on which the MSTP is running. The port level determines the level of the information. The *decimal* variable refers to the level of the port.

```
Brocade# debug mstp port-level 1
```

```
Configuration for port-level 1 :
Brocade# debug mstp port-level
  DECIMAL   0 -- print nothing
Brocade# debug mstp port-level 1
Brocade# configure terminal
Brocade(config)# interface ethernet 1/1/1
Brocade(config-if-e1000-1/1/1)# disable
Brocade(config-if-e1000-1/1/1)# enable
Brocade(config-if-e1000-1/1/1)# ^Z
```

debug mstp state

Syntax: [no] **debug mstp state**

This command displays information about the MSTP port state events.

```
Brocade# debug mstp state
MSTP Port State debugging ON
```

debug mstp verbose

Syntax: [no] **debug mstp verbose**

This command displays the MSTP debug information in the verbose mode.

```
Brocade# debug mstp verbose
MSTP debugging set to VERBOSE mode
```

debug mstp show

Syntax: debug mstp show

This command displays the current MSTP debug parameters that are enabled.

```
Brocade# debug mstp show
mstp debug is on because of enabled, level or port-level
level 4
parameters: Brief, StateMachineEvents BpduEvents are being tracked
Ports: All
MSTP instances: 0
```

If this command is enabled, an output similar to the following is displayed.

```
Brocade#debug mstp show
mstp debug is off because enabled=0, level=0 & port-level=0
parameters: Brief, No events being tracked
Ports: ethe 1/1/1
MSTP instances: All
```

DHCP debug commands

These commands are used to debug the various Dynamic Host Configuration Protocol (DHCP) client features.

debug dhcp-client alarms

Syntax: [no] **debug dhcp-client alarms**

This command displays the debugging of the DHCP client on a particular port ID.

```
Brocade#debug dhcp-client alarms
"DHCPC: failed to initialize port; dhcpc unable to continue”
"DHCPC: failed to initialize protocol timer"
"DHCPC: no tftp server address or name found. unable to download configuration
file”
"DHCPC: unable to construct dns request”
"DHCPC: dns failed to resolve tftp server name”
“DHCPC: dns aborted”
“DHCPC: No DHCP Servers  found on any ports”
"DHCPC: No DHCP Servers  found on any ports"
"DHCPC: setting 0 seconds lease time"
"DHCPC: setting 0 seconds lease time on port port-id"
"DHCPC: No DHCP Servers  found on any ports"
"DHCPC: failed to allocate a dhcpc packet on port port-id"
"DHCPC: failed to send message on port port-id"
"DHCPC: failed to allocate a dhcpc packet on port port-id"
"DHCPC: failed to send message on port port-id "
"DHCPC: get_an_ip_send_packet () failed on port port-id"
```

debug dhcp-client events

Syntax: [no] **debug dhcp-client events**

This command displays information about DHCP client events related to configuration.

```
Brocade# debug dhcp-client events
"DHCPC: failed to delete static ip-address to ip-address"
"DHCPC: failed to set the port ip-address to ip-address; subnet mask ip-address"
"DHCPC: changing port port-id state from REQUEST to BOUND"
"DHCPC: exceeds maximum some-number DNS servers"
"DHCPC: added ip-address dns-server address"
"DHCPC: failed to set the port ip-address to ip-address; subnet mask ip-address"
"DHCPC: TFTP timeout error for bootfile name %s"
"DHCPC: TFTP client busy"
"DHCPC: TFTP error wrong file type"
"DHCPC: TFTP long file name error"
"DHCPC: TFTP vlan-id invalid"
"DHCPC: TFTP flash write errors"
"DHCPC: TFTP error out of buffer space"
"DHCPC: TFTP flash read error"
"DHCPC: TFTP flash preparation for read failed"
"DHCPC: TFTP flash preparation for write failed"
"DHCPC: %s failed to save running-configuration"
"DHCPC: changing protocol from running to stopped"
"DHCPC: changing protocol from stopped to running"
"DHCPC: Auto update in progress, cannot change DHCP client state on port port-id "
"DHCPC: invalid parameter for "
"DHCPC: dhcp_get_next_port() bad port number port-id"
"DHCPC: sent DHCP-REQUEST message on port port-id \"
"DHCPC: sent DHCP-RENEWING message on port port-id"
"DHCPC: sent DHCP-REBINDING message on port port-id "
"DHCPC: sent DHCP-RELEASE message on port port-id "
"DHCPC: received packet port port-id"
"DHCPC: received bad packet port port-id, no DHCP END OPT found"
"DHCPC: received offer message on port port-id"
"DHCPC: recieved non matching 'xid' (0x%X) in offer message"
"DHCPC: changing port port-id state from INIT-SELECTING to INIT-SELECTING"
"DHCPC: changing port port-id state from INIT-SELECTING to REQUEST"
"DHCPC: received DHCPCACK message on port port-id"
"DHCPC: recieved non matching 'xid'(0x%X) in DHCPCACK message"
"DHCPC: changing timer-event to SEEKING_CONFIGURATION_AND_WAITING_FOR_OFFERS"
"DHCPC: recieved non matching 'xid'(0x%X) in DHCPNAK message"
"DHCPC: deleting existing ip address configuration on port port-id"
"DHCPC: received DHCPNAK packet on port port-id"
“DHCPC: changing port port-id state from REQUEST to INIT-SELECTING”
“DHCPC: received DHCPCOFFER packet on port port-id> while in bound state”
"DHCPC: received DHCPCREQUEST packet on port port-id while in bound state"
"DHCPC: received DHCPCDECLINE packet on port port-id while in bound state"
"DHCPC: received DHCPCACK packet on port port-id while in bound state"
"DHCPC: received DHCPCNAK packet on port port-id while in bound state"
"DHCPC: received DHCPCRELEASE packet on port port-id while in bound state"
"DHCPC: received DHCPCINFORM packet on port port-id while in bound state"
"DHCPC: received DHCPCINFORM packet on port port-id while in bound state"
"DHCPC: received unknown packet on port port-id while in bound state"
"DHCPC: received DHCPACK packet in renewal-state on port port-id"
"DHCPC: received non matching 'xid'(0x%X) in RENEWAL message"
"DHCPC: TFTP flash read error"
"DHCPC: changing port port-id state from RENEWING to BOUND"
"DHCPC: received non matching 'xid'(0x%X) in DHCPNAK message"
```

```
"DHCPC: received DHCPNAK packet in renewal-state on port port-id"
"DHCPC: changing port port-id state from RENEWING to INIT-SELECTING"
"DHCPC: received DHCPACK packet in rebind-state on port port-id"
"DHCPC: received non matching 'xid'(0x%X) in REBIND message"
"DHCPC: changing port port-id state from REBINDING to BOUND"
"DHCPC: received non matching 'xid'(0x%X) in DHCPNAK message"
"DHCPC: received DHCPNAK packet in renewal-state on port port-id"
"DHCPC: changing port port-id state from REBIND to INIT-SELECTING"
"DHCPC: sending packet port port-id"
"DHCPC: send completion called on port: port-id"
"DHCPC: Freeing packet"
"DHCPC: unable to get some-number option from dhcp message from port port-id"
"DHCPC: received server id address opton: ip-address from port port-id"
"DHCPC: received lease-time option: some-number from port port-id"
"DHCPC: received domain-name option: %s from port port-id"
"DHCPC: received dns-server address option: ip-address from port port-id"
"DHCPC: received default-router address option: ip-address from port port-id"
"DHCPC: received TFTP server name option: %s from port port-id"
"DHCPC: received TFTP server address opton: ip-address from port port-id"
"DHCPC: received Bootfile name option: %s from port port-id"
"DHCPC: received 'hostname' option: %s from port port-id"
"DHCPC: received maximum message size option: some-number from port port-id"
"DHCPC: lease timer events called with invalid port"
"DHCPC: changing port port-id state from BOUND to REBINDING"
"DHCPC: changing port port-id state from BOUND to RENEWING"
"DHCPC: deleting existing ip address configuration on port port-id"
"DHCPC: changing port port-id state from REQUESTING to INIT-SELECTING"
"DHCPC: failed to renew ip address with dhcp server; continuing with lease period"
"DHCPC: changing port port-id state from REBINDING to INIT_SELECTING"
"DHCPC: changing port port-id state from REBINDING to REQUESTING"
```

GVRP debug commands

The GARP VLAN Registration Protocol (GVRP) debugging can be enabled by using the **debug gvrp** command.

debug gvrp packets

Syntax: [no] **debug gvrp packets**

This command enables the debugging of GVRP packets.

```
Brocade# debug gvrp packets
GVRP:  Packets debugging is on
GVRP: 0x2095ced4: 01 80 c2 00 00 21 00 e0 52 ab 87 40 00 3a 42 42
GVRP: 0x2095cee4: 03 00 01 01 02 00 04 05 00 02 04 05 00 07 04 05
GVRP: 0x2095cef4: 00 09 04 05 00 0b 04 02 03 e9 04 01 03 eb 04 01
GVRP: 0x2095cf04: 03 ec 04 01 03 ef 04 01 03 f1 04 01 05 dd 04 01
GVRP: 0x2095cf14: 09 cb 04 01 0f a1 00 00
GVRP: Port 2/1 RCV
GVRP: 0x2095ced4: 01 80 c2 00 00 21 00 e0 52 ab 87 40 00 28 42 42
GVRP: 0x2095cee4: 03 00 01 01 04 02 03 e9 04 01 03 eb 04 01 03 ec
GVRP: 0x2095cef4: 04 01 03 ef 04 01 03 f1 04 01 05 dd 04 01 09 cb
GVRP: 0x2095cf04: 04 01 0f a1 00 00
GVRP: Port 2/1 TX
GVRP: 0x207651b8: 01 80 c2 00 00 21 00 04 80 2c 0e 20 00 3a 42 42
GVRP: 0x207651c8: 03 00 01 01 02 00 04 05 03 e9 04 05 03 eb 04 05
GVRP: 0x207651d8: 03 ec 04 05 03 ef 04 05 03 f1 04 05 05 dd 04 05
```

```
GVRP: 0x207651e8: 09 cb 04 05 0f a1 04 02 00 02 04 01 00 07 04 01
GVRP: 0x207651f8: 00 09 04 01 00 0b 00 00
GVRP: Port 2/1 TX
GVRP: 0x207651b8: 01 80 c2 00 00 21 00 04 80 2c 0e 20 00 18 42 42
GVRP: 0x207651c8: 03 00 01 01 04 02 00 02 04 01 00 07 04 01 00 09
GVRP: 0x207651d8: 04 01 00 0b 00 00
```

MAC address debug commands

debug mac

Syntax: [no] **debug mac**

This command enables the Media Access Control (MAC) address debugging action.

```
Brocade# debug mac
MAC DB:  Action debugging is on
mac_action_request: done->
```

```
MA -     Normal,             ALL_SYST,                FLUSH
  Ports: All Ports
  Vlans: All Vlans
stack: 20B89ED4 2028D708 2028D8A0 201AB040 20C2D58C 20C2EA48 20C2D6DC 20C2EDA4
20C2B2B8 20B8B3A8 2076C3EC 2076C448 20590734 205146FC 20592568 5010 15B4C 1AAD8
Brocade# aging_timer. call mac_action_handler()
```

```
MA -     Normal,             ALL_SYST,                FLUSH
  Ports: All Ports
  Vlans: All Vlans
stack: 20B89ED4 2028D708 20C6FAB0 20B894BC 20514924 20590738 205146FC 20592568
5010 15B4C 1AAD8
mac_clear_request(). NO_ACTION, SPECIFIC
stack: 20B89ED4 2028CD28 2028D1F0 20C6FAB4 20B894BC 20514924 20590738 205146FC
20592568 5010 15B4C 1AAD8
```

MCT debug commands

Multi-Chassis Trunking (MCT) provides link-level redundancy and load sharing in addition to increased capacity. If any one of the MCT cluster device fails, the data path remains operational through the other MCT cluster device without any disruption.

This section describes the debug commands that display MCT information.

debug cluster all

Syntax: [no] **debug cluster all**

This command enables or disables the MCT debugging information log.

If the **debug cluster all** command is enabled, the output similar to the following is displayed.

```
Brocade# debug cluster all
MCT:: Enabling all debugs
CLUSTER ALL debugging is now ON
```

```
Brocade# rstp_txRSTP(T=25364)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
```

```
rstp_txRSTP(T=25364)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25365, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25384)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25384)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25385, VLAN=20,port=1/7)
port-stp-state=FORWARDING
CCP_PKT(T=25394): Receiving Packet from peer 10.1.1.2
CCP_PKT(T=25394): Packet Info:0100000c000000080003000400000056
CCP_PKT(T=25394): Receiving Packet from peer 10.1.1.2, appId=0
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
CCP_PKT(T=25394): Packet Info:0100000c000000080003000400000056
CCP_PKT(T=25394):Processing ccp packet
CCP_PKT(T=25394): Packet Info:0003000400000056
CCP_PKT(T=25394): Handling received keep alive message
CCP_PKT(T=25394): Packet Info:0003000400000056
CLUSTER CCP_FSM(T=25394): Fsm7 got keepalive from 10.1.1.2
CLUSTER CCP:free data buf   allocated packet count      1       buffer 2291f5be
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 2291f5be
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 2291f5be
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 2291f5be
rstp_txRSTP(T=25404)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25404)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25405, VLAN=20,port=1/7)
port-stp-state=FORWARDING
CLUSTER CCP_FSM(T=25413): Fsm12 10.1.1.2 sending keepalive
CLUSTER CCP PEER(T=25413): Make keep alive message ->
CCP_PKT(T=25413): Sending Keep Alive message to peer
CCP_PKT(T=25413): Packet Info:0100000c000000080003000400000056
CLUSTER CCP_FSM(T=25413): Fsm11  KeepAlive check for peer 10.1.1.2
rstp_txRSTP(T=25424)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25424)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25425, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25444)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25444)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25445, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25464)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25464)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25465, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25484)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25484)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25485, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25504)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25504)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25505, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25524)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
```

```
rstp_txRSTP(T=25524)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25525, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25544)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25544)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25545, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25564)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25564)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25565, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25584)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25584)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25585, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25604)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25604)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25605, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25624)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25624)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25625, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25644)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25644)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25645, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25664)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25664)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25665, VLAN=20,port=1/7)
port-stp-state=FORWARDING
rstp_txRSTP(T=25684)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25684)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
rstputil_mct_handle_incoming_BPDU(T=25685, VLAN=20,port=1/7)
port-stp-state=FORWARDING

Brocade# CCP_PKT(T=25694): Receiving Packet from peer 10.1.1.2
CCP_PKT(T=25694): Packet Info:0100000c000000080003000400000057
CCP_PKT(T=25694): Receiving Packet from peer 10.1.1.2, appId=0
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
CCP_PKT(T=25694): Packet Info:0100000c000000080003000400000057
CCP_PKT(T=25694):Processing ccp packet
CCP_PKT(T=25694): Packet Info:0003000400000057
CCP_PKT(T=25694): Handling received keep alive message
CCP_PKT(T=25694): Packet Info:0003000400000057
CLUSTER CCP_FSM(T=25694): Fsm7 got keepalive from 10.1.1.2
CLUSTER CCP:free data buf   allocated packet count      1       buffer 2291f5be
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 2291f5be
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 2291f5be
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 2291f5be
```

```
Brocade# rstp_txRSTP(T=25704)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25704)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&

Brocade# rstputil_mct_handle_incoming_BPDU(T=25705, VLAN=20,port=1/7)
port-stp-state=FORWARDING
no CLUSTER CCP_FSM(T=25713): Fsm12 10.1.1.2 sending keepalive
CLUSTER CCP PEER(T=25713): Make keep alive message ->
CCP_PKT(T=25713): Sending Keep Alive message to peer
CCP_PKT(T=25713): Packet Info:0100000c000000080003000400000057
debug crstp_txRSTP(T=25724)(vlan=20,port=1/15) Tx on CCEP, MCT peer is superior
rstp_txRSTP(T=25724)(vlan=20,port=1/15) Tx on CCEP while MCT peer is superior and
UP, abandom&
lurstputil_mct_handle_incoming_BPDU(T=25725, VLAN=20,port=1/7)
port-stp-state=FORWARDING
```

If the **debug cluster all** command is disabled, the output similar to the following is displayed.

```
Brocade# no debug cluster all
MCT:: disabling all debugs
CLUSTER ALL debugging is now OFF
```

debug cluster ccp

Syntax: [no] **debug cluster ccp** [**fsm** | **tcp** | **packets** | **itc** | **client** | **events** | **memory**]

- **fsm** - Enables CCP FSM debugging for a given peer IP or all peers.
- **tcp** - Enables CCP TCP debugging for a given peer IP or all peers.
- **packets** - Enables CCP packet debugging for a given peer IP or all peers.
- **itc** - Enables CCP ITC debugging .
- **client** - Enables CCP client related registration debugging.
- **events** - Enables CCP events debugging.
- **memory** - Enables CCP memory debugging.

This command enables debugging of all CCP FSM messages, CCP client registrations, CCP packet exchanges from clients, Keepalive messages and notifications, data buffer usage, and specific peer-related message exchanges. Command output resembles the following example.

```
Brocade# debug cluster ccp
CLUSTER CCP debugging is on for all

============= Packet Receive ======================
SX_4#CCP_PKT(T=104981): Receiving Packet from peer 10.10.10.2
 CCP_PKT(T=104981): Packet
Info:0100001c000400180009001400010065012c0300000000008000002438
CCP_PKT(T=104981): Receiving Packet from peer 10.10.10.2, appId=4
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
 CCP_PKT(T=104981): Packet
Info:0100001c000400180009001400010065012c0300000000008000002438
CCP_PKT(T=104981): Passing packet to app 4 (cluster=1,mac=2,stp=4)
 CCP_PKT(T=104981): Packet
Info:000400180009001400010065012c0300000000008000002438221b1a
================= Packet passed to specific client ======================
ccp_pass_pkt_to_client()(T=104981): cluster id 1, peer ip=168430082, STP msg
CLUSTER CCP: free data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 21a215e8
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 21a215e8
```

```
CCP_PKT(T=105001): Receiving Packet from peer 10.10.10.2
 CCP_PKT(T=105001): Packet
Info:0100001c000400180009001400010065012c03000000000008000002438
CCP_PKT(T=105001): Receiving Packet from peer 10.10.10.2, appId=4
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
 CCP_PKT(T=105001): Packet
Info:0100001c000400180009001400010065012c03000000000008000002438
CCP_PKT(T=105001): Passing packet to app 4 (cluster=1,mac=2,stp=4)
 CCP_PKT(T=105001): Packet
Info:000400180009001400010065012c030000000000800000002438221b1a
ccp_pass_pkt_to_client()(T=105001): cluster id 1, peer ip=168430082, STP msg
CLUSTER CCP: free data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 21a215e8
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 21a215e8
CCP_PKT(T=105021): Receiving Packet from peer 10.10.10.2
 CCP_PKT(T=105021): Packet
Info:0100001c000400180009001400010065012c03000000000008000002438
CCP_PKT(T=105021): Receiving Packet from peer 10.10.10.2, appId=4
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
 CCP_PKT(T=105021): Packet
Info:0100001c000400180009001400010065012c03000000000008000002438
CCP_PKT(T=105021): Passing packet to app 4 (cluster=1,mac=2,stp=4)
 CCP_PKT(T=105021): Packet
Info:000400180009001400010065012c030000000000800000002438221b1a
ccp_pass_pkt_to_client()(T=105021): cluster id 1, peer ip=168430082, STP msg
CLUSTER CCP: free data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 21a215e8
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 21a215e8
CCP_PKT(T=105041): Receiving Packet from peer 10.10.10.2
 CCP_PKT(T=105041): Packet
Info:0100001c000400180009001400010065012c03000000000008000002438
CCP_PKT(T=105041): Receiving Packet from peer 10.10.10.2, appId=4
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
 CCP_PKT(T=105041): Packet
Info:0100001c000400180009001400010065012c03000000000008000002438
CCP_PKT(T=105041): Passing packet to app 4 (cluster=1,mac=2,stp=4)
 CCP_PKT(T=105041): Packet
Info:000400180009001400010065012c030000000000800000002438221b1a
ccp_pass_pkt_to_client()(T=105041): cluster id 1, peer ip=168430082, STP msg
CLUSTER CCP: free data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 21a215e8
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 21a215e8

SX_4#CCP_PKT(T=105059): Receiving Packet from peer 10.10.10.2
 CCP_PKT(T=105059): Packet Info:0100000c000000080003000400000015f
CCP_PKT(T=105059): Receiving Packet from peer 10.10.10.2, appId=0
(CCP_APPLICATION_CCP=0;CCP_APPLICATION_CLUSTER_MGR=1)
 CCP_PKT(T=105059): Packet Info:0100000c000000080003000400000015f
CCP_PKT(T=105059):Processing ccp packet
 CCP_PKT(T=105059): Packet Info:000300040000015f
CCP_PKT(T=105059): Not handling keepalive as keepalive_timeout is set.
 CCP_PKT(T=105059): Packet Info:000300040000015f
================= Data buffer usage ============================
CLUSTER CCP:free data buf   allocated packet count      1       buffer 21a215e8
CLUSTER CCP:internal free data buf   allocated packet count  0  bufer 21a215e8
================= Error notifications ==============
CLUSTER CCP PEER(T=105059): Sending Error Notification 49
```

```
CCP_PKT(T=105059): Sending Notify message to peer
 CCP_PKT(T=105059): Packet
Info:0100001600000120001000e0000015f00000031000000000000
CLUSTER CCP:internal alloc data buf  allocated packet count  1  buffer 21a215e8
CLUSTER CCP:alloc data buf   allocated packet count  1  buffer 21a215e8
============== CCP State machine ==============
CLUSTER CCP_FSM(T=105060): Fsm12 10.10.10.2 sending keepalive
CLUSTER CCP PEER(T=105060): Make keep alive message ->
================== Send Keepalive ======================
CCP_PKT(T=105060): Sending Keep Alive message to peer
 CCP_PKT(T=105060): Packet Info:0100000c00000008000300040000160
```

If the **debug cluster ccp** command is disabled, the output similar to the following is displayed.

```
Brocade# no debug cluster ccp
CCP debugging is off for all
```

debug cluster client-auto-detect

Syntax: [no] **debug cluster client-auto-detect**

This command displays debugging information related to cluster client auto detect functionality, as shown in the following example.

```
Brocade# debug cluster client-auto-detect
CLUSTER AUTOCONFIG INFO - New Client: A-satya-000000147d80 with rbridge id: 3318
discovered on LACP port 4/17
CLUSTER AUTOCONFIG INFO - New Client: A-satya-000000c74d40 with rbridge id: 2668
discovered on Static port 3/13
CLUSTER AUTOCONFIG INFO - Port 3/14 added as Static interface to
Client:A-satya-000000c74d40
LAG MCT-CCAC-LAG_1 deployed successfully!
Spanning tree is disabled on CCEP port 4/17 of MCT Client: A-satya-000000147d80.
CLUSTER AUTOCONFIG INFO - Port 4/17 is successfully programmed as client
interface. Removing from autoconfig port list.

debug clLAG MCT-CCAC-LAG_2 deployed successfully!
Spanning tree is disabled on CCEP port 3/13 of MCT Client: A-satya-000000c74d40.
CLUSTER AUTOCONFIG INFO - Port 3/13 is successfully programmed as client
interface. Removing from autoconfig port list.
CLUSTER AUTOCONFIG INFO - Port 3/14 is successfully programmed as client
interface. Removing from autoconfig port list.
CLUSTER AUTOCONFIG INFO - Cluster 1 client auto-config process stopped
```

debug cluster config

Syntax: [no] **debug cluster config**

This command enables debugging for any addition or deletion of the clusters, addition or deletion of the clients, or exchange of cluster finite state machine (FSM) messages between the peer and the client isolation mode.

If the **debug cluster config** command is enabled, the output similar to the following is displayed.

```
Brocade# debug cluster config
CLUSTER Config debugging is now ON
Brocade# config terminal
Brocade# cluster ABC 1
Brocade(config-cluster-ABC)# rbridge-id 200
Brocade(config-cluster-ABC)# session-vlan 100
Brocade(config-cluster-ABC)# icl icl1 ethernet 4/3
```

```
Brocade(config-cluster-ABC)# peer 10.10.10.1 rbridge-id 100 icl icl1
Brocade(config-cluster-ABC)# deploy

============= cluster create ===============
CLUSTER CCP:ccp_internal_create_cluster::Cluster 1 Configured.
Brocade(config-cluster-ABC)# client c1uster
Brocade(config-cluster-ABC-client-c1)# rbridge-id 300
Brocade(config-cluster-ABC-client-c1)# client-interface ethernet 4/13

=========== Client add ======================
clustercu_add_client_interface(cluseter=1,client_name=c1,client_port=4/13)
clustermgr_add_cluster_client_port(cluseter=1,client_name=c1,client_port=4/13,add
)
Brocade(config-cluster-ABC-client-c1)# deploy

==================== FSM Message ====================
clustermgr_send_loading_info_to_peer(cluster=1)(T=121842) >>>>
clustermgr_send_vlanmask_to_peer && clustermgr_send_intf_mac_to_peer
clustermgr_send_loading_info_to_peer(cluster=1)(T=121842) >>>> DONE

================ Client isolation mode config ==============
Brocade(config-cluster-ABC)# client-isolation strict
clustermgr_client_isolation_mode(cluster_id=1,mode=1)
```

If the **debug cluster config** command is disabled, the output similar to the following is displayed.

```
Brocade# no debug cluster config
CLUSTER Config debugging is now OFF?
```

debug cluster fsm

Syntax: [no] **debug cluster fsm client** *client_id*

This command enables debugging of the MCT cluster FSM for a particular client. Any events that occur in the client such as CCEP Up/Down, remote CCEP Up/Down, or CCP Up/Down, are tracked. The *client_id* variable refers to the port number of the client.

```
Brocade# debug cluster fsm client 300
CLUSTER fsm debugging is now ON for client rbridge 300
================== Local CCEP Down ==================

Brocade(config-if-e1000-4/13)# disable
CLUSTER FSM: cluster id 1, client id 300, old state: Up, event: Local Down
CLUSTER FSM: new state: Remote Up, master: FALSE
Brocade(config-if-e1000-4/13)# CLUSTER FSM: Received CCRR message from peer when
CCP is up

Brocade(config-if-e1000-4/13)# enable
================== Local CCEP UP ==================

Brocade(config-if-e1000-4/13)# CLUSTER FSM: cluster id 1, client id 300, old
state: Remote Up, event: Local Up
CLUSTER FSM: new state: Preforwarding Remote Up, master: FALSE
================== Remote CCEP Down ==================
CLUSTER FSM: Received CCRR message from peer when CCP is up
CLUSTER FSM: cluster id 1, client id 300, old state: Preforwarding Remote Up,
event: CCRR Ack Rcvd
CLUSTER FSM: new state: Up, master: FALSE
CLUSTER FSM: Received CCRR message from peer when CCP is up
CLUSTER FSM: cluster id 1, client id 300, old state: Up, event: Remote Down
CLUSTER FSM: Cluster ABC (Id: 1), client c1 (RBridge Id: 300) - Remote client CCEP
down
CLUSTER FSM: new state: Local Up, master: FALSE
```

```
================== Remote CCEP UP ==================
CLUSTER FSM: Received CCRR message from peer when CCP is up
CLUSTER FSM: cluster id 1, client id 300, old state: Local Up, event: Remote Up
CLUSTER FSM: Cluster ABC (Id: 1), client c1 (RBridge Id: 300) - Remote client CCEP
up
CLUSTER FSM: new state: Up, master: FALSE
==================== CCP Down ============================
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: CCP Down, event: CCP
Down
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: CCP Down, event: CCP
Down
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: CCP Down, event: CCP
Down
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: CCP Down, event: CCP
Down

====================== CCP getting established ========================
CLUSTER FSM: new state: Loading
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: Loading, event: CCP Up
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: Loading, event: CCP Up
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: Loading, event: CCP Up
CLUSTER FSM: Received Loading-Done message from peer
CLUSTER FSM: cluster id 1, peer rbridge id 100, old state: Loading, event: Loading
Done
CLUSTER FSM: getting EVENT_ID_MCT_CCP_UP event
CLUSTER FSM(T=121842)main: cluster id 1, client id 300, old state: Init, event:
CCP Up
CLUSTER FSM(T=121842)main: new state: Local Deploy, master: FALSE
CLUSTER FSM: new state: CCP Up
CLUSTER FSM: Received CCRR message from peer when CCP is up
CLUSTER FSM: cluster id 1, client id 300, old state: Local Deploy, event: Remote
Deploy
CLUSTER FSM: Cluster ABC (Id: 1), client c1 (RBridge Id: 300) - Remote client
deployed
CLUSTER FSM: new state: Admin Up, master: FALSE
CLUSTER FSM: Received CCRR message from peer when CCP is up
```

If the **debug cluster fsm client** command is disabled, the output similar to the following is displayed.

```
Brocade# no debug cluster fsm client 300
CLUSTER fsm debugging is now OFF for client rbridge 300
```

debug cluster intf-mac

Syntax: [no] **debug cluster intf-mac**

This command displays interface MAC debug information for the cluster as shown in the following example.

```
Brocade# debug cluster intf-mac
CLUSTER ROUTER MAC(T=1791244): sent to peer rbridge id 2, base mac address
0000.0021.a500, number of mac addresses 1
CLUSTER ROUTER MAC(T=1791244): received from peer rbridge id 2, router mac address
0000.0083.3200, number of mac addresses 1
```

debug cluster mdup

Syntax: [no] **debug cluster mdup**

This command displays MDUP debug information for the cluster as shown in the following example.

```
Brocade# debug cluster mdup
CLUSTER MDUP: Send Vlan mask message to peer
CLUSTER MDUP: Received vlan mask from peer for cluster 1
mdupmgr_proc_port_oper_status_change(port_id=3/13, oper_status=up, vlan=100 ) ->
mdupmgr_proc_port_oper_status_change(port_id=3/13, oper_status=up, vlan=100 ),
port is CCEP
mdupmgr_proc_port_oper_status_change(port_id=3/13, oper_status=up, vlan=100 ),
port is CCEP -> macmgr_move_mct_mac_from_icl_to_ccep
CLUSTER MDUP: Moving MACs from ICL port to CCEP port 3/13 for client 2668 in VLAN
100
mdupmgr_proc_port_oper_status_change(port_id=3/13, oper_status=up, vlan=100 )
DONE
mdupmgr_proc_port_oper_status_change(port_id=4/17, oper_status=up, vlan=100 ) ->
mdupmgr_proc_port_oper_status_change(port_id=4/17, oper_status=up, vlan=100 ),
port is CCEP
mdupmgr_proc_port_oper_status_change(port_id=4/17, oper_status=up, vlan=100 ),
port is CCEP -> macmgr_move_mct_mac_from_icl_to_ccep
CLUSTER MDUP: Moving MACs from ICL port to CCEP port 4/17 for client 3318 in VLAN
100
mdupmgr_proc_port_oper_status_change(port_id=4/17, oper_status=up, vlan=100 )
DONE
 <<<<< mdupmgr_sync_static_mac_table >>>
mdupmgr_sync_static_mac_table done, sync 0 entries
mdupmgr_sync_mac_table(restart=yes,ccp_up_restart=yes)(T=95629)
mac_mdup_sync_in_progress=0, start_mac_index=0, previous_loop_start_mac_index=0,
g_mac_mdup_sync_delay=0
mdupmgr_sync_mac_table(restart=yes,ccp_up_restart=yes)(T=95629) DELAY SYNC
(mac_mdup_sync_in_progress=1, start_mac_index=0, previous_loop_start_mac_index=0,
g_mac_mdup_sync_delay=2)&
CLUSTER MDUP: Convert CCR to CCL for cluster_id: 1, client rbridge id = 2668
macmgr_convert_ccr_to_ccl(cluster_id=1, port_id=3/13)(T=95629) client
A-satya-000000c74d40 (hw_timer_gone=1 msec)&
CLUSTER MDUP: Convert CCR to CCL for cluster_id: 1, client rbridge id = 3318
macmgr_convert_ccr_to_ccl(cluster_id=1, port_id=4/17)(T=95629) client
A-satya-000000147d80 (hw_timer_gone=1 msec)&
mdupmgr_sync_mac_table(restart=no,ccp_up_restart=no)(T=95630)
mac_mdup_sync_in_progress=1, start_mac_index=0, previous_loop_start_mac_index=0,
g_mac_mdup_sync_delay=2
mdupmgr_sync_mac_table(restart=no,ccp_up_restart=no)(T=95630) DELAY SYNC
(mac_mdup_sync_in_progress=1, start_mac_index=0, previous_loop_start_mac_index=0,
g_mac_mdup_sync_delay=3)&
mdupmgr_sync_mac_table(restart=no,ccp_up_restart=no)(T=95631)
mac_mdup_sync_in_progress=1, start_mac_index=0, previous_loop_start_mac_index=0,
g_mac_mdup_sync_delay=3
mdupmgr_sync_mac_table(restart=no,ccp_up_restart=no)(T=95631) DELAY SYNC
(mac_mdup_sync_in_progress=1, start_mac_index=0, previous_loop_start_mac_index=0,
g_mac_mdup_sync_delay=4)&
```

debug cluster show

Syntax: [no] debug cluster show

This command displays debug flags for the cluster as shown in the following example.

```
Brocade# debug cluster show
CLUSTER debugging is                  :ENABLED

CLUSTER ALL debugging is              :OFF

CLUSTER IPC debugging is              :OFF
CLUSTER Hardware debugging is         :OFF
```

```
CLUSTER Config debugging is           :OFF
CLUSTER FSM debugging is              :OFF
CLUSTER MDUP debugging is             :ON
CLUSTER interface mac debugging is    :OFF
CLUSTER Forwarding debugging is       :OFF
CLUSTER STP debugging is              :OFF
CLUSTER BPDU forwarding debugging is  :OFF

CLUSTER CCP debugging is              :OFF
```

debug cluster stp

Syntax: [no] **debug cluster stp**

This command enables debugging of STP and RSTP modules. The output details how the STP registers the ICL or CCEP ports. Changes occur while supporting MCT on STP, such as ICL guard, CCEP state sync, STP and CCP messages, and topology change messages are tracked. Command output resembles the following example.

```
Brocade# debug cluster stp
CLUSTER STP debugging is now ON

Brocade(config-vlan-101)# spanning-tree 802-1w

STP -> INIT STP for bridge, vlan 10. Port mask 194 201 204 1168 1170 1173 to 1174
2000 2147 to 2149 2154 2157 2185 2187 to 2188 2192 to 2193 2195 to 2196 2201 2205
2368 to 2369 2371 to 2372 2377 2381 2387 to 2389 2394 2397 2425 2427 to 2428

Startup:: In initialize_bridge_spanning_tree::MCT enabled on this VLAN 101

Startup:: In initialize_bridge_spanning_tree:: MCT enabled Clearing ICL port

Searching for ICL and CCEP for this VLAN/STP

stputil_discover_mct:: ICL port is 4/3

Initialize Port 4/3. vlan 101
initialize_port(T=173641): icl-port guard timer cleared on port 4/3, VLAN 101

initialize_port:: Set ICL to FORWARDING. Vlan 101 port 4/3

Searching for ICL and CCEP for this VLAN/STP

Initialize Port 4/10. vlan 101
initialize_port(T=173641): icl-port guard timer cleared on port 4/10, VLAN 101

initialize_port:: Port is neither ICL/ CCEP of Root Bridge. Set to BLOCKING. Vlan
101 port 4/10

Searching for ICL and CCEP for this VLAN/STP

stputil_discover_mct::This is CCEP port 4/13

Initialize Port 4/13. vlan 101
initialize_port(T=173641): icl-port guard timer cleared on port 4/13, VLAN 101

initialize_port:: Set CCEP or root bridge to FORWARDING. Vlan 101 port 4/13
stputil_mct_send_ccep_state_to_peer(T=173641,vlan=101,port=204,state=FORWARDING)
stputil_mct_send_ccep_state_to_peer(T=173641,vlan=101,port=204,state=FORWARDING)
primary_port 4/13 -->
stputil_mct_send_ccep_state_to_peer(T=173641,vlan=101,port=204,state=FORWARDING)
primary_port 4/13 done

rstp_setup_this_bridge_values:: mct_enable = TRUE

rstputil_mct_enable_port.vlan 101 port 4/3
```

```
rstputil_mct_enable_port:: port is trunk

rstputil_mct_enable_port. Per VLAN STP. VLAN 101 Port 4/3 mct_icl_port = TRUE

rstp_enable_port:: VLAN 101 port 4/3. skip_Initialize_RSTP_state_machines = 0.

rstp_enable_port:: PIM state 0. PRT state 0. PTX state 0. TCM state 0. PPM state
0. PST state 0

rstp_enable_port::rstp_role 0 rstp_selectedRole 0 state 5 rstp_infols 0
rstp_initialize_port(T=173641) VLAN101, port 4/3. Port is trunk. primary = 4/3
active stp port = 4/3
 rstp_RoleSelection_State_Machine. set_flag 1 for vlan 101

New root:: vlan 101 INVALID
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

Vlan 101 port 4/3. rstp_selectedRole = RSTP_DESIGNATED_PORT as rstp_infols ==
RSTP_PORT_INFO_STATE_AGED.
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE

rstp_port_Role_Transition_State_Machine:: port 4/3 vlan 101 role = 0 selected role
= 3. old PRT state 529 new state 529. Designated_Port_transitions.
rstp_enable_forwarding(T=173641) VLAN101, port 4/3

rstp_Port_State_Transition_State_Machine_Action: VLAN 101 Port 4/3. ICL port set
to FWD.

rstputil_mct_enable_port.vlan 101 port 4/10
rstp_enable_port:: VLAN 101 port 4/10. skip_Initialize_RSTP_state_machines = 0.

rstp_enable_port:: PIM state 0. PRT state 0. PTX state 0. TCM state 0. PPM state
0. PST state 0

rstp_enable_port::rstp_role 0 rstp_selectedRole 0 state 2 rstp_infols 0
rstp_initialize_port(T=173641) VLAN101, port 4/10. Port is not trunk.
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE

Vlan 101 port 4/10. rstp_selectedRole = RSTP_DESIGNATED_PORT as rstp_infols ==
RSTP_PORT_INFO_STATE_AGED.
rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE
```

```
rstp_compute_port_role_infols_Mine:: vlan 101 port 4/10 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =  FALSE

rstp_port_Role_Transition_State_Machine:: port 4/10 vlan 101 role = 0 selected
role = 3. old PRT state 529 new state 529. Designated_Port_transitions.
rstp_disable_learning(T=173641) VLAN101, port 4/10 BLOCKING

rstp_Port_State_Transition_State_Machine_Action: VLAN 101 Port 4/10 set to
BLOCKING.
rstp_disable_forwarding(T=173641) VLAN101, port 4/10 BLOCKING

rstputil_mct_enable_port.vlan 101 port 4/13

rstputil_mct_enable_port:: port is trunk

rstputil_mct_enable_port. Per VLAN STP. VLAN 101 Port 4/13 CCEP = TRUE

rstputil_mct_enable_port.rstputil_mct_send_rstp_state_reset_to_peer. VLAN 101
Port 4/13 CCEP = TRUE
rstputil_mct_send_stp_state_RESET_to_peer(vlan=173641,port=101) primary_port 4/13
-->
rstputil_mct_send_stp_state_RESET_to_peer(T=173641,vlan=101,port=204)
primary_port 4/13 done

rstp_enable_port:: VLAN 101 port 4/13. skip_Initialize_RSTP_state_machines = 0.

rstp_enable_port:: PIM state 0. PRT state 0. PTX state 0. TCM state 0. PPM state
0. PST state 0

rstp_enable_port::rstp_role 0 rstp_selectedRole 0 state 5 rstp_infols 0
rstp_initialize_port(T=173641) VLAN101, port 4/13. Port is trunk. primary = 4/13
active stp port = 4/13
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/10 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =  FALSE
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/10 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =  FALSE

Vlan 101 port 4/13. rstp_selectedRole = RSTP_DESIGNATED_PORT as rstp_infols ==
RSTP_PORT_INFO_STATE_AGED.
 rstp_RoleSelection_State_Machine. set_flag 2 for vlan 101

New root:: vlan 101 INVALID

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/3 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =   FALSE

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/10 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =  FALSE

rstp_compute_port_role_infols_Mine:: vlan 101 port 4/13 rstp_selectedRole =
RSTP_DESIGNATED_PORT.rstp_updtInfo =  FALSE
```

```
rstp_port_Role_Transition_State_Machine:: port 4/13 vlan 101 role = 0 selected
role = 3. old PRT state 529 new state 529. Designated_Port_transitions.

rstputil_mct_handle_port_role_transition:: old state = 529, new = 3
(RSTP_DESIGNATED_PORT/RSTP_ROOT_PORT). Send ccep_state_to_peer. VLAN 101 port
4/13
rstputil_mct_send_ccep_state_to_peer(T=173641,vlan=101,port=4/13,role=3)
primary_port 4/13 -->
rstputil_mct_send_ccep_state_to_peer(T=173641,vlan=101,port=4/13,role=3)
primary_port 4/13 done
rstp_disable_learning(T=173641) VLAN101, port 4/13 BLOCKING

rstp_Port_State_Transition_State_Machine_Action: VLAN 101 Port 4/13 set to
BLOCKING.
rstp_disable_forwarding(T=173641) VLAN101, port 4/13 BLOCKING
rstp_txRSTP(T=173641)(vlan=101,port=4/13) Tx on CCEP, MCT peer is inferior -> Tx
BPDU ->
Brocade(config-vlan-101)# rstputil_mct_handle_incoming_BPDU(T=173641,
VLAN=101,port=4/10) port-stp-state=DISCARDING
rstputil_mct_handle_incoming_BPDU(T=173641, VLAN=101,port=4/10)
port-stp-state=DISCARDING
```

MCT show commands

This section describes the show commands that display MCT information.

show cluster

Syntax: show cluster

This command displays the complete cluster state information about the ICL, peer, and client, as shown in the following example.

```
Brocade# show cluster
Cluster abc 1
=============
Rbridge Id: 1, Session Vlan: 3999, Keep-Alive Vlan: 4001
Cluster State: Deploy
Client Isolation Mode: Loose
Member Vlan Range: 100

 ICL Info:
 ---------
 Name            Port  Trunk
 icl1            4/9   11

 Peer Info:
 ----------
 Peer IP: 10.1.1.2, Peer Rbridge Id: 2, ICL: icl1
 KeepAlive Interval: 10 , Hold Time: 90, Fast Failover
 Active Vlan Range:  100
 Last Reason for CCP Down: Not Down
 Peer State: CCP Up (Up Time:   0 days: 0 hr: 0 min:41 sec)

 Client Info:
 ------------
 Number of Clients configured: 2
```

```
 Name                    Rbridge-id Config      Port  Trunk FSM-State
 A-satya-000000147d80    3318       Deployed    4/17    3    Admin Up
 A-satya-000000c74d40    2668       Deployed    3/13    2    Up
```

show cluster ccp client

Syntax: show cluster *cluster_name* | *cluster_id* **ccp client** *client_ID*

- *cluster_name*—Specifies the cluster name.
- *cluster_id*—Specifies the cluster port number.
- *client_ID*—Specifies the client port ID.

This command displays the registered CCP clients such as cluster manager, MDUP, or STP.

The following output is displayed when the **show cluster** command is configured with the cluster ID.

```
Brocade# show cluster 3000 ccp client
Client Node Info:
-----------------
  Name: cluster_mgr  ID: 1  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP: 10.1.1.2  Registered Events:  All Events
Client Node Info:
-----------------
  Name: mdup  ID: 2  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  PACKET_RECEIVED
Client Node Info:
-----------------
  Name: stp  ID: 4  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  All Events
Client Node Info:
-----------------
  Name: mcast  ID: 8  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  All Events
Client Node Info:
-----------------
  Name: vrrp  ID: 16  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  All Events
```

The following output is displayed when the **show cluster** command is configured with the cluster name.

```
Brocade# show cluster SX ccp client
Client Node Info:
-----------------
  Name: cluster_mgr  ID: 1  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP: 10.1.1.2  Registered Events:  All Events
Client Node Info:
-----------------
  Name: mdup  ID: 2  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
```

```
Peer IP:  All Peers
  Registered Events:  PACKET_RECEIVED
Client Node Info:
-----------------
  Name: stp  ID: 4  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  All Events
Client Node Info:
-----------------
  Name: mcast  ID: 8  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  All Events
Client Node Info:
-----------------
  Name: vrrp  ID: 16  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP:  All Peers
  Registered Events:  All Events
```

The following output is displayed when the **show cluster** command is configured with the cluster name and the CCP client ID.

```
Brocade# show cluster SX ccp client  1
Client Node Info:
-----------------
  Name: cluster_mgr  ID: 1  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP: 10.1.1.2  Registered Events:  All Events
```

The following output is displayed when the **show cluster** command is configured with the cluster ID and the CCP client ID.

```
Brocade# show cluster 3000 ccp client  1
Client Node Info:
-----------------
  Name: cluster_mgr  ID: 1  Cluster ID: 3000    Number of Peers: 1
Peer Node Info:
Peer IP: 10.1.1.2  Registered Events:  All Events
```

show cluster ccp peer

Syntax: show cluster ccp peer

This command displays the cluster peer-state configuration information as shown in the following example.

```
Brocade# show cluster ccp peer
Cluster Name : abc   Cluster ID: 1
PEER IP ADDRESS          STATE                UP TIME
---------------      -------------           --------------
10.1.1.1                OPERATIONAL       1 days: 4 hr: 0 min:42 sec
```

show cluster ccp peer detail

Syntax: show cluster ccp peer detail

This command displays the cluster peer session and the TCP connection configuration information as shown in the following example.

```
Brocade# show cluster ccp peer detail
**************Peer Session Details*********************
IP address of the peer                          10.1.1.1
Rbridge ID of the peer                          100
Session state of the peer                       OPERATIONAL
Next message ID to be send                      3369
Keep Alive interval in seconds                  30
Hold Time Out in seconds                        90
Fast Failover is enable for the session
UP Time                                          1 days: 4 hr: 3 min:55 sec
Number of  tcp packet allocations failed         0
Message    Init          Keepalive     Notify        Application   Badmessages
Send       9             3932          0             102           0
Receive    9             3932          0             100           0
TCP connection is up
TCP connection is initiated by                  10.1.1.2
TCP connection tcbHandle not pending
TCP connection packets received

**************TCP Connection Details********************
    TCP Connection state: ESTABLISHED      Maximum segment size: 1436
       Local host:  10.1.1.2, Local  Port: 12071
       Remote host: 10.1.1.1, Remote Port: 4175
       ISentSeq:  160671712  SendNext:  160726330  TotUnAck:          0
       TotSent:       54618  ReTrans:           1  UnAckSeq:  160726330
       IRcvSeq:      751804  RcvNext:      806410  SendWnd:       16384
       TotalRcv:      54606  DupliRcv:          0  RcvWnd:        16384
       SendQue:           0  RcvQue:            0  CngstWnd:       1436
```

show cluster client

Syntax: show cluster *cluster_name* | *cluster_id* **client** *client_name* | *client_RbridgeID*

- *cluster_name*—Specifies the cluster name.
- *cluster_id*—Specifies the cluster port ID.
- *client_name*—Specifies the client name.
- *client_RbridgeID*—Specifies the client Rbridge ID.

This command displays cluster information for all clients configured in a cluster or for a specified client.

The following output is displayed when the **show cluster** command is configured with the cluster ID.

```
Brocade# show cluster 3000 client
Client Info:
------------
Number of Clients Configured: 1
Name   Rbridge-id       Config      LACP  Port  Trunk FSM-State
 R1         100        Deployed      no   1/15    -   Preforwarding
                                                        Remote
```

The following output is displayed when the **show cluster** command is configured with the cluster name.

```
Brocade# show cluster SX client
Client Info:
------------
Number of Clients Configured: 1
Name   Rbridge-id       Config      LACP  Port  Trunk FSM-State
```

```
 R1         100        Deployed      no   1/15    -   Preforwarding
                                                       Remote
```

The command output resembles the following example for the speciifed client..

```
Brocade# show cluster 1 client c1
Cluster abc 1
=============
Rbridge Id: 101, Session Vlan: 3999, Keep-Alive Vlan: 4001
Cluster State: Deploy
Client Isolation Mode: Loose
Configured Member Vlan Range: 100 to 105
Active Member Vlan Range: 100 to 105
MCT Peer's Reachability status using Keep-Alive Vlan: Peer Reacheable

Client Info:
------------
Client: c1, rbridge-id: 300, Deployed
Client Port: 3/3
State: Up
Number of times Local CCEP down: 1
Number of times Remote CCEP down: 0
Number of times Remote Client undeployed: 0
Total CCRR packets sent: 11
Total CCRR packets received: 12
```

show cluster config

Syntax: show cluster config [*cluster_id* | *cluster_name*]

This command displays only the cluster configuration information as shown in the following example.

```
Brocade# show cluster 1 config
cluster abc 1
rbridge-id 20
session-vlan 3999
keep-alive-vlan 4001
icl icl1 ethernet 3/1
peer 10.1.1.1 rbridge-id 100 icl icl1
deploy
client c1
rbridge-id 300
client-interface ethernet 3/3
deploy
client c2
rbridge-id 301
client-interface ethernet 3/13
deploy
```

show mac cluster

Syntax: show mac cluster *cluster_ID* | *cluster_name* [**local** | **remote** | **client** | **vlan**]

- *cluster_id - Specifies the cluster ID.*
- *cluster_name - Specifies the cluster name.*
- **local** *- Displays MAC entries that are learned locally.*
- **remote** *- Displays MAC entries that are learned through the MDUP message from the peer.*

- **client** - Displays cluster client MAC entries for all clients or for a specified client.
- **vlan** - Displays cluster MAC entries for the specified VLAN.

The following command output displays all MAC entries for the specified cluster.

```
Brocade# show mac cluster 3000
Total Cluster Enabled(CL+CR+CCL+CCR) MACs: 4
Total Cluster Local(CL) MACs: 1
Total Cluster Client Macs(CCL+CCR) for all clients: 3
Total Cluster Client Local(CCL) MACs for all clients: 3
CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local CR:Remote
Total active entries from all ports = 15
MAC-Address     Port           Type          Index  MCT-Type VLAN
0000.0000.0014  17/8            Dynamic       53172   CL      4081
0000.0082.1d89  11/17-11/17     Dynamic       24337   CCL     3359
0000.00a8.9c8f  1/15-1/15       Dynamic       5709    CCL     3203
0000.0082.1d89  11/17-11/17     Dynamic       7199    CCL     3360
```

The following command output displays all locally learned MAC entries for the specified cluster.

```
Brocade# show mac cluster 3000 local
Total Cluster Local(CL) MACs: 2
CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local CR:Remote
Total active entries from all ports = 6
MAC-Address     Port           Type          Index  MCT-Type VLAN
0000.0086.8e00  7/5-7/6        Dynamic       33994   CL      3000
0000.0000.0004  7/1            Dynamic       28197   CL      10
```

The following command output displays all remote MAC entries for the specified cluster.

```
Brocade# show mac cluster 3000 remote exclude-interface
Total Cluster Remote(CR) MACs: 1
CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local CR:Remote
Total active entries from all ports = 6
MAC-Address     Port           Type          Index  MCT-Type VLAN
0000.0000.0002  7/5-7/6        Dynamic       9111    CR      10
```

The following command output displays all cluster client MAC entries from all the clients.

```
Brocade# show mac cluster 3000 client
Total Cluster Client Macs(CCL+CCR) for all clients: 2
Total Cluster Client Local(CCL) MACs for all clients: 2
CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local CR:Remote
Total active entries from all ports = 6
MAC-Address     Port           Type          Index  MCT-Type VLAN
0000.0000.0005  7/7-7/8        Dynamic       44352   CCL     20
0000.0076.92c0  7/7-7/8        Dynamic       15031   CCL     20
```

The following command output displays all cluster MAC entries for a particular VLAN.

```
Brocade# show mac cluster 3000 vlan 20
Total Cluster Enabled(CL+CR+CCL+CCR) MACs for vlan 20: 2
Total Cluster Client Macs(CCL+CCR) for all clients in vlan 20: 2
Total Cluster Client Local(CCL) MACs for all clients in vlan 20: 2
CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local CR:Remote
Total active entries from VLAN 20 = 2
MAC-Address     Port           Type          Index  MCT-Type VLAN
0000.0000.0005  7/7-7/8        Dynamic       44352   CCL     20
0000.0076.92c0  7/7-7/8        Dynamic       15031   CCL     20
```

show mac mdb

Syntax: show mac mdb source-rbridge *Rbridge_id* **client-rbridge** *Rbridge_id*

- **source-rbridge** *Rbridge_id* - Specifies the source Rbridge ID.
- **client-rbridge** *Rbridge_id* - Specifies the client Rbridge ID.

This command displays the MDB information of the cluster MAC entries. Command output resembles the following example.

```
Brocade# show mac mdb source-rbridge 3
Total Local MDB from all ports = 2
Total Remote MDB from all ports = 3
Code- CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local  CR:Remote
Total active entries from all ports = 6
MAC Address    Port   VLAN      Type S-RBridge C-RBridge P-Portid  Cost Best
0000.0000.0005 4/7    20        CCR  3         100                1    *
0000.0000.0004 4/5    10        CR   3         3         384      1    *
0000.0076.92c0 4/7    20        CCR  3         100                1    *
```

show mac mdup-stats

Syntax: show mac mdup-stats

This command displays the statistics for MDUP packets as shown in the following example.

```
Brocade# show mac mdup-stats
MDUP Information
================
MDUP Data buffers in queue : 0

MDUP Statistics
===============
MDUP Update Messages sent: 9
  Add Mac sent: 14
  Add Intf Mac sent: 0
  Del Mac sent: 0
  Del Intf Mac sent: 0
  Move Mac sent: 0
MDUP Mac Info Messages sent: 19
MDUP Static Mac Messages sent: 0
MDUP Flush Messages sent: 17
MDUP Update Messages received: 6
  Add Mac received: 7
  Add Intf Mac received: 0
  Del Mac received: 2
  Del Intf Mac received: 0
  Move Mac received: 0
MDUP Mac Info Messages received: 16
MDUP Static Mac Messages received: 0
MDUP Flush Messages received: 11
MDUP Add Mac Errors: 3
MDUP Del MAC Errors: 0
MDUP Move MAC Errors: 0
```

Packet-capture debug commands

The following **debug packet-capture** command helps to debug packet flows and capture packets designated for the management module for debugging. This command has additional **filter** and **mode** option parameters for refining the traffic sources and the format of the saved data.

debug packet-capture

Syntax: [no] **debug packet-capture** [**all** | **count-reset** | **count-show** | **exclude-mgmt-port** | **filter** | **max** | **mode** | **no-limit** | **no-mgmt-port** | **receive** | **send**]

- **all**—Displays the debugging information of the packets transmitted and received.
- **count-reset—**Clears the total packet count of the packets captured.
- **count-show —**Displays the total packet count of the packets captured for debugging.
- **exclude-mgmt-port**—Enables debugging of the packets that are excluded for the management port.
- **filter**—Enables the raw packet filter for debugging.
- **max—**Displays the maximum number of packets.
- **mode**—Displays the various packet modes.
- **no-limit—**Displays the debugging information of an unlimited number of packets.
- **no-mgmt-port—**Toggles the display of packets from the management port.
- **receive**—Debugs only the packets that are received.
- **send**—Debugs only the packets that are transmitted.

The **debug packet-capture** command displays information about packet-capture activity.

debug packet-capture filter

Syntax: [no] **debug packet-capture filter** *filter_index filter_options*

The *filter_index* parameter specifies the filter number. The decimal value ranges from 1 through 20.

The *filter_options* parameter specifies the filter option. The following are the various options in filter:

- all—Matches all packet
- arp—Matches an ARP request or response
- broadcast-mac—Matches the destination broadcast MAC address (0000.00FF.FFFF)
- cdp—Matches with the CDP
- clear—Clears the filter entry
- clear-counter—Clears the counter
- cpu-code—Matches the CPU code
- da—Matches the destination MAC address
- dhcp—Matches with the DHCP
- dot1x—Matches the dot1x protocol
- dpa—Matches the destination IP address
- dport—Matches the destination protocol port

- ether-type—Matches the Ethernet type packets
- fdp—Matches wit h the FDP
- in-port—Matches the input port
- l2-802.1w—Matches the 802.1w BPDU
- l2-lacp—Matches the LACP PDU
- l2-mstp—Matches the MSTP BPDU
- l2-stp—Matches the spanning tree BPDU
- l3-bgp—Matches with the BGP
- l3-ospf—Matches the OSPF protocol
- l3-vrrp—Matches the VRRP
- l3-vrrpe—Matches the VRRP-E
- lldp—Matches the LLDP
- mrp—Matches the MRP
- out-port—Matches the output port
- priority—Matches the priority
- protocol—Matches the IP
- pvst—Matches the PVST protocol
- sa—Matches the source MAC address
- snmp—Matches the SNMP
- spa—Matches the source IP address
- sport—Matches the source protocol port
- udld—Matches the UDLD protocol
- vlan-id—Matches the VLAN ID
- vsrp—Matches the VSRP

This command enables the filter option for the raw packets. There are 32 filter options that can be configured within a single filter. The filter index limit ranges from 1 through 20.

```
Brocade# debug packet-capture filter 2 protocol
```

debug packet-capture mode

Syntax: [no] **debug packet-capture mode** [**brief** | **no-display** | **normal** | **pcap-fmt** [**default** | *decimal*]]

- **brief**—Specifies the increment counters and displays the packet summary.
- **no-display**—Specifies the increment counters alone.
- **normal**—Specifies the increment counters and displays the packet header and the first 48 raw bytes.
- **pcap-fmt**—Specifies the increment counters and dumps the packet in pcap (packet capture) hexadecimal format.
- **default**—Specifies the normal mode.
- *decimal*—Specifies the first 48 bytes of the packets received.

This command displays the format of the packets to be captured for debugging information.

debug packet-capture filter show

Syntax: debug packet-capture filter show [**all** | *decimal*] **none**

This command displays the filter information for the particular filter index. The *decimal* variable refers to the number of the packet captured for filter.

Execute the following command to display the complete filter information.

```
Brocade# debug packet-capture filter show all
Filter 1:
        Match STP
Filter 2:
        Match 802.1W
```

Execute the following command to display filter information for a particular filter number.

```
Brocade# debug packet-capture filter show 2
Filter 2:
        Match 802.1W
```

Execute the following command to remove all the filters that are configured.

```
Brocade# debug packet-capture filter none
```

VXLAN debug commands

Virtual eXtensible Local Area Network (VXLAN) is an overlay technology to interconnect two or more logical Layer 2 networks on top of an Layer 3 IP network to support multitenant environment.

This section describes the debug commands used for debugging the VXLAN Layer 2 tunnel configurations.

debug l2-tunnel all

Syntax: [no] **debug l2-tunnel all**

This command enables debug of all flows of any Layer 2 tunnel. Command output resembles the following example.

```
Brocade# debug l2-tunnel all
[VXLAN]: L2-tunnel 1 -  next hop 0.0.0.0 is not router dest - update Tunnel
[VXLAN]: Update 2 all maps on L2-tunnel 1
[VXLAN]: Success to update all maps on L2-tunnel 1[VXLAN]: Create vxlan tables for
L2-tunnel 1
[VXLAN]: Create L3 intf 65535 for L2-tunnel 1
[VXLAN]: Reusing L3 intf 6096 for L2-tunnel 1 with exising L3 intf 65535 which
reused from L2-tunnel 65535
[VXLAN]: Success to create L3 intf 6096 for L2-tunnel 1
[VXLAN]: Create vxlan egress ip tunnel for L2-tunnel 1 with index ffffffff
[VXLAN]: Success to create vxlan egress ip tunnel for L2-tunnel 1 with index
4c000002
[VXLAN]: Create Forward nextHop 65535 for L2-tunnel 1 on network port 1/1/2
[VXLAN]: Success to reuse Forward nexthop 66 for L2-tunnel 1 from L2-tunnel 2 on
network port 1/1/2
[VXLAN]: Create vxlan terminator for L2-tunnel 1
[VXLAN]: Fail to get vxlan terminator on device 0 : Entry not found
[VXLAN]: Success to create vxlan terminator for L2-tunnel 1
[VXLAN]: Success to create vxlan tables for L2-tunnel 1
```

```
[VXLAN]: Update 1 all maps on L2-tunnel 1
[VXLAN]: Create VXLAN map for port 1/1/1, vlan 31, vni 3100, L2-tunnel 1
[VXLAN]: Create VXLAN VPN for vni 3100
[VXLAN]: Success to create VXLAN VPN for vni 3100 with VPN 7000
[VXLAN]: Add access port 1/1/1, vlan 31 to vpn 7000
[VXLAN]: Success to create access egress nexthop 100004
[VXLAN]: Success to create access vxlan port 80000001 for access port 1/1/1 on
vlan 31 in vpn 7000
[VXLAN]: Add network port 1/1/2 to vpn 7000 with existing vxlan port 80000002 on
L2-tunnel 1
[VXLAN]: Success to add network port 1/1/2 to vpn 7000 with new vxlan port
80000002 on L2-tunnel 1
[VXLAN]: Success to create VXLAN map
[VXLAN]: Success to update all maps on L2-tunnel 1
[VXLAN]: bring tunnel id 1 up
[VXLAN]: Create tunnel id 1 with next hop 64 - old next hop is 4
```

debug l2-tunnel vxlan all

Syntax: [no] debug l2-tunnel vxlan all

This command enables debug of all flows of a VXLAN tunnel. Command output resembles the following example.

```
Brocade# debug l2-tunnel vxlan all
[VXLAN]: L2-tunnel 1 -  next hop 0.0.0.0 is not router dest - update Tunnel
[VXLAN]: Update 2 all maps on L2-tunnel 1
[VXLAN]: Success to update all maps on L2-tunnel 1[VXLAN]: Create vxlan tables for
L2-tunnel 1
[VXLAN]: Create L3 intf 65535 for L2-tunnel 1
[VXLAN]: Reusing L3 intf 6096 for L2-tunnel 1 with exising L3 intf 65535 which
reused from L2-tunnel 65535
[VXLAN]: Success to create L3 intf 6096 for L2-tunnel 1
[VXLAN]: Create vxlan egress ip tunnel for L2-tunnel 1 with index ffffffff
[VXLAN]: Success to create vxlan egress ip tunnel for L2-tunnel 1 with index
4c000002
[VXLAN]: Create Forward nextHop 65535 for L2-tunnel 1 on network port 1/1/2
[VXLAN]: Success to reuse Forward nexthop 66 for L2-tunnel 1 from L2-tunnel 2 on
network port 1/1/2
[VXLAN]: Create vxlan terminator for L2-tunnel 1
[VXLAN]: Fail to get vxlan terminator on device 0 : Entry not found
[VXLAN]: Success to create vxlan terminator for L2-tunnel 1
[VXLAN]: Success to create vxlan tables for L2-tunnel 1
[VXLAN]: Update 1 all maps on L2-tunnel 1
[VXLAN]: Create VXLAN map for port 1/1/1, vlan 31, vni 3100, L2-tunnel 1
[VXLAN]: Create VXLAN VPN for vni 3100
[VXLAN]: Success to create VXLAN VPN for vni 3100 with VPN 7000
[VXLAN]: Add access port 1/1/1, vlan 31 to vpn 7000
[VXLAN]: Success to create access egress nexthop 100004
[VXLAN]: Success to create access vxlan port 80000001 for access port 1/1/1 on
vlan 31 in vpn 7000
[VXLAN]: Add network port 1/1/2 to vpn 7000 with existing vxlan port 80000002 on
L2-tunnel 1
[VXLAN]: Success to add network port 1/1/2 to vpn 7000 with new vxlan port
80000002 on L2-tunnel 1
[VXLAN]: Success to create VXLAN map
[VXLAN]: Success to update all maps on L2-tunnel 1
[VXLAN]: bring tunnel id 1 up
[VXLAN]: Create tunnel id 1 with next hop 64 - old next hop is 4
```

debug l2-tunnel vxlan tunnel all

Syntax: [no] **debug l2-tunnel vxlan tunnel all**

This command enables debug of all flows of a VXLAN tunnel creation, updation, and deletion. Command output resembles the following example.

```
Brocade# debug l2-tunnel vxlan tunnel all
[VXLAN]: L2-tunnel 1 -  next hop 0.0.0.0 is not router dest - update Tunnel
[VXLAN]: Update 2 all maps on L2-tunnel 1
[VXLAN]: Success to update all maps on L2-tunnel 1[VXLAN]: Create vxlan tables for
L2-tunnel 1
[VXLAN]: Create L3 intf 65535 for L2-tunnel 1
[VXLAN]: Reusing L3 intf 6096 for L2-tunnel 1 with exising L3 intf 65535 which
reused from L2-tunnel 65535
[VXLAN]: Success to create L3 intf 6096 for L2-tunnel 1
[VXLAN]: Create vxlan egress ip tunnel for L2-tunnel 1 with index ffffffff
[VXLAN]: Success to create vxlan egress ip tunnel for L2-tunnel 1 with index
4c000002
[VXLAN]: Create Forward nextHop 65535 for L2-tunnel 1 on network port 1/1/2
[VXLAN]: Success to reuse Forward nexthop 66 for L2-tunnel 1 from L2-tunnel 2 on
network port 1/1/2
[VXLAN]: Create vxlan terminator for L2-tunnel 1
[VXLAN]: Fail to get vxlan terminator on device 0 : Entry not found
[VXLAN]: Success to create vxlan terminator for L2-tunnel 1
[VXLAN]: Success to create vxlan tables for L2-tunnel 1
[VXLAN]: Update 1 all maps on L2-tunnel 1
[VXLAN]: Success to update all maps on L2-tunnel 1
[VXLAN]: bring tunnel id 1 up
[VXLAN]: Create tunnel id 1 with next hop 64 - old next hop is 4
```

debug l2-tunnel vxlan tunnel sw-prog

Syntax: [no] **debug l2-tunnel vxlan tunnel sw-prog**

This command enables debug of software programming flows of VXLAN tunnel creation, updation, and deletion. Command output resembles the following example.

```
Brocade# debug l2-tunnel vxlan tunnel sw-prog
[VXLAN]: L2-tunnel 1 -  next hop 0.0.0.0 is not router dest - update Tunnel
[VXLAN]: Update 2 all maps on L2-tunnel 1
[VXLAN]: Success to update all maps on L2-tunnel 1
[VXLAN]: Update 1 all maps on L2-tunnel 1
[VXLAN]: Success to update all maps on L2-tunnel 1
[VXLAN]: bring tunnel id 1 up
[VXLAN]: Create tunnel id 1 with next hop 64 - old next hop is 4
```

debug l2-tunnel vxlan tunnel hw-prog

Syntax: [no] **debug l2-tunnel vxlan tunnel hw-prog**

This command enables debug of hardware programming flows of VXLAN tunnel creation, updation, and deletion. Command output resembles the following example.

```
Brocade# debug l2-tunnel vxlan tunnel hw-prog
[VXLAN]: Create vxlan tables for L2-tunnel 2
[VXLAN]: Create L3 intf 65535 for L2-tunnel 2
[VXLAN]: Reusing L3 intf 6096 for L2-tunnel 2 with exising L3 intf 65535 which
reused from L2-tunnel 65535
[VXLAN]: Success to create L3 intf 6096 for L2-tunnel 2
```

```
[VXLAN]: Create vxlan egress ip tunnel for L2-tunnel 2 with index ffffffff
[VXLAN]: Success to create vxlan egress ip tunnel for L2-tunnel 2 with index
4c000002
[VXLAN]: Create Forward nextHop 65535 for L2-tunnel 2 on network port 1/1/2
[VXLAN]: Success to reuse Forward nexthop 66 for L2-tunnel 2 from L2-tunnel 1 on
network port 1/1/2
[VXLAN]: Create vxlan terminator for L2-tunnel 2
[VXLAN]: Fail to get vxlan terminator on device 0 : Entry not found
[VXLAN]: Success to create vxlan terminator for L2-tunnel 2
[VXLAN]: Success to create vxlan tables for L2-tunnel 2
```

debug l2-tunnel vxlan mapping

Syntax: [no] **debug l2-tunnel vxlan mapping**

This command enables debug of flows of VXLAN mapping creation, updation, and deletion. Command output resembles the following example.

```
Brocade# debug l2-tunnel vxlan mapping
[VXLAN]: Create VXLAN map for port 1/1/1, vlan 31, vni 3100, L2-tunnel 1
[VXLAN]: Create VXLAN VPN for vni 3100
[VXLAN]: Success to create VXLAN VPN for vni 3100 with VPN 7000
[VXLAN]: Add access port 1/1/1, vlan 31 to vpn 7000
[VXLAN]: Success to create access egress nexthop 100004
[VXLAN]: Success to create access vxlan port 80000001 for access port 1/1/1 on
vlan 31 in vpn 7000
[VXLAN]: Add network port 1/1/2 to vpn 7000 with existing vxlan port 80000002 on
L2-tunnel 1
[VXLAN]: Success to add network port 1/1/2 to vpn 7000 with new vxlan port
80000002 on L2-tunnel 1
[VXLAN]: Success to create VXLAN map
```

Link aggregation debug commands

The trunk group feature allows you to manually configure multiple high-speed load-sharing links between two Brocade Layer 2 switches or Layer 3 switches or between a Brocade Layer 2 switch and Layer 3 switch and a server.

In addition to enabling load sharing of traffic, trunk groups provide redundant, alternate paths for traffic if any of the segments fail. Trunk groups are manually configured aggregate links containing multiple ports. The 802.3ad link aggregation protocol dynamically creates and manages trunk groups.

NOTE
The terms "trunk group" and "link aggregation group" are used interchangeably in this document.

The following commands are used for troubleshooting trunk groups:

- debug pp-trunk-hash
- debug pp-bum-trunk-hash

debug pp-trunk-hash

Traces and displays the output port through which a given packet can be sent out in a trunk.

Syntax **debug pp-trunk-hash** [*trunk_port* | *ingress_port*]

debug pp-trunk-hash *trunk_port*

unicast-distribution ip *dmac smac* [*dst_ipv4 src_ipv4 dst_tcp/udp_port src_tcp/udp_port* | *dst_ipv6 src_ipv6 dst_tcp/udp_port src_tcp/udp_port*]

unicast-distribution non-ip *dmac smac*

debug pp-trunk-hash *ingress_ port*

v4-device *sw_ trunk_ ID* [**registered-multicast** *vlan vidx* | **routed-flow** *dst_ipv4 src_ipv4 IPHdr_Protocol* | **switched-flow ip** *dst_ipv4 src_ipv4* | **switched-flow non-ip** *dmac smac* | **unregistered-multicast** *vlan*]

v6-device *sw_trunk_ID* [**routed-multicast-distribution** *vlan* | **unicast-distribution** [**ip** *dmac smac* [*dst_ipv4 src_ipv4* [*dst_tcp* | *udp_port*] [*src_tcp* | *udp_port*] | *dst_ipv6 src_ipv6* [*dst_tcp* | *udp_port*] [*src_tcp* | *udp_port*]]| **non-ip** *dmac smac*]]

no debug pp-trunk-hash

Parameters

trunk_port Specifies the trunk member port number.

ingress_ port Specifies the ingress port number.

unicast-distribution Generates debugging information related to unicast distribution.

ip *dmac smac* Generates debugging information related to unicast IPv4 and IPv6 distribution for the specified destination MAC address and source MAC address. The following variables are supported with this option.

dst_ipv4 Specifies the destination IPv4 address.

src_ipv4 Specifies the source IPv4 address.

dst_tcp | *udp_port*

Specifies the destination TCP or UDP port number.

src_tcp | *udp_port*

Specifies the source TCP or UDP port number.

dst_ipv6 Specifies the destination IPv6 address.

src_ipv6 Specifies the source IPv6 address.

non-ip *dmac smac*

Generates debugging information related to unicast non-IP distribution for the specified destination MAC address and source MAC address.

sw_ trunk_ ID Specifies the software trunk ID.

v4-device Generates debugging information related to the IPv4 device. The following parameters are supported with this option.

registered-multicast

Generates debugging information related to registered multicast distribution. The following variables must be specified with this option.

vlan Specifies the VLAN ID.

vidx Specifies the multicast ID.

routed-flow Generates debugging information related to routed flow distribution. The following variables must be specified with this option.

dst_ipv4 Specifies the destination IPv4 address.

src_ipv4 Specifies the source IPv4 address.

IPHdr_Protocol Specifies the protocol value in the IP header.

switched-flow ip

Generates debugging information related to switched flow IPv4 and IPv6 distribution. The following variables must be specified with this option.

dst_ipv4 Specifies the destination IPv4 address.

src_ipv4 Specifies the source IPv4 address.

switched-flow non-ip

Generates debugging information related to switched flow non-IP distribution. The following variables must be specified with this option.

dmac Specifies the destination MAC address.

smac Specifies the source MAC address

unregistered-multicast *vlan*

Generates debugging information related to unregistered multicast distribution for the specified VLAN ID.

v6-device Generates debugging information related to the IPv6 device. The following parameters are supported with this option.

routed-multicast-distribution *vlan*

Generates debugging information related to routed multicast distribution for the specified VLAN ID.

unicast-distribution

Generates debugging information related to unicast distribution.

ip Generates debugging information related to unicast IP distribution.

non-ip Generates debugging information related to unicast non-IP distribution.

Modes Privileged EXEC mode

Usage Guidelines The **debug pp-trunk-hash** *trunk_port* command is supported only on Brocade FCX and Brocade ICX devices.

The **debug pp-trunk-hash** *ingress_ port* command is supported only on Brocade FastIron SX devices.

Examples

For the unicast IPv4 distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/1/3 unicast-distribution ip 0000.0082.ba04
0000.00c9.5680 10.10.10.2 10.10.10.1 10 10
trunk hashed port: 1/1/2
```

For the unicast IPv6 distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/1/3 unicast-distribution ip 0000.0082.ba04
0000.00c9.5680 2001:DB8::100:10 2001:DB8::10:19 10 17
trunk hashed port: 1/1/1
```

For the unicast non-IP distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/1/3 unicast-distribution non-ip 0000.0082.ba04
0000.00c9.5680
trunk hashed port: 1/1/9
```

For the registered multicast distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/5 v4-device 1 registered-multicast 10 10
trunk hash port: 1/1
```

For the routed flow distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/5 v4-device 1 routed-flow 10.10.10.1 10.10.10.5 9
trunk hashed port: 1/13
```

For the switched flow IP distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/5 v4-device 1 switched-flow ip 10.10.10.1
10.10.10.2
trunk hashed port: 2/2
```

For the switched flow non-IP distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/5 v4-device 1 switched-flow non-ip 0000.00c9.5680
0000.0082.ba04
trunk hash port: 1/1
```

For the unregistered multicast distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/5 v4-device 1 unregistered-multicast 10
trunk hash port: 1/1
```

For the routed multicast distribution, the command output resembles the following example.

```
Brocade# debug pp-trunk-hash 1/1 v6-device 2 routed-multicast-distribution 20
trunk hash port: 1/3
```

History

Release	Command History
Release 08.0.00a	The command was introduced.

Related Commands

debug pp-bum-trunk-hash

debug pp-bum-trunk-hash

Traces and displays the output port through which a given packet can be sent out in a trunk for a broadcast, unknown-unicast, and multicast (BUM) flow.

Syntax **debug pp-bum-trunk-hash** *ingress_port sw_trunk_id dmac smac*

ipv4 *dst_ipv4 src_ipv4* [*dst_tcp* | *udp_port*] [*src_tcp* | *udp_port*] *vlan vidx*

ipv6 *flow_label dst_ipv6 src_ipv6* [*dst_tcp* | *udp_port*] [*src_tcp* | *udp_port*] *vlan vidx*

non-ip

no debug pp-bum-trunk-hash

Parameters

ingress_ port	Specifies the ingress port number.
sw_ trunk_ id	Specifies the software trunk ID.
dmac	Specifies the destination MAC address.
smac	Specifies the source MAC address.
ipv4	Generates debugging information related to IPv4 distribution for a BUM flow.
ipv6	Generates debugging information related to IPv6 distribution for a BUM flow. The following variables are supported with the ipv4 and ipv6 options.
dst_ipv4	Specifies the destination IPv4 address.
src_ipv4	Specifies the source IPv4 address.
dst_tcp | *udp_port*	Specifies the destination TCP or UDP port number.
src_tcp | *udp_port*	Specifies the source TCP or UDP port number.
vlan	Specifies the VLAN ID.
vidx	Specifies the multicast ID.
flow_label	Specifies the flow label in the IPv6 header.
dst_ipv6	Specifies the destination IPv6 address.
src_ipv6	Specifies the source IPv6 address.
non-ip	Generates debugging information related to non-IP distribution for a BUM flow.

Modes Privileged EXEC mode

Examples For the IPv4 BUM flow, the command output resembles the following example.

```
Brocade# debug pp-bum-trunk-hash 1/1/1 3 0000.00aa.aaaa 0000.00bb.1111 ipv4
10.1.1.1 10.1.1.2 10 10 1 100
Designated Trunk Port = 1/1/5
```

For the IPv6 BUM flow, the command output resembles the following example.

```
Brocade# debug pp-bum-trunk-hash 1/1/11 3 0000.00aa.aaaa 0000.00bb.1111 ipv6 128
2001:DB8::1:1 2001:DB8::1:1 10 10 1 100
Designated Trunk Port = 1/1/4
```

For the non-IP BUM flow, the command output resembles the following example.

```
Brocade# debug pp-bum-trunk-hash 1/1/1 3 0000.00aa.aaaa 0000.00bb.1111 non-ip
Designated Trunk Port = 1/1/5
```

History

Release	Command History
Release 08.0.00a	The command was introduced.

Related Commands

debug pp-trunk-hash

Chapter

IP Diagnostics

4

This chapter contains the following sections:

ARP debug commands

Address Resolution Protocol (ARP) is a standard protocol that enables a router to obtain the Media Access Control (MAC) address of an interface on another device when the router knows the IP address of the interface. ARP is enabled by default and cannot be disabled.

The following commands perform actions related to ARP debugging.

debug ip arp

Syntax: [no] **debug ip arp** *ip-addr*

This command activates the debuggging of ARP packets. The *ip-addr* variable specifies a particular route.

```
Brocade# debug ip arp 10.29.29.53
ARP:  packets debugging is on
Brocade# show arp

Total number of ARP entries: 3, maximum capacity: 4000
No.   IP Address       MAC Address    Type     Age Port           Status
1     10.29.29.53      0000.0034.1b60 Dynamic  0    2/1/15-2/1/16 Valid
2     10.200.200.45    0000.00cf.c400 Dynamic  0    1/1/13-1/1/14 Valid
3     10.30.30.31      0000.0023.0f00 Dynamic  0    2/1/9-2/1/10  Valid
```

```
Brocade# clear arp
next hop router 10.29.29.53 ARP mapping deleted
Brocade# ARP: sent packet oper=request, src 10.29.29.111 0000.0076.2c80: dst
10.29.29.53 0000.0000.0000: Port v29
ARP: sent packet oper=request, src 10.200.200.111 0000.0076.2c80: dst
10.200.200.45 0000.0000.0000: Port v2000
ARP: sent packet oper=request, src 10.30.30.111 0000.0076.2c80: dst 10.30.30.31
0000.0000.0000: Port v300
ARP: sent packet oper=request, src 10.20.67.111 0000.0076.2c80: dst 10.20.10.11
0000.0000.0000: Port mgmt1
```

debug ip arp inspection

Syntax: [no] **debug ip arp inspection**

This command enables the ARP inspection debugging.

```
Brocade# debug ip arp inspection
ARP:  inspection debugging is on
RX: rcvd ARP packet, brd 1, us 0, vlan 14 port 5/1/2
ARP: rcvd packet oper=request, src 10.3.3.13 0000.009d.8c69: dst 10.3.3.1
0000.0000.0000: Port e5/1/2
ARP: REQ inspection pass, 10.3.3.13->10.3.3.1, port 5/1/2
```

BGP debug commands

The following **debug ip bgp** commands display information about Border Gateway Protocol (BGP) IP transactions.

debug ip bgp

Syntax: [no] **debug ip bgp** *neighbor_ip_address*

This command enables common BGP debugging information to be displayed for all virtual routing and forwarding events (VRFs) or for a specific VRF. The *neighbor_ip_address* variable refers to the IP address of the immediate neighbor.

```
Brocade# debug ip bgp 10.1.34.10 updates
          BGP:  neighbor 10.1.34.10 debugging is on
Brocade# BGP: 10.1.34.10 rcv UPDATE w/attr: Origin=IGP
AS_PATH=AS_CONFED_SEQUENCE(3) 64519 NextHop=10.1.34.10 MED=0 LOCAL_PREF=100
BGP: 10.1.34.10 rcv UPDATE 10.1.22.0/24
BGP: 10.1.34.10 rcv UPDATE w/attr: Origin=IGP AS_PATH=AS_CONFED_SEQUENCE(3) 64519
NextHop 10.1.34.10 MED=0 LOCAL_PREF=100
BGP: 10.1.34.10 rcv UPDATE 10.1.23.0/24
```

debug ip bgp dampening

Syntax: [no] **debug ip bgp dampening**

This command displays information about dampening process configurations, route penalties, durations, restraint, and release. Command output resembles the following example.

```
Brocade# debug ip bgp dampening
BGP: 10.1.1.2 Decay 10.1.1.32/32, ostate <d>, oPnlty=6805, nPnlty=840, time=2712
BGP: 10.1.1.2 reuse_list_index=258, curr_offset=45
BGP: 10.1.1.2 10.1.1.32/32 not ready, state <d>, reuse_threshold=750,
new_index=61, offset=45
```

```
BGP: 10.1.1.2 Decay 10.1.1.32/32, ostate <d>, oPnlty=6805, nPnlty=840, time=2712
BGP: 10.1.1.2 reuse_list_index=258, curr_offset=45
BGP: 10.1.1.2 10.1.1.32/32 not ready, state <d>, reuse_threshold=750,
new_index=61, offset=45
BGP: 10.1.1.2 Decay 10.1.1.32/32, ostate <d>, oPnlty=6805, nPnlty=840, time=2712
BGP: 10.1.1.2 reuse_list_index=258, curr_offset=45
BGP: 10.1.1.2 10.1.1.32/32 not ready, state <d>, reuse_threshold=750,
new_index=61, offset=45
BGP: 10.1.1.2 Decay 10.1.1.32/32, ostate <d>, oPnlty=6805, nPnlty=840, time=2712
BGP: 10.1.1.2 reuse_list_index=258, curr_offset=45
BGP: 10.1.1.2 10.1.1.32/32 not ready, state <d>, reuse_threshold=750,
new_index=61, offset=45
BGP: 10.1.1.2 Decay 10.1.1.32/32, ostate <d>, oPnlty=6805, nPnlty=840, time=2712
BGP: 10.1.1.2 reuse_list_index=258, curr_offset=45
BGP: 10.1.1.2 10.1.1.32/32 not ready, state <d>, reuse_threshold=750,
new_index=61, offset=45
```

debug ip bgp events

Syntax: [no] debug ip bgp events

This command generates information about BGP events, such as connection attempts and keepalive timer activity.

```
Brocade# debug ip bgp events
          BGP:  events debugging is on
Brocade# BGP: 10.1.34.10 rcv notification: CEASE Message
BGP: 10.1.34.10 Peer went to IDLE state (Rcv Notification)
BGP: 10.1.35.10 rcv notification: CEASE Message
BGP: 10.1.35.10 Peer went to IDLE state (Rcv Notification)
BGP: 10.1.34.10 sending Graceful Restart cap, rbit 0, fbit 0, time 120, length 6
BGP: 10.1.35.10 sending Graceful Restart cap, rbit 0, fbit 0, time 120, length 6
BGP: 10.1.34.10 rcv GR capability afi/safi=1/1 fbit 0
BGP: 10.1.34.10 Peer went to ESTABLISHED state
BGP: 10.1.35.10 rcv GR capability afi/safi=1/1 fbit 0
BGP: 10.1.35.10 Peer went to ESTABLISHED state
BGP: 10.1.34.10 rcv UPDATE EOR (0), waiting EOR 0
BGP: 10.1.35.10 rcv UPDATE EOR (0), waiting EOR 0
BGP: 10.1.34.10 sending EOR (safi 0)...
BGP: 10.1.35.10 sending EOR (safi 0)...
```

debug ip bgp graceful-restart

Syntax: [no] debug ip bgp graceful-restart

Enable this command to receive information about BGP graceful restarts. The graceful restart feature minimizes disruptions in forwarding and route flapping when a router restarts.

```
Brocade# debug ip bgp graceful-restart
          BGP:  graceful-restart debugging is on
BGP: 10.1.251.6 save graceful restart parameters, #RIB_out 2 (safi 0)
10.1.251.6 RIB_out peer reset  #RIB_out 2 (safi 0)
 BGP: 10.1.251.6 sending Graceful Restart cap, rbit 0, fbit 0, time 120, length 6
BGP: 10.1.251.6 sending Graceful Restart cap, rbit 0, fbit 0, time 120, length 6
BGP: 10.1.251.6 sending Graceful Restart cap, rbit 0, fbit 0, time 120, length 6
BGP: 10.1.251.6 rcv GR capability afi/safi=1/1 fbit 1
BGP: 10.1.251.6 sending EOR (safi 0)...
```

debug ip bgp keepalives

Syntax: [no] **debug ip bgp keepalives**

Brocade devices use keepalives to collect information about applications and services. For example, you can configure a keepalive to continually monitor and report on the online status of a resource, such as BGP.

```
Brocade# debug ip bgp keepalives
BGP: 10.1.1.2 sending KEEPALIVE
BGP: 10.1.1.2 KEEPALIVE received
```

debug ip bgp updates

Syntax: [no] **debug ip bgp updates**

This command displays BGP receive, transmit, or receive and transmit update messages about debug processing.

```
Brocade# debug ip bgp updates
BGP: safi(0): 10.1.1.2 sending UPDATE w/attr: Origin=IGP AS_PATH=65538
NEXT_HOP=10.1.1.1 MED=1
BGP: (0): 10.1.1.2 sending UPDATE Label=0 2001:DB8:83:e8:00:04:78:64:10.1.1.1/32
BGP: 10.1.1.2 rcv bad UPDATE (saif 0) due to AS loop, take as implicit withdraw!
BGP: BGP: 10.1.1.2  rcv UPDATE w/attr: Origin=INCOMP AS_PATH= AS_SEQ(2) 3
NextHop=10.1.1.2 MED=30
BGP: (0): 10.1.1.2 rcv UPDATE 10.1.1.2/32
BGP: 10.1.1.2 rcv bad UPDATE (saif 0) due to AS loop, take as implicit withdraw!
BGP: BGP: 10.1.1.2  rcv UPDATE w/attr: Origin=IGP AS_PATH= AS_SEQ(2) 3
NextHop=10.1.1.2 MED=5
BGP: (0): 10.1.1.2 rcv UPDATE 10.10.10.10/32
BGP: (0): 10.1.1.2 rcv UPDATE 10.213.0.0/16
BGP: (0): 10.1.1.2 rcv UPDATE 10.200.200.200/32
BGP: (0): 10.1.1.2 rcv UPDATE 10.18.18.0/24
BGP: (0): 10.1.1.2 rcv UPDATE 10.1.1.32/32
```

DHCP snooping debug commands

Dynamic Host Configuration Protocol (DHCP) snooping enables the Brocade device to filter untrusted DHCP packets in a subnet. DHCP snooping can ward off man-in-the-middle (MiM) attacks, such as a malicious user posing as a DHCP server sending false DHCP server reply packets with the intention of misdirecting other users. DHCP snooping can also stop unauthorized DHCP servers and prevent errors due to user misconfiguration of the DHCP servers.

Often, DHCP snooping is used together with Dynamic ARP Inspection (DAI) and IP Source Guard.

debug ip dhcp-snooping

Syntax: [no] **debug ip dhcp-snooping**

This command enables debugging of the DHCP snooping activity.

```
Brocade# debug ip dhcp-snooping
Brocade(config-vif-11)# DHCP: snooping on for vlan 14, port 5/1/2
DHCP: rcv on port 5/1/2, intercept DHCP pkt, Discovr
DHCP: snooping on port 5/1/2, smac 0000.00ce.ac79, type 1, VRF 1
0.0.0.0->0000.00ce.ac79
DHCP Option82: Adding option 82
```

```
DHCP Option82: 20 bytes added.
DHCP Option82: Adding option 82
DHCP: snooping on for vlan 14, port 3/1/23
DHCP: rcv on port 3/1/23, intercept DHCP pkt, Offer
DHCP Option82: Removing option 82
DHCP Option82: 19 bytes removed.
DHCP Option82: Removing option 82
DHCP: snooping on for vlan 14, port 5/1/2
DHCP: rcv on port 5/1/2, intercept DHCP pkt, Request
DHCP: snooping on port 5/1/2, smac 0000.00ce.ac79, type 3, VRF 1
0.0.0.0->0000.00ce.ac79
DHCP Option82: Adding option 82
DHCP Option82: 20 bytes added.
DHCP Option82: Adding option 82
```

GRE debug commands

debug ip gre log

Syntax: [no] **debug ip gre log**

This command activates the tunnel events debugging.

```
Brocade# debug ip gre log
Apply mtu 1400 failed for tunnel 11 as all mtu profiles are used
Clearing PMTU for tunnel 11 due to ip mtu 1400 configuration
Tunnel Create Sync-Receive event for tunnel: 11
Tunnel Create Sync-Successful for tunnel: 11
Tunnel Update TS Sync-Receive event for tunnel: 11
Tunnel Update TS Sync-Successful for tunnel: 11
Tunnel Update Nhop Sync-Receive event for tunnel: 11
Tunnel Update Nhop Sync-Successful for tunnel: 11
Tunnel Update TTI Sync-Receive event for tunnel: 11
Tunnel Port Up Sync-Receive event for tunnel: 11
Tunnel Port Up Sync-Successful for tunnel: 11
Tunnel Port Down Sync-Receive event for tunnel: 11
Tunnel Port Down Sync-Successful for tunnel: 11
Tunnel Update MTU Sync-Receive event for tunnel: 11
Tunnel Update MTU Sync-Successful for tunnel: 11
Tunnel Delete Sync-Receive event for tunnel: 11
Tunnel Delete Sync-Successful for tunnel: 11
Tunnel Delete Nhop Sync-Receive event for tunnel: 11
Tunnel Delete Nhop Sync-Successful for tunnel: 11
Tunnel Delete TTI Sync-Receive event for tunnel: 11
Tunnel Delete Origination entry Sync-Receive event for tunnel: 11
Tunnel Delete Origination entry Sync-Successful for tunnel: 11
process_one_l3_unicast_update: entryType 6
Fill Sync Tunnel Entry-Bad parameters. Failed for tunnel: 11
Fill Sync Tunnel Entry-Bad TS entry. Failed for tunnel: 11
pack_tunnel_entry_change called with tunnel_index: 11 operation:
TUNNEL_SYNC_UPDATE_NHOP
pack_tunnel_entry_list called with tunnel_index: 11 operation:
TUNNEL_SYNC_CREATE_ALL, from_beginning: 1
Sync-ing tunnel entry 11
[pp_puma_tunnel_hotswap_insert] No tunnel start info for tunnel 11
DEBUG 0x8: Tunnel 11 already deleted
DEBUG 0x8: Tunnel 11 pp_delete_tunnel_entry
```

```
DEBUG 0x8: Can not find route for tunnel id 11, hw index 23, src 10.22.33.254,
dest 10.22.33.252
DEBUG 0x8: create GRE tunnel origination because find route for tunnel id 11, hw
index 23, src 10.22.33.254, dest 10.22.33.252
DEBUG 0x8: Route for tunnel id 11 is via another tunnel or management port 22
DEBUG 0x8: Direct route for tunnel id 11 dest 10.22.33.252 has unresolved ARP
DEBUG 0x8: indirect route for tunnel id 11 dest 10.22.33.252 next hop 10.11.25.5
has unresolved ARP
DEBUG 0x8: indirect route for tunnel id 11 destination 10.22.33.252 next hop
10.11.25.5 has resolved ARP - CMD => route
DEBUG 0x8: Set next hop to Drop for  tunnel id 11, hw index 23, src 10.22.33.254,
dest 10.22.33.252
DEBUG 0x8: Tunnel 11 - next hop 10.11.25.5  has no link-layer
DEBUG 0x8: bring tunnel id 11 up with drop
DEBUG 0x8: Create tunnel id 11 with next hop 185276677 - old next hop is 125276633
DEBUG 0x8: bring tunnel id 11 up
Update_hw_routes for Tunnel 11
Path MTU: Tunnel 11 - reset path mtu timer 456 (3344)
Path MTU: Tunnel 11 - failed to reset path mtu timer 456 (3344)
```

debug ip gre keepalive

Syntax: [no] **debug ip gre keepalive**

This command activates the Generic Routing Encapsulation (GRE) keepalive debugging.

```
Brocade# debug ip gre keepalive
GRE_MP: Keepalive Bring DOWN GRE Tunnel  11
GRE_MP: Keepalive Bring UP GRE on Tunnel 11 - port is 2/1/2
GRE_MP: RX Keepalive packet on tnnl 11 src 10.22.33.254, dst 10.22.33.252
GRE_MP: TX Keepalive packet on tnnl 11 we need to find outgoing port
GRE_MP: TX Keepalive packet on tnnl 11 - outgoing_port is 2/1/2 with mac
0000.00bb.ef40 vlan 2
GRE_MP_KEEPALIVE: NO FREE BUFFER AVAILABLE
GRE_MP: For Tunnel 11, Keepalive timeout after 3 seconds
GRE_MP: For Tunnel 11, stop keep-alive
GRE_MP: For Tunnel 11, start keep-alive
GRE_MP: enqueue Keepalive packet on tunnel 11 at index 23
GRE_MP: dequeue Keepalive packet on tunnel 11 at index 23
```

debug ip gre packet

Syntax: [no] **debug ip gre packet**

This command activates the debugging of GRE packet processing.

```
Brocade# debug ip gre packet
GRE: Error - IP GRE packet with invalid Ptype 0x1A4
GRE: Error - IP GRE packet with invalid first word 0x2C
GRE: Error - IP GRE packet with invalid first word 0x2C
GRE fragment : Rx IP GRE Pkt: src 10.22.33.254, dst 224.0.0.5, len 64 id 324
GRE Packet too big - need extra processing: Rx IP GRE Pkt: src 10.22.33.254, dst
224.0.0.5, len 64
GRE Reassembly: Rx IP GRE Pkt: src 10.22.33.254, dst 224.0.0.5, len 64
GRE : Rx IP GRE Pkt: src 10.22.33.254, dst 224.0.0.5, len 64
GRE: Error - Rx IP Pkt with invalid Inner IPv4 header
GRE: Error - Rx IP Pkt with 2 GRE headers: Inner src 10.22.33.254, Inner dst
224.0.0.5, len 64
GRE: Dropping the packet as GRE tnnl 11  is not UP
GRE_LP: Error - LP TX GRE packet is in LOOP
```

```
GRE : FORWARD IP to GRE tunnel 11 - Pkt: src 10.22.33.254, dst 224.0.0.5, len 64
GRE : Failed to FORWARD IP to GRE tunnel 11  - Pkt: src 10.22.33.254, dst
224.0.0.5, No route
```

ICMP debug commands

The following **debug ip icmp** commands display information about Internal Control Message Protocol (ICMP) transactions. These commands are useful in determining if a router is sending or receiving ICMP messages, and for troubleshooting end-to-end connections.

debug ip icmp events

Syntax: [no] **debug ip icmp events**

This command activates the ICMP events debugging.

```
Brocade# debug ip icmp events
         ICMP:  events debugging is on

Brocade# ICMP: rcvd echo request packet of length 40 from 10.44.22.11
ICMP: send echo reply packet of length 60 to 10.44.22.11
ICMP: rcvd echo request packet of length 40 from 10.44.22.11
ICMP: send echo reply packet of length 60 to 10.44.22.11
ICMP: rcvd echo request packet of length 40 from 10.44.22.11
ICMP: send echo reply packet of length 60 to 10.44.22.11
ICMP: rcvd echo request packet of length 40 from 10.44.22.11
ICMP: send echo reply packet of length 60 to 10.44.22.11

Brocade# no debug ip icmp events
         ICMP:  events debugging is off
```

debug ip icmp packets

Syntax: [no] **debug ip icmp packets**

This command activates the ICMP packets debugging.

```
Brocade# debug ip icmp packets
         ICMP:  packets debugging is on
Brocade# !SR_SWITCH_ROUTER!ICMP_DEBUG_RX
ICMP: Received message from 10.44.22.11 to 10.44.22.36 port 1/1/1 size 40
!SR_SWITCH_ROUTER!ICMP_DEBUG_RX
ICMP: Received message from 10.44.22.11 to 10.44.22.36 port 1/1/1 size 40
!SR_SWITCH_ROUTER!ICMP_DEBUG_RX
ICMP: Received message from 10.44.22.11 to 10.44.22.36 port 1/1/1 size 40
!SR_SWITCH_ROUTER!ICMP_DEBUG_RX
ICMP: Received message from 10.44.22.11 to 10.44.22.36 port 1/1/1 size 40

Brocade# no debug ip icmp packets
         ICMP:  packets debugging is off
```

OSPF debug commands

The following debug commands display information about Open Shortest Path First (OSPF) transactions.

debug ip ospf

Syntax: [**no**] **debug ip ospf** *A.B.C.D*

This command generates OSPF debugging information for the specified IP address. Output indicates state transitions, hello packets received, LSA acknowledgements received, LSA processing, flooding information, and database descriptions, as shown in the following example.

```
Brocade# debug ip ospf 10.1.1.2
 OSPF: debug ospf neighbor 10.1.1.2
Brocade(config)# OSPF: rcvd hello from 10.1.1.2 area 0 on interface 10.1.1.1,
state DR, DR 10.1.1.1, BDR 10.1.1.2
OSPF: Neighbor 10.1.1.2, int v115,  state FULL processing event HELLO_RECEIVED
OSPF: rcvd hello from 10.1.1.2 area 0 on interface 10.1.1.1, state DR, DR
10.1.1.1, BDR 10.1.1.2
OSPF: Neighbor 10.1.1.2, int v115,  state FULL processing event HELLO_RECEIVED
OSPF: rcvd hello from 10.1.1.2 area 0 on interface 10.1.1.1, state DR, DR
10.1.1.1, BDR 10.1.1.2
OSPF: Neighbor 10.1.1.2, int v115,  state FULL processing event HELLO_RECEIVED
```

debug ip ospf adj

Syntax: [**no**] **debug ip ospf adj** *decimal*

This command displays information about OSPF adjacencies and authentication, including designated router (DR) and backup designated router (BDR) elections, sent and received hello packets, neighbor state transitions, and database description information. The *decimal* variable refers to a specific adjacency event.

```
Brocade# debug ip ospf adj
OSPF:  adjacency events debugging is on
Brocade# debug ip ospf adj OSPF: send hello on area 0 interface 10.1.1.1
OSPF: rcvd hello from 10.1.1.2 area 0 on interface 10.1.1.1, state DR, DR
10.1.1.1, BDR 10.1.1.2
OSPF: Neighbor 10.1.1.2, int v115,  state FULL processing event HELLO_RECEIVED
OSPF: send hello on area 0 interface 10.1.1.1
OSPF: rcvd hello from 10.1.1.2 area 0 on interface 10.1.1.1, state DR, DR
10.1.1.1, BDR 10.1.1.2
OSPF: Neighbor 10.1.1.2, int v115,  state FULL processing event HELLO_RECEIVED
OSPF: send hello on area 0 interface 10.1.1.1
OSPF: rcvd hello from 10.1.1.2 area 0 on interface 10.1.1.1, state DR, DR
10.1.1.1, BDR 10.1.1.2
OSPF: Neighbor 10.1.1.2, int v115,  state FULL processing event HELLO_RECEIVED
```

debug ip ospf all-vrfs route

Syntax: [**no**] **debug ip ospf all-vrfs route**

This command displays OSPF debugging information for all VPN routing and forwarding activity.

```
Brocade# debug ip ospf all-vrfs route
SPF: redistribute into ospf 10.3.13.0 with ffffff00 forwarding address 10.4.4.3
OSPF: originate external lsa 10.3.13.0 with ffffff00
OSPF: Originate external advs ext fwding address 10.4.4.3
```

debug ip ospf error

Syntax: [no] **debug ip ospf error**

This command reports the receipt of OSPF packets with errors, or mismatches between hello packet options.

```
Brocade# debug ip ospf error
OSPF:  errors debugging is on
Brocade(config-vif-9)# Debug: Jan  1 03:37:11 OSPF: invalid header or unable to
find neighbor, drop
Debug: Jan  1 03:37:11 OSPF: recv from:10.9.9.2 Intf:ve 9  Hello  L:48 A:1
Rid:10.2.2.2 DR:10.9.9.2 BDR:0.0.0.0
Debug: Jan  1 03:37:23 OSPF: invalid header or unable to find neighbor, drop
Debug: Jan  1 03:37:23 OSPF: recv from:10.9.9.2 Intf:ve 9  Hello  L:48 A:1
Rid:10.2.2.2 DR:10.9.9.2 BDR:0.0.0.0
Debug: Jan  1 03:37:34 OSPF: invalid header or unable to find neighbor, drop
Debug: Jan  1 03:37:34 OSPF: recv from:10.9.9.2 Intf:ve 9  Hello  L:48 A:1
Rid:10.2.2.2 DR:10.9.9.2 BDR:0.0.0.0
```

debug ip ospf event

Syntax: [no] **debug ip ospf event**

This command displays information about internal OSPF events related to configuration or interaction with the standby management processor and interface state transitions.

```
Brocade# debug ip ospf event
         OSPF:  events debugging is on
Brocade# OSPF: Interface ve 18 (10.1.18.1) state Waiting processing event Wait
Timer
OSPF: DR/BDR election for 10.1.18.1 on ve 18
OSPF: Neighbor 10.213.213.213 int v18, state 2_WAY processing event ADJACENCY_OK
OSPF: send DBD to 10.213.213.213 on ve 18 flag 0x7 seq 0x18a3b9 len 32
OSPF: elect BDR(backup designated router): Router ID 10.213.213.213 IP interface
10.1.18.2
OSPF: elect DR(designated router): Router ID 10.10.10.77, IP interface 10.1.18.1
OSPF: Neighbor 10.116.116.1 int 2/1/21, state FULL processing event HELLO_RECEIVED
OSPF: Neighbor 10.116.116.1 int v511, state FULL processing event HELLO_RECEIVED
OSPF: Neighbor 10.116.116.1 int v911, state FULL processing event HELLO_RECEIVED
OSPF: Neighbor 41.41.41.41 int v35, state FULL processing event HELLO_RECEIVED
OSPF: Neighbor 10.213.213.213 int 2/1/14, state 2_WAY processing event
HELLO_RECEIVED
OSPF: Neighbor 10.116.116.1 int v511, state FULL processing event HELLO_RECEIVED
OSPF: send DBD to 10.213.213.213 on ve 18 flag 0x7 seq 0x18a3b9 len 32
OSPF: Neighbor 49.2.3.4 int v34, state FULL processing event HELLO_RECEIVED
OSPF: Neighbor 10.213.213.213 int v18, state EXCHANGE_START processing event
HELLO_RECEIVED
OSPF: Neighbor 10.213.213.213 int 4/1/12, state FULL processing event
HELLO_RECEIVED

Brocade# OSPF: Interface ethernet 4/1/12 (10.1.51.2) state Down processing event
Interface Up
virtual interface 10.1.51.2 up, state changed to Other from Point To Point
no deOSPF: Neighbor 10.213.213.213 int 4/1/12, state DOWN processing event
HELLO_RECEIVED
OSPF: Neighbor 10.213.213.213 int 4/1/12, state INITIALIZING processing event
ONE_WAY
bug ip OSPF: Neighbor 10.213.213.213 int 4/1/12, state INITIALIZING processing
event HELLO_RECEIVED
```

```
OSPF: Neighbor 10.213.213.213 int 4/1/12, state INITIALIZING processing event
ONE_WAY
ospOSPF: send DBD to 10.213.213.213 on ve 18 flag 0x7 seq 0x18a3b9 len 32
f event
Brocade# no debug ip ospf event
       OSPF:  events debugging is off
```

debug ip ospf flood

Syntax: [no] **debug ip ospf flood**

This command displays information about LSA flooding activity as shown in the following example.

```
Brocade# debug ip ospf flood
OSPF:  flooding debugging is on
Brocade(config-if-e1000-1/1/15)#OSPF: flood LSA Type:1 AdvRtr:10.2.1.1 Age:0
LsId:10.2.1.1
OSPF: flood advertisement 10.2.1.1 throughout a specific area = 0
OSPF: flood LSA Type:1 AdvRtr:10.2.1.1 Age:0 LsId:10.2.1.1
OSPF: flood advertisement 10.2.1.1 throughout a specific area = 0
OSPF: flooding type 1 advertisement out interface 10.1.1.1
OSPF: attempting to flood rcvd LSA area = 0 interface type = 1
OSPF: flood LSA Type:1 AdvRtr:10.3.1.1 Age:1 LsId:10.3.1.1
OSPF: flood advertisement 10.3.1.1 throughout a specific area = 0
OSPF: attempting to flood rcvd LSA area = 0 interface type = 1
OSPF: flood LSA Type:2 AdvRtr:10.3.1.1 Age:3600 LsId:10.1.1.2
OSPF: flood advertisement 10.1.1.2 throughout a specific area = 0
OSPF: flood LSA Type:1 AdvRtr:10.2.1.1 Age:0 LsId:10.2.1.1
OSPF: flood advertisement 10.2.1.1 throughout a specific area = 0
OSPF: flooding type 1 advertisement out interface 10.1.1.1
OSPF: attempting to flood rcvd LSA area = 0 interface type = 1
OSPF: flood LSA Type:1 AdvRtr:10.3.1.1 Age:1 LsId:10.3.1.1
OSPF: flood advertisement 10.3.1.1 throughout a specific area = 0
OSPF: attempting to flood rcvd LSA area = 0 interface type = 1
OSPF: flood LSA Type:2 AdvRtr:10.3.1.1 Age:1 LsId:10.1.1.2
OSPF: flood advertisement 10.1.1.2 throughout a specific area = 0
```

debug ip ospf graceful-restart

Syntax: [no] **debug ip ospf graceful-restart**

Enable this command to receive information about OSPF graceful restart events, including restart phases, graceful Link-State Advertisement (LSA) transmit and receive activity, and syslog messages.

```
Brocade# debug ip ospf graceful-restart
         OSPF:  graceful-restart debugging is on
Brocade# LSA flush rcvd Type:4 AdvRtr:83.83.10.11 LsId:10.213.213.213
LSA flush rcvd Type:4 AdvRtr:83.83.10.11 LsId:10.213.213.213
LSA flush rcvd Type:4 AdvRtr:10.116.116.1 LsId:10.205.205.205
LSA flush rcvd Type:4 AdvRtr:10.116.116.1 LsId:10.205.205.205
rcv GRACE LSA from 10.1.14.1, age 0, Adv 10.213.213.213
 install new GraceLSA, int 269, neighbor 10.1.14.1, age 0
rcv Grace_LSA from 10.1.14.1, area 0.0.0.10
Recvd grace lsa id=50331648 state=8 0x2dd5d5d5 0x2dd5d5d5 age=0 gr-state=0
 neighbor 10.1.14.1 entering graceful restart state, timer 120, lsa age 0, max
120, helping 0
flood grace LSA, AdvRtr:10.213.213.213,  Age:0
rcv GRACE LSA from 10.1.51.1, age 0, Adv 10.213.213.213
 install new GraceLSA, int 779, neighbor 10.1.51.1, age 0
```

```
rcv Grace_LSA from 10.1.51.1, area 0.0.0.10
Recvd grace lsa id=50331648 state=8 0x2dd5d5d5 0x2dd5d5d5 age=0 gr-state=0
 neighbor 10.1.51.1 entering graceful restart state, timer 120, lsa age 0, max
120, helping 0
flood grace LSA, AdvRtr:10.213.213.213,  Age:0
rcv GRACE LSA from 10.1.18.2, age 0, Adv 10.213.213.213
 install new GraceLSA, int 2050, neighbor 10.1.18.2, age 0
rcv Grace_LSA from 10.1.18.2, area 0.0.0.10
Recvd grace lsa id=50331648 state=8 0x2dd5d5d5 0x2dd5d5d5 age=0 gr-state=0
 neighbor 10.1.18.2 entering graceful restart state, timer 120, lsa age 0, max
120, helping 0
flood grace LSA, AdvRtr:10.213.213.213,  Age:0
rcv GRACE LSA from 10.1.14.1, age 0, Adv 10.213.213.213
Update same instance GRACE LSA age to 0 in database, refresh neighbor 10.1.14.1
restart timer to 120
rcv GRACE LSA from 10.1.51.1, age 0, Adv 10.213.213.213
Update same instance GRACE LSA age to 0 in database, refresh neighbor 10.1.51.1
restart timer to 120
rcv GRACE LSA from 10.1.18.2, age 0, Adv 10.213.213.213
Update same instance GRACE LSA age to 0 in database, refresh neighbor 10.1.18.2
restart timer to 120
rcv GRACE LSA from 10.1.14.1, age 0, Adv 10.213.213.213
Update same instance GRACE LSA age to 0 in database, refresh neighbor 10.1.14.1
restart timer to 120
rcv GRACE LSA from 10.1.51.1, age 0, Adv 10.213.213.213
Update same instance GRACE LSA age to 0 in database, refresh neighbor 10.1.51.1
restart timer to 120
rcv GRACE LSA from 10.1.18.2, age 0, Adv 10.213.213.213
Update same instance GRACE LSA age to 0 in database, refresh neighbor 10.1.18.2
restart timer to 120

Brocade# no debug ip ospf graceful-restart
         OSPF:  graceful-restart debugging is off
```

debug ip ospf log-empty-lsa

Syntax: [no]debug ip ospf log-empty-lsa

This command displays information about empty link state advertisements (LSAs) as shown in the following example.

```
Brocade# debug ip ospf log-empty-lsa
Brocade(config-if-e1000-1/1/15)# debug ip ospf log-empty-lsa
         OSPF:  empty-LSA logging debugging is on
Brocade(config-if-e1000-1/1/15)# debug ip ospf flOSPF: originate router LSA, area
0
OSPF: No difference found, restart 0
OSPF: send ls request to neighbor 10.1.1.2, retran 0
OSPF: sending ls request last size 36, count 1
OSPF: LSA rcvd Type:1 AdvRtr:10.2.1.1 Age:297 LsId:10.2.1.1 Seq-Num 8000000e from
Neighbor 10.1.1.2
OSPF: Received  self originated LSA type 1 with id = 10.2.1.1 Seq-Num 8000000e
OSPF: install a new lsa, type 1, ls_id 10.2.1.1, age 0, seq 8000000f area-id 0
OSPF:  NSR Sync ACK received for LSA
OSPF:  ls_header.id 10.2.1.1 type 1 ToBesyncedState 2
OSPF: NSR : Sync node add, type 1, ls_id 10.2.1.1, age 0, seq 8000000f
OSPF: send_ls_update to interface 10.1.1.1 (224.0.0.6)  tb 962478204, retran 0
OSPF: tx LSA Type:1 AdvRtr:10.2.1.1 Age:1 LsId:10.2.1.1
OSPF: originate router LSA, area 0
```

```
OSPF:  originate_router_links_advertisement gen new instance set 10.2.1.1 (not
sent - wait for MinLSInterval)
OSPF: LSA rcvd Type:1 AdvRtr:10.3.1.1 Age:1 LsId:10.3.1.1 Seq-Num 8000000f from
Neighbor 10.1.1.2
OSPF: install a new lsa, type 1, ls_id 10.3.1.1, age 1, seq 8000000f area-id 0
OSPF:  NSR Sync ACK received for LSA
OSPF:  ls_header.id 10.3.1.1 type 1 ToBesyncedState 2
OSPF: NSR : Sync node add, type 1, ls_id 10.3.1.1, age 1, seq 8000000f
OSPF: LSA rcvd Type:2 AdvRtr:10.3.1.1 Age:3600 LsId:115.1.1.2 Seq-Num 80000005
from Neighbor 10.1.1.2
OSPF: install a new lsa, type 2, ls_id 10.1.1.2, age 3600, seq 80000005 area-id 0
OSPF:  NSR Sync ACK received for LSA
OSPF:  ls_header.id 105.1.1.2 type 2 ToBesyncedState 2
OSPF: NSR : Sync node add, type 2, ls_id 10.1.1.2, age 3600, seq 80000005
oOSPF: trying age out LSA, id 10.1.1.2, len 32, type 2, from 10.3.1.1, age 3600
genNewLsa 0 area 0
OSPF: age out and remove lsa data base 10.1.1.2
odsOSPF: OSPF TIMER: Minlsa: sptr_database_entry generate_new_instance == TRUE
for 1.2.1.1
OSPF: originate_delayed LSA, type 1, parm1: 0x0
OSPF: originate router LSA, area 0
OSPF: difference found, restart 0
OSPF: install a new lsa, type 1, ls_id 10.2.1.1, age 0, seq 80000010 area-id 0
OSPF:  NSR Sync ACK received for LSA
OSPF:  ls_header.id 10.2.1.1 type 1 ToBesyncedState 2
OSPF: NSR : Sync node add, type 1, ls_id 10.2.1.1, age 0, seq 80000010
OSPF: OSPF: rcv LSA ack from 10.1.1.2, type 1, id 1.2.1.1, seq 0x8000000f, adv
10.2.1.1, age 1
OSPF: send_ls_update to interface 10.1.1.1 (224.0.0.6)  tb 1017253324, retran 0
OSPF: tx LSA Type:1 AdvRtr:10.2.1.1 Age:1 LsId:10.2.1.1
OSPF: LSA rcvd Type:1 AdvRtr:10.3.1.1 Age:1 LsId:10.3.1.1 Seq-Num 80000010 from
Neighbor 10.1.1.2
OSPF: install a new lsa, type 1, ls_id 10.3.1.1, age 1, seq 80000010 area-id 0
OSPF:  NSR Sync ACK received for LSA
OSPF:  ls_header.id 10.3.1.1 type 1 ToBesyncedState 2
OSPF: NSR : Sync node add, type 1, ls_id 10.3.1.1, age 1, seq 80000010
OSPF: LSA rcvd Type:2 AdvRtr:10.3.1.1 Age:1 LsId:10.1.1.2 Seq-Num 80000006 from
Neighbor 10.1.1.2
OSPF: install a new lsa, type 2, ls_id 10.1.1.2, age 1, seq 80000006 area-id 0
OSPF:  NSR Sync ACK received for LSA
OSPF:  ls_header.id 10.1.1.2 type 2 ToBesyncedState 2
OSPF: NSR : Sync node add, type 2, ls_id 10.1.1.2, age 1, seq 80000006
OSPF: OSPF: rcv LSA ack from 10.1.1.2, type 1, id 10.2.1.1, seq 0x80000010, adv
10.2.1.1, age 1
OSPF: originate router LSA, area 0
OSPF: No difference found, restart 0
```

debug ip ospf lsa-generation

Syntax: [no] **debug ip ospf lsa-generation**

This command generates information about LSAs as shown in the following example.

```
Brocade# debug ip ospf lsa-generation
OSPF:  lsa generation debugging is on
Brocade(config-if-e1000-1/1/15)#OSPF: originate router LSA, area 0
OSPF: trying age out LSA, id 10.2.1.1, len 36, type 1, from 10.2.1.1, age 3600
genNewLsa 0 area 0
OSPF: age out and remove lsa data base 10.2.1.1
```

debug ip ospf lsa-id

Syntax: [no] **debug ip ospf lsa-id**

This command generates information about OSPF LSA IDs as shown in the following example.

```
Brocade# debug ip ospf lsa-id
Debug message destination:  Console
IP Routing:
         OSPF:  flooding debugging is on
         OSPF:  lsa generation debugging is on
         OSPF:  ls-id 10.2.2.2 debugging is on

Brocade(config-ospf-router)# Debug: Jan  1 04:12:50 OSPF: LSA rcvd Type:1
AdvRtr:10.2.2.2 Age:1 LsId:10.2.2.2 Seq-Num 8000000e from Neighbor 10.9.9.2
Debug: Jan  1 04:12:50 OSPF: install a new lsa, type 1, ls_id 10.2.2.2, age 1, seq
8000000e area-id 0
Debug: Jan  1 04:12:50 OSPF:  NSR Sync ACK received for LSA
Debug: Jan  1 04:12:50 OSPF:  ls_header.id 10.2.2.2 type 1 ToBesyncedState 2
Debug: Jan  1 04:12:50 OSPF: NSR : Sync node add, type 1, ls_id 10.2.2.2, age 1,
seq 8000000e
Debug: Jan  1 04:12:50 OSPF: attempting to flood rcvd LSA area = 0 interface type
= 1
Debug: Jan  1 04:12:50 OSPF: flood LSA Type:1 AdvRtr:10.2.2.2 Age:1 LsId:10.2.2.2
Debug: Jan  1 04:12:50 OSPF: flood advertisement 10.2.2.2 throughout a specific
area = 0
Debug: Jan  1 04:12:51 OSPF: LSA rcvd Type:1 AdvRtr:10.2.2.2 Age:1 LsId:10.2.2.2
Seq-Num 8000000f from Neighbor 10.9.9.2
Debug: Jan  1 04:12:51 OSPF: install a new lsa, type 1, ls_id 10.2.2.2, age 1, seq
8000000f area-id 0
Debug: Jan  1 04:12:51 OSPF:  NSR Sync ACK received for LSA
Debug: Jan  1 04:12:51 OSPF:  ls_header.id 10.2.2.2 type 1 ToBesyncedState 2
Debug: Jan  1 04:12:51 OSPF: NSR : Sync node add, type 1, ls_id 10.2.2.2, age 1,
seq 8000000f
Debug: Jan  1 04:12:51 OSPF: attempting to flood rcvd LSA area = 0 interface type
= 1
Debug: Jan  1 04:12:51 OSPF: flood LSA Type:1 AdvRtr:10.2.2.2 Age:1 LsId:10.2.2.2
Debug: Jan  1 04:12:51 OSPF: flood advertisement 10.2.2.2 throughout a specific
area = 0
```

NOTE
The **debug ip ospf ls-id** command must be enabled with **debug ip ospf flood** or **debug ip ospf lsa-generation** command.

debug ip ospf max-metric

Syntax: [no] **debug ip ospf max-metric**

This command displays information about a max-metric configuration.

```
Brocade# debug ip ospf max-metric
OSPF:  max-metric debugging is on
Brocade(config-ospf-router)# max-metric router-lsa all-lsas
Debug: Jan  1 04:20:05 OSPF: Max-metric advertisement started due to configuration
change for vrf 0
```

debug ip ospf packet

Syntax: [no] **debug ip ospf packet** [**detail** *decimal* | **in** | **out** | **peer** *ip-addr* | **port** [**ethernet** *stackid/slot/port* | **ve** *decimal*] **src-ip** *ip-addr* | **type** [**ack** | **dd** | **hello** | **request** | **update**]]

- **detail** *decimal*—Refers to the level by bit 0 and 1 and stack trace send by bit 2.
- **in**—Refers only to the input.
- **out**—Refers only to the output.
- **peer** *ip-addr*—Matches with the peer (advertisement router).
- **port**—Matches with the I/O port.
- **ethernet** *stackid/slot/port*—Refers to the stack ID, slot, or Ethernet port.
- **ve** *decimal*—Refers to the number of the virtual Ethernet interface.
- **src-ip** *ip-addr*—Matches with the **src-ip** option and only to the input.
- **type**—Matches with the packet type.
- **ack**—Refers to the acknowledgement of the packets received.
- **dd**—Refers to the number of days.
- **hello**—Refers to the hello interval period.
- **request**—Refers to the request sent from a client to a server.
- **update**—Refers to the update to the packets.

This command generates information about the OSPF packets.

```
Brocade# debug ip ospf packet
Brocade# OSPF: recv from:10.1.14.1 to 224.0.0.5 Intf:e 2/1/14 LS-Ack L:104 Auth:0
ID:10.213.213.213

OSPF: recv from:10.1.18.2 to 224.0.0.5 Intf:ve 18  LS-Ack L:104 Auth:0
ID:10.213.213.213

OSPF: send to:224.0.0.5 Intf:ve 36  Hello  L:44 Auth:0 ID:10.10.10.77 DR:10.36.2.1
BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:ve 36  Hello  L:44 Auth:0 ID:10.10.10.77
DR:10.36.100.1 BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:ve 511  Hello  L:48 Auth:0 ID:10.10.10.77
DR:10.1.251.7 BDR:10.1.251.6
OSPF: send to:224.0.0.5 Intf:e 2/1/14 Hello  L:48 Auth:0 ID:10.10.10.77
DR:10.1.14.2 BDR:10.1.14.1
OSPF: send to:224.0.0.5 Intf:e 2/1/21 LS-Ack L:104 Auth:0 ID:10.10.10.77

OSPF: send to:224.0.0.5 Intf:e 2/1/21 Hello  L:48 Auth:0 ID:10.10.10.77 DR:0.0.0.0
BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:ve 35  Hello  L:48 Auth:0 ID:10.10.10.77
DR:10.1.35.16 BDR:10.1.35.15
OSPF: send to:10.1.251.6 Intf:ve 511  Hello  L:48 Auth:0 ID:10.10.10.77 DR:0.0.0.0
BDR:0.0.0.0
OSPF: send to:10.1.18.2 Intf:e 4/1/12 Hello  L:48 Auth:0 ID:10.10.10.77 DR:0.0.0.0
BDR:0.0.0.0
OSPF: recv from:10.1.35.15 to 224.0.0.5 Intf:ve 35  LS-Ack L:104 Auth:0
ID:41.41.41.41

OSPF: recv from:10.1.251.18 to 10.1.251.17 Intf:e 2/1/21 Hello  L:48 Auth:0
ID:10.116.116.1 DR:0.0.0.0 BDR:0.0.0.0
OSPF: recv from:10.1.34.10 to 224.0.0.5 Intf:ve 34  LS-Ack L:104 Auth:0
ID:49.2.3.4
```

```
OSPF: recv from:10.1.34.10 to 224.0.0.5 Intf:ve 34  Hello  L:48 Auth:0 ID:49.2.3.4
DR:10.1.34.16 BDR:10.1.34.10
OSPF: recv from:10.1.251.18 to 224.0.0.5 Intf:e 2/1/21 LS-Upd L:136 Auth:0
ID:10.116.116.1 Cnt:3

OSPF: send to:224.0.0.5 Intf:ve 18  LS-Upd L:136 Auth:0 ID:10.10.10.77 Cnt:3
OSPF: send to:224.0.0.5 Intf:e 2/1/14 LS-Upd L:136 Auth:0 ID:10.10.10.77 Cnt:3
OSPF: send to:224.0.0.5 Intf:e 4/1/12 LS-Upd L:136 Auth:0 ID:10.10.10.77 Cnt:3
OSPF: send to:224.0.0.5 Intf:ve 34  LS-Upd L:136 Auth:0 ID:10.10.10.77 Cnt:3
OSPF: send to:224.0.0.5 Intf:ve 35  LS-Upd L:136 Auth:0 ID:10.10.10.77 Cnt:3
OSPF: recv from:10.1.251.6 to 224.0.0.5 Intf:ve 511  LS-Upd L:136 Auth:0
ID:10.116.116.1 Cnt:3
OSPF: send to:10.1.251.6 Intf:ve 511  LS-Ack L:84 Auth:0 ID:10.10.10.77
OSPF: recv from:10.1.91.18 to 224.0.0.5 Intf:ve 911  LS-Upd L:136 Auth:0
ID:10.116.116.1 Cnt:3
OSPF: send to:10.1.91.18 Intf:ve 911  LS-Ack L:84 Auth:0 ID:10.10.10.77
OSPF: recv from:10.1.14.1 to 224.0.0.5 Intf:e 2/1/14 LS-Upd L:136 Auth:0
ID:10.213.213.213 Cnt:3
OSPF: recv from:10.1.51.1 to 224.0.0.5 Intf:e 4/1/12 LS-Upd L:136 Auth:0
ID:10.213.213.213 Cnt:3
OSPF: recv from:10.1.18.2 to 224.0.0.5 Intf:ve 18  LS-Upd L:136 Auth:0
ID:10.213.213.213 Cnt:3
OSPF: recv from:10.1.14.1 to 224.0.0.5 Intf:e 2/1/14 LS-Ack L:84 Auth:0
ID:10.213.213.213
OSPF: recv from:10.1.18.2 to 224.0.0.5 Intf:ve 18  LS-Ack L:84 Auth:0
ID:10.213.213.213
OSPF: send to:224.0.0.5 Intf:ve 911  Hello  L:48 Auth:0 ID:10.10.10.77
DR:10.1.91.16 BDR:10.1.91.18
OSPF: send to:224.0.0.5 Intf:loopback 1 Hello  L:44 Auth:0 ID:10.10.10.77
DR:10.10.10.77 BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:loopback 2 Hello  L:44 Auth:0 ID:10.10.10.77
DR:10.10.62.10 BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:loopback 3 Hello  L:44 Auth:0 ID:10.10.10.77
DR:10.10.63.10 BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:loopback 4 Hello  L:44 Auth:0 ID:10.10.10.77
DR:10.10.64.10 BDR:0.0.0.0
OSPF: send to:224.0.0.5 Intf:e 2/1/21 LS-Ack L:84 Auth:0 ID:10.10.10.77
OSPF: recv from:10.1.35.15 to 224.0.0.5 Intf:ve 35  LS-Ack L:84 Auth:0
ID:41.41.41.41
OSPF: recv from:10.1.34.10 to 224.0.0.5 Intf:ve 34  LS-Ack L:84 Auth:0 ID:49.2.3.4
OSPF: recv from:10.1.251.18 to 224.0.0.5 Intf:e 2/1/21 LS-Upd L:100 Auth:0
ID:10.116.116.1 Cnt:2
OSPF: send to:224.0.0.5 Intf:ve 18  LS-Upd L:100 Auth:0 ID:10.10.10.77 Cnt:2
OSPF: send to:224.0.0.5 Intf:e 2/1/14 LS-Upd L:100 Auth:0 ID:10.10.10.77 Cnt:2
OSPF: send to:224.0.0.5 Intf:e 4/1/12 LS-Upd L:100 Auth:0 ID:10.10.10.77 Cnt:2
OSPF: send to:224.0.0.5 Intf:ve 34  LS-Upd L:100 Auth:0 ID:10.10.10.77 Cnt:2
OSPF: send to:224.0.0.5 Intf:ve 35  LS-Upd L:100 Auth:0 ID:10.10.10.77 Cnt:2
OSPF: recv from:10.1.251.6 to 224.0.0.5 Intf:ve 511  LS-Upd L:100 Auth:0
ID:10.116.116.1 Cnt:2
OSPF: send to:10.1.251.6 Intf:ve 511  LS-Ack L:64 Auth:0 ID:10.10.10.77
OSPF: recv from:10.1.91.18 to 224.0.0.5 Intf:ve 911  LS-Upd L:100 Auth:0
ID:10.116.116.1 Cnt:2
OSPF: send to:10.1.91.18 Intf:ve 911  LS-Ack L:64 Auth:0 ID:10.10.10.77
OSPF: recv from:10.1.14.1 to 224.0.0.5 Intf:e 2/1/14 LS-Upd L:100 Auth:0
ID:10.213.213.213 Cnt:2
OSPF: recv from:10.1.51.1 to 224.0.0.5 Intf:e 4/1/12 LS-Upd L:100 Auth:0
ID:10.213.213.213 Cnt:2
OSPF: recv from:10.1.18.2 to 224.0.0.5 Intf:ve 18  LS-Upd L:100 Auth:0
ID:10.213.213.213 Cnt:2
```

```
OSPF: recv from:10.1.251.6 to 224.0.0.5 Intf:ve 511  LS-Upd L:748 Auth:0
ID:10.116.116.1 Cnt:20
OSPF: send to:224.0.0.5 Intf:ve 18  LS-Upd L:748 Auth:0 ID:10.10.10.77 Cnt:20
OSPF: send to:224.0.0.5 Intf:e 2/1/14 LS-Upd L:748 Auth:0 ID:10.10.10.77 Cnt:20
OSPF: send to:224.0.0.5 Intf:e 4/1/12 LS-Upd L:748 Auth:0 ID:10.10.10.77 Cnt:20
OSPF: send to:224.0.0.5 Intf:ve 34  LS-Upd L:748 Auth:0 ID:10.10.10.77 Cnt:20
OSPF: send to:224.0.0.5 Intf:ve 35  LS-Upd L:748 Auth:0 ID:10.10.10.77 Cnt:20
```

debug ip ospf route

Syntax: [no] **debug ip ospf route**

Enable this command to receive information about OSPF graceful restart events, including restart phases, graceful Link-State Advertisement (LSA) transmit and receive activity, and syslog messages.

```
Brocade# debug ip ospf route 10.3.3.0
OSPF: debug ospf route 10.3.3.0
FCX648S-ADV Router(config-if-e1000-1/1/15)# debug ip ospf route  192.168.30.0
 OSPF: debug ospf route 192.168.30.0
FCX648S-ADV Router(config-if-e1000-1/1/15)#OSPF: invalidate whole table - entry
192.168.30.0, state 0, path type 3
OSPF: calc ext route 192.168.30.0
OSPF: delete route 192.168.30.0 from rtm 0x367e7270, not_in_main 0
OSPF: calc ext route 192.168.30.0
OSPF: calc ext route 192.168.30.0
OSPF: calc ext route 192.168.30.0
OSPF: calc ext route 192.168.30.0
OSPF: ext route, net = 192.168.30.0, mask = 10.255.255.0 advrtr = 10.3.1.1, fwd =
0.0.0.0
OSPF: ext route changed 192.168.30.0, state 0
OSPF: ext route new  192.168.30.0, state 2, path type 3
OSPF: add route 192.168.30.0 to rtm, next hop 115.1.1.2, type 3, state 3,
not_in_main 1
OSPF: add to ospf route table, to valid  192.168.30.0, state 3, path type 3
OSPF: Modify route 192.168.30.0, type 3, state 3, not_in_main 0, next hop 10.1.1.2
OSPF: modify/modify route 192.168.30.0 (fwd 367e7333), type 3, state 3,
not_in_main 0, nhp 381c51c0
OSPF: validate route, new->valid  192.168.30.0, state 3, path type 3
```

debug ip ospf retransmission

Syntax: [no] **debug ip ospf retransmission**

This command generates internal information about OSPF retransmission of LSAs.

```
Brocade# debug ip ospf retransmission
Brocade(config)# clear ip routeOSPF: examine each neighbor and add advertisement
ls-id 10.3.13.0 to the retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.13.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.1.2.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.1.2.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.3.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.3.0 to the
retransmission list if necessary
```

```
OSPF: examine each neighbor and add advertisement ls-id 10.1.2.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.1.2.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.24.2.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.24.2.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.3.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.3.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.13.0 to the
retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.3.13.0 to the
retransmission list if necessary

Brocade(config)# OSPF: examine each neighbor and add advertisement ls-id 10.24.2.0
to the retransmission list if necessary
OSPF: examine each neighbor and add advertisement ls-id 10.24.2.0 to the
retransmission list if necessary
```

debug ip ospf spf

Syntax: [no] **debug ip ospf spf**

This command generates information about OSPF Shortest Path First (SPF) activity including SPF runs and calculations. Command output resembles the following example.

```
Brocade# debug ip ospf spf
OSPF:  spf-short debugging is on
Disable neighbor
Brocade(config-ospf-router)# Debug: Jan  1 02:38:55 OSPF: Schedule SPF(8001), in
prog 0, ospf build_routing_table 0 phase 1
Debug: Jan  1 02:38:55 OSPF: schedule spf, init spf delay 0, next hold 0 (ticks)
Debug: Jan  1 02:38:55 OSPF: Add to spf pending list, current time 96788,
scheduled 96788, next run 96788
Debug: Jan  1 02:38:55 OSPF: Schedule SPF(7001), in prog 0, ospf
build_routing_table 0 phase 1
Debug: Jan  1 02:38:55 OSPF: Schedule SPF(6002), in prog 0, ospf
build_routing_table 0 phase 1
Debug: Jan  1 02:38:55 OSPF: timer: give semaphore, start spf phase 1, time 96789,
scheduled 96788, run time 96788
Debug: Jan  1 02:38:55 OSPF: begin intra SPF run, chunk-id 0/-1 just_become_abr 0,
is_abr 0
Debug: Jan  1 02:38:55 OSPF: invalidate whole routing table, recal_just_become_abr
0,  just_become_abr 0
Debug: Jan  1 02:38:55 OSPF: running SPF for area 0 area-18ce7068 nextarea 0
next-area-id -1
Debug: Jan  1 02:38:55 OSPF: completed SPF for all areas
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC PHASE_INTRA end at 96789, is_abr 0
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC PHASE_TRANSIT end at 96789
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC PHASE_TYPE5 end at 96789
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC PHASE_TYPE7 end at 96789
Debug: Jan  1 02:38:55 OSPF: summary phase, is_abr 0
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC PHASE_SUMMARY end at 96789
Debug: Jan  1 02:38:55 OSPF: translation phase, is_abr 0
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC PHASE_TRANSLATION end at 96789
Debug: Jan  1 02:38:55 OSPF: SPF_cleanup: current 96789, set next run time 96788,
current hold 0, next hold 0
```

```
Debug: Jan  1 02:38:55 OSPF: ROUTE CALC end at 96789, pending 0
```

debug ip ospf vrf

Syntax: [no] **debug ip ospf vrf**

This command generates debugging information about the OSPF Virtual Routing and Forwarding (VRF) instance.

```
Brocade# debug ip ospf vrf
OSPF(one): send hello on area 0 interface 10.3.4.1
```

RIP debug commands

The following debug commands display debugging information for the Routing Information Protocol (RIP).

debug rip database

Syntax: [no] **debug rip database**

This command displays RIP database events.

```
Brocade# debug rip database
Brocade(config-vif-11)# RIP(default-vrf):  (v2) process response packet
        header: type:RESPONSE PACKET, version:2
RIP(default-vrf):    refresh 10.1.2.0/24 metric 3 from 10.5.5.2 ve 11
RIP(default-vrf):          existing route metric 2 from 10.4.4.1 ve 12
RIP(default-vrf):    refresh 10.24.2.0/24 metric 4 from 10.5.5.2 ve 11
RIP(default-vrf):          existing route metric 4 from 10.5.5.2 ve 11
RIP(default-vrf):  (v2) process response packet
        header: type:RESPONSE PACKET, version:2
RIP(default-vrf):    refresh 10.24.2.0/24 metric 4 from 10.4.4.2 ve 12
RIP(default-vrf):          existing route metric 4 from 10.5.5.2 ve 11
```

debug rip events

Syntax: [no] **debug rip events**

This command displays RIP events.

```
Brocade# debug rip events
Brocade(config-vif-11)# RIPng: update timer expired
clear ipv routeRIPng: triggered update
RIPng: garbage prefix 2001:DB8::/64 timer 1, metric 0, tag 0
         from :: on interface NULL
RIPng: garbage prefix 2001:DB8::/64 timer 1, metric 0, tag 0
         from :: on interface NULL
RIPng: Adding local connected route 2001:DB8::1/64 on interface v11
RIPng: Adding local connected route 2001:DB8::1/64 on interface v12
RIPng: update timer expired
RIPng: Redistribute add route 2001:DB8::/64, type CONNECTED (1/0)
RIPng: Redistribute add route 2001:DB8::/64, type CONNECTED (1/0)
```

NTP debug commands

This section describes the debug commands that generate debugging information about the NTP configurations.

debug ip ntp

Syntax: [no] debug ip ntp [algorithms | association | broadcast | clockadjust | errors | packet | server]

- algorithms - Displays information about the NTP system algorithms.
- association - Displays information about the NTP server and peer association.
- broadcast - Displays information about the NTP broadcast server and client.
- clockadjust - Displays information about the NTP clock-adjust process.
- errors - Displays information about the NTP error events.
- packet - Displays information about the NTP input and output packets.
- server - Displays information about the NTP server.

debug ip ntp algorithms

Syntax: [no] debug ip ntp algorithms

This command displays information about the NTP system algorithms. Command output resembles the following example.

```
Brocade# debug ip ntp algorithms
NTP: ntp_peer_unfit: dist exceeded - root dist 16.00527999 of peer 10.25.96.13 has
exceeded max dist 1.50096000
NTP: ntp_peer_unfit: unreachable - peer 10.25.96.13 is not reachable [peer->reach
0]

Brocade(config-ntp)# NTP: ntp_clock_filter: Adding offset 0, delay 0, disp 16 to
filter[0] for peer 10.25.96.13
NTP: ntp_clock_filter: No acceptable samples available
NTP: ntp_clock_filter: Adding offset 0.01133625, delay 0.27379156, disp
0.00001936 to filter[1] for peer 10.25.96.13
NTP: ntp_clock_filter: mitigated sample stats: n 1 offset 0.01133625 del
0.27379156 dsp 7.93750968 jit 0.00001525
NTP: ntp_peer_unfit: dist exceeded - root dist 8.07442072 of peer 10.25.96.13 has
exceeded max dist 1.50096000
NTP: ntp_clock_select: number of final survivors 0 and leap vote 0
NTP: ntp_clock_select: No survivors found. sys_peer is set to NULL
NTP: ntp_clock_filter: Adding offset 0.02724471, delay 0.30800050, disp
0.00001988 to filter[2] for peer 10.25.96.13
NTP: ntp_clock_filter: mitigated sample stats: n 2 offset 0.02724471 del
0.30800050 dsp 3.93752228 jit 0.00001525
NTP: ntp_peer_unfit: dist exceeded - root dist 4.09153779 of peer 10.25.96.13 has
exceeded max dist 1.50096000
NTP: ntp_clock_update: at 430 sample 430 associd 2
NTP: ntp_rstclock: mu 67 new state 5 old state 5 offset -0.00001326
```

debug ip ntp association

Syntax: [no] debug ip ntp association

This command displays information about the NTP server and peer association. Command output resembles the following example.

```
Brocade# debug ip ntp association
NTP: peer_clear: peer 10.25.96.13 next 646 refid INIT
NTP: newpeer: 10.25.96.13  mode client vers 4 poll 6 10 key 00000000
 Flags  Flags -> iBURST

SYSLOG: <14>Mar 21 17:44:33 Murali NTP: client association is mobilized for
10.25.96.13.

SYSLOG: <14>Mar 21 17:44:33 Murali NTP: The system clock is not synchronized to
any time source.
SYSLOG: <14>Mar 21 17:44:40 Murali NTP: Stratum is changed to 2.

SYSLOG: <14>Mar 21 17:44:40 Murali NTP: System clock is synchronized to
10.25.96.13.
```

debug ip ntp broadcast

Syntax: [no] **debug ip ntp broadcast**

This command displays information about the NTP broadcast server and client. Command output resembles the following example.

```
Brocade# debug ip ntp broadcast
Oct 19 18:32:46 NTP: ntp_timer: interface mgmt1 is up, we may send broadcast
packet
Oct 19 18:32:49 NTP: Sending NTP broadcast packet to subnet 10.20.111.255 via port
mgmt1
Oct 19 18:33:56 NTP: Sending NTP broadcast packet to subnet 10.20.111.255 via port
mgmt1
```

debug ip ntp clockadjust

Syntax: [no] **debug ip ntp clockadjust**

This command displays information about the NTP clock-adjust process. Command output resembles the following example.

```
Brocade# debug ip ntp clockadjust
NTP: ntp_clock_update: at 327 sample 327 associd 3
NTP: ntp_local_clock: hufbuf - ptr 1 mindly 0.23329046 huffpuff correction
-0.00067095
NTP: ntp_local_clock: clk offset 0.00917431 clk jit 0 clk stab 0 sys_poll 6
NTP: ntp_set_freq: drift 0.00000010, old freq 50000000
NTP: ntp_set_freq: new freq 49999995
NTP: ntp_adj_host_clock: new offset 0.00917431, freq 49999995
NTP: Adjusting the clock. offset 0.00917431, calib used 251687
NTP: After adjusting the clock. offset 0.00817431, calib used 252687
NTP: Adjusting the clock. offset 0.00817431, calib used 252687
NTP: After adjusting the clock. offset 0.00717431, calib used 253687
NTP: Adjusting the clock. offset 0.00717431, calib used 253687
```

debug ip ntp errors

Syntax: [no] **debug ip ntp errors**

This command displays information about the NTP error events.

debug ip ntp packet

Syntax: [no] **debug ip ntp packet**

This command displays information about the NTP input and output packets. Command output resembles the following example.

```
Brocade# debug ip ntp packet
NTP: Sending the NTP client packet to 10.25.96.13 port 123 via port id INVALID
    Leap 3, Version 4, Mode client, Stratum 16, Poll 6,
    Precision 2**-16, Root delay 0, Root disp 167, Ref Id INIT,
    Ref time 0.0 (00:00:00.0 GMT+00 Mon Jan 01 0)
    Org 0.0 (00:00:00.0 GMT+00 Mon Jan 01 0)
    Rec 0.0 (00:00:00.0 GMT+00 Mon Jan 01 0)
    Xmt 230.745068249 (00:03:50.745068249 GMT+00 Mon Jan 01 0) pkt len = 48 key 0

NTP: Received NTP server packet from 10.25.96.13 on port 123 via port id mgmt1 at
00:03:50.1959316492 GMT+00 Mon Jan 01 0
    Leap 0, Version 4, Mode server, Stratum 1, Poll 6,
    Precision 2**-29, Root delay 0, Root disp 0, Ref Id 10.67.84.83,
    Ref time 3572876227.2200200252 (17:37:07.2200200252 GMT+00 Thu Mar 21 2013)
    Org 230.745068249 (00:03:50.745068249 GMT+00 Mon Jan 01 0)
    Rec 3572876241.37257170 (17:37:21.37257170 GMT+00 Thu Mar 21 2013)
    Xmt 3572876241.37302765 (17:37:21.37302765 GMT+00 Thu Mar 21 2013) pkt len =
48 key 0

NTP: Sending the NTP client packet to 10.25.96.13 port 123 via port id INVALID
    Leap 3, Version 4, Mode client, Stratum 16, Poll 6,
    Precision 2**-16, Root delay 0, Root disp 169, Ref Id INIT,
    Ref time 0.0 (00:00:00.0 GMT+00 Mon Jan 01 0)
    Org 3572876241.37302765 (17:37:21.37302765 GMT+00 Thu Mar 21 2013)
    Rec 230.1959316492 (00:03:50.1959316492 GMT+00 Mon Jan 01 0)
    Xmt 232.784360585 (00:03:52.784360585 GMT+00 Mon Jan 01 0) pkt len = 48 key 0

NTP: Received NTP server packet from 10.25.96.13 on port 123 via port id mgmt1 at
00:03:52.1904063455 GMT+00 Mon Jan 01 0
    Leap 0, Version 4, Mode server, Stratum 1, Poll 6,
    Precision 2**-29, Root delay 0, Root disp 0, Ref Id 10.67.84.83,
    Ref time 3572876227.2200200252 (17:37:07.2200200252 GMT+00 Thu Mar 21 2013)
    Org 232.784360585 (00:03:52.784360585 GMT+00 Mon Jan 01 0)
    Rec 3572876243.75963892 (17:37:23.75963892 GMT+00 Thu Mar 21 2013)
    Xmt 3572876243.76010686 (17:37:23.76010686 GMT+00 Thu Mar 21 2013) pkt len =
48 key 0
```

debug ip ntp server

Syntax: [no] **debug ip ntp server**

This command displays information about the NTP server. Command output resembles the following example.

```
Brocade# debug ip ntp server
NTP: Received NTP client packet from 172.26.67.52 on port 123 via port id mgmt1 at
18:02:44.1139927355 GMT+00 Thu Mar 21 2013
    Leap 0, Version 4, Mode client, Stratum 3, Poll 6,
    Precision 2**-16, Root delay 16102, Root disp 90471, Ref Id 172.26.67.65,
    Ref time 3572877762.1183164018 (18:02:42.1183164018 GMT+00 Thu Mar 21 2013)
    Org 3572877762.1141785444 (18:02:42.1141785444 GMT+00 Thu Mar 21 2013)
    Rec 3572877762.1183164018 (18:02:42.1183164018 GMT+00 Thu Mar 21 2013)
    Xmt 3572877764.1137502159 (18:02:44.1137502159 GMT+00 Thu Mar 21 2013) pkt len
= 48 key 0
```

```
Brocade(config-ntp)# NTP: poll_update: for peer 10.250.229.100 hpoll 6 burst 0
retry 0 throttle 62 next poll 64
NTP: Received NTP server packet from 10.250.229.100 on port 123 via port id mgmt1
at 18:03:09.1891314446 GMT+00 Thu Mar 21 2013
    Leap 0, Version 4, Mode server, Stratum 1, Poll 6,
    Precision 2**-29, Root delay 0, Root disp 0, Ref Id 10.67.84.83,
    Ref time 3572877762.155297680 (18:02:42.155297680 GMT+00 Thu Mar 21 2013)
    Org 3572877789.878229501 (18:03:09.878229501 GMT+00 Thu Mar 21 2013)
    Rec 3572877789.1401109720 (18:03:09.1401109720 GMT+00 Thu Mar 21 2013)
    Xmt 3572877789.1401162509 (18:03:09.1401162509 GMT+00 Thu Mar 21 2013) pkt len
= 48 key 0

NTP: poll_update: for peer 10.250.229.100 hpoll 6 burst 0 retry 0 throttle 62 next
poll 67
```

Source Guard debug commands

The following debug command display information about the Source Guard transactions.

debug ip source guard

Syntax: [no] **debug ip source guard**

This command activates the IP source guard debugging.

```
Brocade# debug ip source guard
SOURCE GUARD:  debugging is on
Brocade(config-vif-11)# IPSrcSec: Add ip addr 10.3.3.12 on port 5/1/2 vlan 14
```

SSH debug commands

The following **debug ip ssh** command displays information about the Secure Shell (SSH) transactions.

debug ip ssh

Syntax: [no] **debug ip ssh**

This command activates the SSH debugging.

```
Brocade# debug ip ssh
       SSH:  debugging is on

Brocade(config)#SSH: Incoming connection request received
SSH: ssh_get_free_session_id: ssh.client[0].in_use is 0
SSH: Client session (0) established
SSH: Outgoing connection is ready

ShtcpConnectionStatus[0]: connection established

SSH:ShtcpSend[0]: eSendComplete: the string length [24] !

ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [47]
```

```
SSH:ShtcpSend[0]: eSendComplete: the string length [216] !

ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [464]
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [144]

SSH:ShtcpSend[0]: eSendComplete: the string length [640] !
SSH: Outgoing connection is ready
SSH: Data is ready to receive
SSH: Data is ready to receive
SSH: Remote closed connection
SSH: ssh_close_session#0, No. of Clients#0.

ShtcpReceiveStatus[0]: Closed
SSH: ShListen failed.
ShtcpOpenPassive[0]: WaitingForConnection
SSH: Incoming connection request received
SSH: ssh_get_free_session_id: ssh.client[0].in_use is 0
SSH: Client session (0) established
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpConnectionStatus[0]: connection established

ShtcpReceiveStatus[0]: the string length [47]

SSH:ShtcpSend[0]: eSendComplete: the string length [24] !

ShtcpSendStatus[0]: eSendComplete

SSH:ShtcpSend[0]: eSendComplete: the string length [216] !

ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [464]
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [144]

SSH:ShtcpSend[0]: eSendComplete: the string length [640] !
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [16]

ShtcpSendStatus[0]: eSendComplete

SSH:ShtcpSend[0]: eSendComplete: the string length [16] !

ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [52]

SSH:ShtcpSend[0]: eSendComplete: the string length [104] !

ShtcpSendStatus[0]: eSendComplete
```

```
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [68]

SSH:ShtcpSend[0]: eSendComplete: the string length [68] !

ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [116]

SSH:ShtcpSend[0]: eSendComplete: the string length [36] !
ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [68]

SSH:ShtcpSend[0]: eSendComplete: the string length [52] !

ShtcpSendStatus[0]: eSendComplete
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [148]

SSH:ShtcpSend[0]: eSendComplete: the string length [36] !
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [68]

ShtcpSendStatus[0]: eSendComplete

SSH:ShtcpSend[0]: eSendComplete: the string length [36] !
SSH: Outgoing connection is ready
SSH: Data is ready to receive

ShtcpReceiveStatus[0]: the string length [52]

ShtcpSendStatus[0]: eSendComplete

SSH:send_scp_data_to_ssh_client:buf len: 18, Qlen:18, Qaddr:256cd7e0,  Qdepth#1
SSH: ShListen event for SSH session[0].

SSH:ShtcpSend[0]: eSendComplete: the string length [36] !

ShtcpSendStatus[0]: eSendComplete

SSH:ShtcpSend[0]: eSendComplete: the string length [68] !

ShtcpSendStatus[0]: eSendComplete

SSH:ssh_event_handler: Freeing tx_buf 18, QAddress: 256cd7e0
SSH: Outgoing connection is ready
SSH: Outgoing connection is ready

Brocade# no debug ip ssh
          SSH:  debugging is off
```

Synchronization debug commands

You can use the **debug ip sync** command to debug IP synchronization and faults in synchronization.

debug ip sync

Syntax: debug ip sync

This command displays debugging information about IP synchronization and faults in synchronization.

```
Brocade# debug ip sync
Brocade(config-vif-11)# ND6 add sync: sent ip:2001:DB8::10 mac:none state:INCOMP
port:ve 12  vlan:12 isR:0 to all other units
ND6 add sync: sent ip:2001:DB8::10 mac:0000.0062.9f2d state:REACH port:e 3/3/1
vlan:12 isR:0 to all other units
ND6 add sync: sent ip:2001:DB8::10 mac:0000.0062.9f2d state:STALE port:e 3/3/1
vlan:12 isR:0 to all other units
```

TCP debug commands

The following **debug ip tcp** commands display debugging information about the Transmission Control Protocol (TCP) transactions.

debug ip tcp

Syntax: [**no**] **debug ip tcp** [**x:x::x:x** | **A.B.C.D**]

- **x:x::x:x** - Specifies the IPv6 address.
- **A.B.C.D** - Specifies the IPv4 address.

This command displays TCP information for the specified IPv6 or IPv4 address. The following is the sample output from the **debug ip tcp x:x::x:x** command.

```
Brocade# debug ip tcp 2001:DB8::192:111:101:25
          TCP:  ipv6 address 2001:DB8::192:111:101:25 debugging is on
Debug: Feb 11 10:46:57 TCP: sent packet (len=41) 2001:DB8::192:111:101:111:8197 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:18c2,seqn:11211247,ackn:835929143
Debug: Feb 11 10:46:57 TCP: sent packet (len=41) 2001:DB8::192:111:101:111:8197 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:1,hlen:5,chksum:14ac,seqn:11211268,ackn:835929143
Debug: Feb 11 10:46:57 TCP: rcvd packet (len=20) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8197
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:3bfa,seqn:835929143,ackn:11211268
Debug: Feb 11 10:46:57 TCP: rcvd packet (len=20) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8197
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:3bf9,seqn:835929143,ackn:11211290
Debug: Feb 11 10:46:57 TCP: rcvd packet (len=20) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8197
     packet:
syn:0,ack:1,rst:0,fin:1,hlen:5,chksum:3bce,seqn:835929143,ackn:11211290
Debug: Feb 11 10:46:57 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8197 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:1bce,seqn:11211290,ackn:835929144
Debug: Feb 11 10:47:05 TCP: sent packet (len=24) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
```

```
     packet: syn:1,ack:0,rst:0,fin:0,hlen:6,chksum:26f,seqn:4002716695,ackn:0
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=24) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:1,ack:1,rst:0,fin:0,hlen:6,chksum:17a4,seqn:2607451983,ackn:4002716696
Debug: Feb 11 10:47:05 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:ef48,seqn:4002716696,ackn:2607451984
Debug: Feb 11 10:47:05 TCP: sent packet (len=75) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:17de,seqn:4002716696,ackn:2607451984
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=20) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:2f49,seqn:2607451984,ackn:4002716751
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=57) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:fc84,seqn:2607451984,ackn:4002716751
Debug: Feb 11 10:47:05 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:ef11,seqn:4002716751,ackn:2607452021
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=39) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:2af6,seqn:2607452021,ackn:4002716751
Debug: Feb 11 10:47:05 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:ef11,seqn:4002716751,ackn:2607452040
Debug: Feb 11 10:47:05 TCP: sent packet (len=39) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:eabe,seqn:4002716751,ackn:2607452040
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=20) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:2f11,seqn:2607452040,ackn:4002716770
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=108) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:aa71,seqn:2607452040,ackn:4002716770
Debug: Feb 11 10:47:05 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:eec6,seqn:4002716770,ackn:2607452128
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=39) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:2a8b,seqn:2607452128,ackn:4002716770
Debug: Feb 11 10:47:05 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:eec6,seqn:4002716770,ackn:2607452147
Debug: Feb 11 10:47:05 TCP: rcvd packet (len=108) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
```

```
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:80c6,seqn:2607452147,ackn:4002716770
Debug: Feb 11 10:47:05 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:eec6,seqn:4002716770,ackn:2607452235
Debug: Feb 11 10:47:06 TCP: rcvd packet (len=108) 2001:DB8::192:111:101:25:179 ->
2001:DB8::192:111:101:111:8178
     packet:
syn:0,ack:1,rst:0,fin:0,hlen:5,chksum:cd4a,seqn:2607452235,ackn:4002716770
Debug: Feb 11 10:47:06 TCP: sent packet (len=20) 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:17
```

debug ip tcp driver

Syntax: [no] **debug ip tcp driver**

This command activates the TCP driver events debugging.

```
Brocade# debug ip tcp driver
TCP:  driver debugging is on
```

debug ip tcp memory

Syntax: [no] **debug ip tcp memory**

This command activates the TCP memory debugging.

```
Brocade# debug ip tcp memory
TCP:  memory debugging is on
```

debug ip tcp packet

Syntax: [no] **debug ip tcp packet**

This command activates the TCP packets debugging.

```
Brocade# debug ip tcp packet
TCP:  packet debugging is on
```

debug ip tcp sack

Syntax: [no] **debug ip tcp sack**

This command activates the TCP Selective Acknowledgment (SACK) debugging.

```
Brocade# debug ip tcp sack
TCP:  sack debugging is on
```

debug ip tcp transactions

Syntax: [no] **debug ip tcp transactions**

This command activates the TCP transactions debugging.

```
Brocade# debug ip tcp transactions
TCP:  transactions debugging is on
TCP:  transactions debugging is on
Debug: Feb 11 10:47:39 TCP: 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179: state change ESTABLISHED -> FIN-WAIT-1
Debug: Feb 11 10:47:39 TCP: sending FIN to 2001:DB8::192:111:101:25 port 179
```

```
Debug: Feb 11 10:47:39 TCP: FIN to 2001:DB8::192:111:101:111 port 8178 acked
Debug: Feb 11 10:47:39 TCP: 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179: state change FIN-WAIT-1 -> FIN-WAIT-2
Debug: Feb 11 10:47:39 TCP: 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179: state change FIN-WAIT-2 -> TIME-WAIT
Debug: Feb 11 10:47:39 TCP: FIN processed
Debug: Feb 11 10:47:39 TCP: 2001:DB8::192:111:101:111:8178 ->
2001:DB8::192:111:101:25:179: state change TIME-WAIT -> CLOSED
Debug: Feb 11 10:47:39 TCP: TCB 14440934 destroyed
Debug: Feb 11 10:47:58 TCP: 2001:DB8::192:111:101:111:8064 ->
2001:DB8::192:111:101:25:179: state change CLOSED -> SYN-SENT
Debug: Feb 11 10:47:58 TCP: connected to 2001:DB8::192:111:101:111 port 8064
advertising MSS 1436
Debug: Feb 11 10:47:58 TCP: connection to 10.146.1.17 port 179 received MSS 1436
MSS is 1436
Debug: Feb 11 10:47:58 TCP: 2001:DB8::192:111:101:111:8064 ->
2001:DB8::192:111:101:25:179: state change SYN-SENT -> ESTABLISHED
```

UDP debug commands

You can use the **debug ip udp** command to display debugging information about the User Datagram Protocol (UDP) transactions.

debug ip udp

Syntax: [no] **debug ip udp**

This commands activates the UDP debugging.

```
Brocade# debug ip udp
UDP:  debugging is on
```

VRRP and VRRP-E debug commands

Use the following commands to filter the Virtual Router Redundancy Protocol (VRRP) and Virtual Router Redundancy Protocol - Extended (VRRP-E) messages for IPv4 or IPv6, and to debug specific IPv4 or IPv6 packets within the router.

debug ip vrrp packet

Syntax: [no] **debug ip vrrp packet**

This command displays debugging information of any VRRP or VRRP-E packets within the device.

```
Brocade# debug ip vrrp packet
VRRP: packet debugging is on
VRRP (IPv6): send advertise! ver:3 type:1 vrid:100 pri:255 num of ip:1 adv:100
chk:44853
Num of ip addr 1
2000::7:1
VRRP (IPv4): rcvd packet! ver:2 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:1
chk:52198
Num of ip addr 1 10.10.10.2 from sender 10.10.10.2
VRRP (IPv6): rcvd packet! ver:3 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:100
chk:53518
```

```
Num of ip addr 1
fe80::7:2 from sender 2000::7:2
VRRP (IPv6): rcvd packet! ver:3 type:1 vrid:11 pri:255 #ip:1 aut:0 adv:200
chk:53417
Num of ip addr 1
fe80::8:2 from sender ::
```

debug ip vrrp packet v4

Syntax: [no] **debug ip vrrp packet v4**

This command displays debugging information of VRRP or VRRP-E packets specific to IPv4 within the device.

```
Brocade# debug ip vrrp packet v4
VRRP (IPv4): packet debugging is on
VRRP (IPv4): rcvd packet! ver:2 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:1 chk:52198

Num of ip addr 1
10.10.10.2 from sender 10.10.10.2
VRRP (IPv4): rcvd packet! ver:2 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:1 chk:52198

Num of ip addr 1
10.10.10.2 from sender 10.10.10.2
VRRP (IPv4): rcvd packet! ver:2 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:1 chk:52198

Num of ip addr 1
10.10.10.2 from sender 10.10.10.2
```

debug ip vrrp packet v6

Syntax: [no] **debug ip vrrp packet v6**

This command displays debugging information of VRRP or VRRP-E packets specific to IPv6 within the device.

```
Brocade# debug ip vrrp packet v6
VRRP (IPv6): packet debugging is on
VRRP (IPv6): rcvd packet! ver:3 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:100
chk:53518
Num of ip addr 1
fe80::7:2 from sender 2000::7:2
VRRP (IPv6): rcvd packet! ver:3 type:1 vrid:11 pri:255 #ip:1 aut:0 adv:200
chk:53417
Num of ip addr 1
fe80::8:2 from sender ::
VRRP (IPv6): send advertise! ver:3 type:1 vrid:100 pri:255 num of ip:1 adv:100
chk:44853
Num of ip addr 1
2000::7:1
VRRP (IPv6): rcvd packet! ver:3 type:1 vrid:10 pri:255 #ip:1 aut:0 adv:100
chk:53518
Num of ip addr 1
fe80::7:2 from sender 2000::7:2
10.10.10.2 from sender 10.10.10.2
```

debug ip vrrp events

Syntax: [no] **debug ip vrrp events**

This command displays debugging information of VRRP events only within the device.

```
Brocade# debug ip vrrp events
VRRP: events debugging is on
[44fd]VRRP (IPv4): 10.10.10.1 transit to master! IP addr 10.10.10.2 vrid 10, pri
100
[44fd]VRRP (IPv6): fe80::7:1 transit to master! IP addr fe80::7:1 vrid 10, pri 100
[4511]VRRP (IPv6): fe80::8:1 transit to master! IP addr fe80::8:1 vrid 11, pri 255
```

debug ip vrrp vrid

Syntax: [no] **debug ip vrrp vrid** *decimal*

This command filters VRRP or VRRP-E debugging using a virtual router identifier (VRID). The *decimal* variable refers to the VRID.

```
Brocade# debug ip vrrp vrid 100
Debug VRID: 100 for both IPv4 and IPv6 instances
```

Web debug commands

The following **debug ip web** commands display information about the web transactions.

debug ip web

Syntax: [no] **debug ip web**

This command activates the web debugging.

```
Brocade# debug ip web
WEB:  debugging is on
```

debug ip web-ssl

Syntax: [no] **debug ip web-ssl**

This commands activates web Secured Socket Layer (SSL) debugging.

```
Brocade# debug ip web-ssl
WEB SSL:  debugging is on
```

Chapter

IPv6 Diagnostics

5

This chapter contains the following sections:

General IPv6 debug commands

The **debug ipv6** command enables the collection of information about IPv6 configurations for troubleshooting.

debug ipv6

Syntax: debug ipv6 *address cache icmp mld nd packet ra*

- *address*—The IPv6 address.
- *cache*—The IPv6 cache entry.
- *icmp*—The Internet Control Message Protocol version 6 (ICMPv6) address.
- *mld*—The Multicast Listener Discovery (MLD) protocol activity.
- *nd*—The neighbor discovery.
- *packet*—The IPv6 packet.
- *ra*—The router address.

IPv6 MLD debug commands

The following debug commands enable the functions related to Multicast Listening Discovery (MLD) debugging.

debug ipv6 mld

Syntax: [no] **debug ipv6 mld**

This command displays the debugging information about the received and sent packets of the MLD.

```
Brocade# debug ipv6 mld
Brocade# MLD: rcvd Report-V1(ty=131)  g=ff03::26:2641 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2642 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2643 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2644 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2645 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2646 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2647 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2648 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2649 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=ff03::26:264a resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy 1/3/8), mld_size=24
```

debug ipv6 mld add-del-oif

Syntax: [no] **debug ipv6 mld add-del-oif**

This command is enabled with the MLD **debug ipv6 mcache-source** or **debug ipv6 mcache-group** commands at the same time.

```
Brocade# debug ipv6 mld show
debug ip6 mld mcache-source 50F5 is enabled
debug ip6 mld add-del-oif is enabled
Brocade# show ipv6 mld mcache
Example: (S G) cnt=: (S G) are the lowest 32 bits, cnt: SW proc. count
    OIF: 1/1/22 TR(1/1/32,1/1/33), TR is trunk, 1/1/32 primary, 1/1/33 output
vlan 400, 0 cache
vlan 601, 0 cache
vlan 602, 0 cache
vlan 701, 0 cache
vlan 888, 0 cache
vlan 1000, 5 caches. use 1 VIDX
1    (* 24:2402) cnt=85
     OIF: tag TR(3/1/12) 7/1/17
     age=0m up-time=24m, change=24m vidx=4130 (ref-cnt=0)
2    (* 24:2403) cnt=87
     OIF: tag TR(3/1/12) 7/1/17
     age=0m up-time=24m, change=24m vidx=4130 (ref-cnt=0)
3    (* 24:2404) cnt=249
     OIF: tag TR(3/1/12) 7/1/17
     age=0m up-time=24m, change=24m vidx=4130 (ref-cnt=0)
4    (* 24:2400) cnt=88
     OIF: tag TR(3/1/12) 7/1/17
     age=0m up-time=24m, change=24m vidx=4130 (ref-cnt=0)
5    (* 24:2401) cnt=254
     OIF: tag TR(3/1/12) 7/1/17
     age=0m up-time=24m, change=24m vidx=4130 (ref-cnt=0)
Brocade# Debug: Sep 19 17:39:49 Del 7/1/17  from (0x0 0x242404) vlan 1000
Debug: Sep 19 17:39:49 Del 7/1/17  from (0x0 0x242403) vlan 1000
Debug: Sep 19 17:39:49 Del 7/1/17  from (0x0 0x242402) vlan 1000
Debug: Sep 19 17:39:49 Del 7/1/17  from (0x0 0x242401) vlan 1000
```

```
Debug: Sep 19 17:39:49 Del 7/1/17  from (0x0 0x242400) vlan 1000

Brocade# show ipv6 mld mcache
Example: (S G) cnt=: (S G) are the lowest 32 bits, cnt: SW proc. count
    OIF: 1/1/22 TR(1/1/32,1/1/33), TR is trunk, 1/1/32 primary, 1/1/33 output
vlan 400, 0 cache
vlan 601, 0 cache
vlan 602, 0 cache
vlan 701, 0 cache
vlan 888, 0 cache
vlan 1000, 5 caches. use 1 VIDX
1    (* 24:2402) cnt=85
     OIF: tag TR(3/1/12)
     age=1m up-time=25m, change=25m vidx=4131 (ref-cnt=0) HW-AGE
2    (* 24:2403) cnt=87
     OIF: tag TR(3/1/12)
     age=1m up-time=25m, change=25m vidx=4131 (ref-cnt=0) HW-AGE
3    (* 24:2404) cnt=249
     OIF: tag TR(3/1/12)
     age=1m up-time=25m, change=25m vidx=4131 (ref-cnt=0) HW-AGE
4    (* 24:2400) cnt=88
     OIF: tag TR(3/1/12)
     age=1m up-time=25m, change=25m vidx=4131 (ref-cnt=0) HW-AGE
5    (* 24:2401) cnt=254
     OIF: tag TR(3/1/12)
     age=1m up-time=25m, change=25m vidx=4131 (ref-cnt=0) HW-AGE
Brocade# Debug: Sep 19 17:40:45 Add 7/1/17  to (0x0 0x242400) vlan 1000
Debug: Sep 19 17:40:45 Add 7/1/17  to (0x0 0x242401) vlan 1000
Debug: Sep 19 17:40:45 Add 7/1/17  to (0x0 0x242402) vlan 1000
Debug: Sep 19 17:40:45 Add 7/1/17  to (0x0 0x242403) vlan 1000
Debug: Sep 19 17:40:45 Add 7/1/17  to (0x0 0x242404) vlan 1000

Brocade# show ipv6 mld mc
Example: (S G) cnt=: (S G) are the lowest 32 bits, cnt: SW proc. count
    OIF: 1/1/22 TR(1/1/32,1/1/33), TR is trunk, 1/1/32 primary, 1/1/33 output
vlan 400, 0 cache
vlan 601, 0 cache
vlan 602, 0 cache
vlan 701, 0 cache
vlan 888, 0 cache
vlan 1000, 5 caches. use 1 VIDX
1    (* 24:2402) cnt=88
     OIF: 7/1/17 tag TR(3/1/12)
     age=0m up-time=25m, change=25m vidx=4130 (ref-cnt=0)
2    (* 24:2403) cnt=88
     OIF: 7/1/17 tag TR(3/1/12)
     age=0m up-time=25m, change=25m vidx=4130 (ref-cnt=0)
3    (* 24:2404) cnt=416
     OIF: 7/1/17 tag TR(3/1/12)
     age=0m up-time=25m, change=25m vidx=4130 (ref-cnt=0)
4    (* 24:2400) cnt=90
     OIF: 7/1/17 tag TR(3/1/12)
     age=0m up-time=25m, change=25m vidx=4130 (ref-cnt=0)
5    (* 24:2401) cnt=421
     OIF: 7/1/17 tag TR(3/1/12)
     age=0m up-time=25m, change=25m vidx=4130 (ref-cnt=0)
```

debug ipv6 mld add-del-oif all

Syntax: [no] **debug ipv6 mld add-del-oif all**

This command monitors outgoing interfaces (OIFs) that are added or deleted for all the MLD groups.

```
Brocade# debug ipv6 mld add-del-oif all
Brocade# debug ipv6 mld show
debug ip6 mld mcache-source 681D is enabled
debug ip6 mld add-del-oif all is enabled

Brocade# Debug: Sep 16 00:06:37 Del 7/1/17  from (0x0 0x252504) vlan 1000
Debug: Sep 16 00:06:37 Del 7/1/17  from (0x0 0x252503) vlan 1000
Debug: Sep 16 00:06:37 Del 7/1/17  from (0x0 0x252502) vlan 1000
Debug: Sep 16 00:06:37 Del 7/1/17  from (0x0 0x252501) vlan 1000
Debug: Sep 16 00:06:37 Del 7/1/17  from (0x0 0x252500) vlan 1000

Brocade# Debug: Sep 16 00:07:09 Add 7/1/17  to (0x0 0x252500) vlan 1000
Debug: Sep 16 00:07:09 Add 7/1/17  to (0x0 0x252501) vlan 1000
Debug: Sep 16 00:07:09 Add 7/1/17  to (0x0 0x252502) vlan 1000
Debug: Sep 16 00:07:09 Add 7/1/17  to (0x0 0x252503) vlan 1000
Debug: Sep 16 00:07:09 Add 7/1/17  to (0x0 0x252504) vlan 1000
```

debug ipv6 mld add-del-oif stack

Syntax: [no] **debug ipv6 mld add-del-oif stack**

This command displays the MLD stack trace.

```
Brocade# debug ipv6 mld add-del-oif stack
Brocade# debug ipv6 mld show
debug ip6 mld mcache-source 681D is enabled
debug ip6 mld add-del-oif stack is enabled

Brocade# Debug: Sep 16 00:33:08 MLD Snoop: Create (0x0 0x252502) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 2064096C 206414A8 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 Add 7/1/17  to (0x0 0x252502) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20640EB4 20642100 208C0D68
2064217C 20641554 20732A14 202884C4 204C1EEC 204C2288 204C235C 204C2420 20569EF0
205F0474 2056E85C 205F23E8 5010 15B58 1AAF4
Debug: Sep 16 00:33:08 Add 3/1/12  to (0x0 0x252502) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20641588 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 MLD Snoop: Create (0x0 0x252503) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 2064096C 206414A8 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 Add 7/1/17  to (0x0 0x252503) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20640EB4 20642100 208C0D68
2064217C 20641554 20732A14 202884C4 204C1EEC 204C2288 204C235C 204C2420 20569EF0
205F0474 2056E85C 205F23E8 5010 15B58 1AAF4
Debug: Sep 16 00:33:08 Add 3/1/12  to (0x0 0x252503) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20641588 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 MLD Snoop: Create (0x0 0x252504) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 2064096C 206414A8 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 Add 7/1/17  to (0x0 0x252504) vlan 1000
```

```
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20640EB4 20642100 208C0D68
2064217C 20641554 20732A14 202884C4 204C1EEC 204C2288 204C235C 204C2420 20569EF0
205F0474 2056E85C 205F23E8 5010 15B58 1AAF4
Debug: Sep 16 00:33:08 Add 3/1/12  to (0x0 0x252504) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20641588 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 MLD Snoop: Create (0x0 0x252500) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 2064096C 206414A8 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 Add 7/1/17  to (0x0 0x252500) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20640EB4 20642100 208C0D68
2064217C 20641554 20732A14 202884C4 204C1EEC 204C2288 204C235C 204C2420 20569EF0
205F0474 2056E85C 205F23E8 5010 15B58 1AAF4
Debug: Sep 16 00:33:08 Add 3/1/12  to (0x0 0x252500) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20641588 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 MLD Snoop: Create (0x0 0x252501) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 2064096C 206414A8 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
Debug: Sep 16 00:33:08 Add 7/1/17  to (0x0 0x252501) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20640EB4 20642100 208C0D68
2064217C 20641554 20732A14 202884C4 204C1EEC 204C2288 204C235C 204C2420 20569EF0
205F0474 2056E85C 205F23E8 5010 15B58 1AAF4
Debug: Sep 16 00:33:08 Add 3/1/12  to (0x0 0x252501) vlan 1000
Debug: Sep 16 00:33:08 stack: 20C3DA54 20640AB8 20641588 20732A14 202884C4
204C1EEC 204C2288 204C235C 204C2420 20569EF0 205F0474 2056E85C 205F23E8 5010 15B58
1AAF4
```

debug ipv6 mld clear

Syntax: [no] **debug ipv6 mld clear**

This command clears all the MLD debug settings.

```
Brocade# debug ipv6 mld clear
no debug ipv6 mld is enabled
```

debug ipv6 mld detail

Syntax: [no] **debug ipv6 mld detail**

This command displays the details of the MLD messages.

```
Brocade# debug ipv6 mld detail
Brocade# debug ipv6 mld show
debug ip6 mld  is enabled
debug ip6 mld detail is enabled
Brocade# Debug: Sep 19 16:25:34 MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2400
resp=0, pkt S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2400 to VL1000(ethe 7/1/21 ) rsp=10000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2401 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2401 to VL1000(ethe 7/1/22 ) rsp=10000 mld=24B, pkt=86B
```

```
Debug: Sep 19 16:25:34 MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2402 to VL1000(ethe 7/1/23 ) rsp=10000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2403 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2403 to VL1000(ethe 7/1/21 ) rsp=10000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2404 to VL1000(ethe 7/1/22 ) rsp=10000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2404 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2403 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2402 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2401 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:34 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2400 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:35 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2404 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:35 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2403 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:35 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2402 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:35 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2401 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
Debug: Sep 19 16:25:35 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0
2001:DB8::24:2400 to VL1000(ethe 7/1/21 ) rsp=1000 mld=24B, pkt=86B
```

debug ipv6 mld enable

Syntax: [no] **debug ipv6 mld enable**

This command enables debugging of MLD configurations.

```
Brocade# debug ipv6 mld enable
debug ipv6 mld is enabled
```

debug ipv6 mld error

Syntax: [no] **debug ipv6 mld error**

This command displays error information related to MLD.

```
Brocade# debug ipv6 mld error
ERR: Dropped because group address not valid 2001:DB8::1:ff00:2. .
MLD.VRF0.ERR: Rx packet is invalid. Dropping packet
ERR: Dropped because group address not valid 2001:DB8::1:ff00:2. .
MLD.VRF0.ERR: Rx packet is invalid. Dropping packet
error 0x1 0x0 : failed to add prefix 2001:DB8::/64 to hw
ERR: Dropped because group address not valid 2001:DB8::1:ff00:2. .
MLD.VRF0.ERR: Rx packet is invalid. Dropping packet
ERR: Dropped because group address not valid 2001:DB8::1:ff00:2. .
MLD.VRF0.ERR: Rx packet is invalid. Dropping packet
ERR: Dropped because group address not valid 2001:DB8::1:ff00:2. .
```

debug ipv6 mld group

Syntax: [no] **debug ipv6 mld group** *ipv6addr*

This command debugs the MLD group matching. The *ipv6addr* variable specifies the address of the IPv6 route.

```
Brocade# debug ipv6 mld group ff03::26:2641
Brocade# debug ipv6 mld show
debug ip6 mld group ff03::26:2641 is enabled
Brocade# MLD send Query(t=130) V1, s=fe80::224:38ff:fe26:d900 0000::0000 to
VL888(all) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Report-V1(ty=131)  g=ff03::26:2641 resp=0, pkt
S=fe80::200:3aff:fe01:3a86, on VL611 (phy e6/48), mld_size=24
MLD: rcvd Leave(ty=132)  g=ff03::26:2641 resp=0, pkt S=fe80::200:3aff:fe01:3a86,
on VL611 (phy e6/48), mld_size=24
MLD: rcvd Query(ty=130) V1 g=ff03::26:2641 resp=10000, pkt
S=fe80::21b:f3ff:fe84:124, on VL611 (phy e8/1), mld_size=24
MLD: rcvd Query(ty=130) V1 g=ff03::26:2641 resp=1000, pkt
S=fe80::21b:f3ff:fe84:124, on VL611 (phy e8/1), mld_size=24
MLD: rcvd Query(ty=130) V1 g=ff03::26:2641 resp=1000, pkt
S=fe80::21b:f3ff:fe84:124, on VL611 (phy e8/1), mld_size=24
MLD del VL611 (e6/48) from group entry ff03::26:2641
```

debug ipv6 mld level

Syntax: [no] **debug ipv6 mld level** *decimal*

This command displays the different levels of debugging output, and it must be enabled in combination with the other MLD debug commands. The *decimal* variable refers to the number of the MLD level from 1 through 3.

```
Brocade# debug ipv6 mld level 1
MLD Debug level = 1
Brocade# Debug: Sep 19 16:44:07 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2400
resp=0, pkt S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:44:07     no routr fid, consume pkt
Debug: Sep 19 16:44:07 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2401 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:44:07     no routr fid, consume pkt
Debug: Sep 19 16:44:07 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:44:07     no routr fid, consume pkt
Debug: Sep 19 16:44:07 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2403 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:44:07     no routr fid, consume pkt
Debug: Sep 19 16:44:07 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:44:07     no routr fid, consume pkt

Brocade# Debug: Sep 19 16:45:03 MLD send Query(t=130) V1,
s=fe80::224:38ff:fec6:d0c0 :: to VL1000(all) rsp=10000 mld=24B, pkt=86B
Debug: Sep 19 16:45:06 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff84:55ae resp=0,
pkt S=fe80::200:bff:fe84:55ae, on VL1000 (phy 2/1/1), mld_size=24
Debug: Sep 19 16:45:06     no routr fid, consume pkt
Debug: Sep 19 16:45:07 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff82:50f5 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:45:07     no routr fid, consume pkt
Debug: Sep 19 16:45:08 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff20:84 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
```

```
Debug: Sep 19 16:45:08     no routr fid, consume pkt

Brocade# Debug: Sep 19 16:45:10 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff20:70
resp=0, pkt S=fe80::200:bff:fe84:55ae, on VL1000 (phy 2/1/1), mld_size=24
Debug: Sep 19 16:45:10     no routr fid, consume pkt
Debug: Sep 19 16:45:10 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2400 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:45:10     no routr fid, consume pkt
Debug: Sep 19 16:45:10 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2401 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:45:10     no routr fid, consume pkt
Debug: Sep 19 16:45:10 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:45:10     no routr fid, consume pkt
Debug: Sep 19 16:45:10 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2403 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:45:10     no routr fid, consume pkt
Debug: Sep 19 16:45:10 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:45:10     no routr fid, consume pkt

Brocade# debug ipv6 mld level 3
MLD Debug level = 3
Brocade# Debug: Sep 19 16:45:58 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2400
resp=0, pkt S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:45:58     group: ff04::24:2400, life = 260
Debug: Sep 19 16:45:58     no routr fid, consume pkt
Debug: Sep 19 16:45:58 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2401 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:45:58     group: ff04::24:2401, life = 260
Debug: Sep 19 16:45:58     no routr fid, consume pkt
Debug: Sep 19 16:45:58 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:45:58     group: ff04::24:2402, life = 260
Debug: Sep 19 16:45:58     no routr fid, consume pkt
Debug: Sep 19 16:45:58 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2403 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:45:58     group: ff04::24:2403, life = 260
Debug: Sep 19 16:45:58     no routr fid, consume pkt
Debug: Sep 19 16:45:58 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:45:58     group: ff04::24:2404, life = 260
Debug: Sep 19 16:45:58     no routr fid, consume pkt
Debug: Sep 19 16:47:08 MLD send Query(t=130) V1, s=fe80::224:38ff:fec6:d0c0 :: to
VL1000(all) rsp=10000 mld=24B, pkt=86B
Debug: Sep 19 16:47:10 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff20:84 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:47:10     group: ff02::1:ff20:84, life = 260
Debug: Sep 19 16:47:10     no routr fid, consume pkt
Debug: Sep 19 16:47:12 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff20:70 resp=0, pkt
S=fe80::200:bff:fe84:55ae, on VL1000 (phy 2/1/1), mld_size=24
Debug: Sep 19 16:47:12     group: ff02::1:ff20:70, life = 260
Debug: Sep 19 16:47:12     no routr fid, consume pkt
Debug: Sep 19 16:47:15 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff84:55ae resp=0,
pkt S=fe80::200:bff:fe84:55ae, on VL1000 (phy 2/1/1), mld_size=24
Debug: Sep 19 16:47:15     group: ff02::1:ff84:55ae, life = 260
Debug: Sep 19 16:47:15     no routr fid, consume pkt
Debug: Sep 19 16:47:16 MLD: rcvd Report-V1(ty=131)  g=ff02::1:ff82:50f5 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:47:16     group: ff02::1:ff82:50f5, life = 260
```

```
Debug: Sep 19 16:47:16     no routr fid, consume pkt
Debug: Sep 19 16:47:18 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2400 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:47:18     group: ff04::24:2400, life = 260
Debug: Sep 19 16:47:18     no routr fid, consume pkt
Debug: Sep 19 16:47:18 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2401 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:47:18     group: ff04::24:2401, life = 260
Debug: Sep 19 16:47:18     no routr fid, consume pkt
Debug: Sep 19 16:47:18 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/21), mld_size=24
Debug: Sep 19 16:47:18     group: ff04::24:2402, life = 260
Debug: Sep 19 16:47:18     no routr fid, consume pkt
Debug: Sep 19 16:47:18 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2403 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/22), mld_size=24
Debug: Sep 19 16:47:18     group: ff04::24:2403, life = 260
Debug: Sep 19 16:47:18     no routr fid, consume pkt
Debug: Sep 19 16:47:18 MLD: rcvd Report-V1(ty=131)  g=ff04::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL1000 (phy 7/1/23), mld_size=24
Debug: Sep 19 16:47:18     group: ff04::24:2404, life = 260
Debug: Sep 19 16:47:18     no routr fid, consume pkt
```

debug ipv6 mld mcache-group

Syntax: [no] **debug ipv6 mld mcache-group** *hex*

This command displays a filter of only desired mcache group debugging output, and it must be enabled in combination with the other MLD debug commands. The <*hex*> variable refers to the hexadecimal value of the MLD mcache group.

```
Brocade# debug ipv6 mld  is enabled
debug ip6 mld mcache-group 2403 is enabled
debug ip6 mld level 3 is enabled

Brocade# Debug: Sep 28 00:09:51 MLD: rcvd Report-V1(ty=131)
g=2001:DB8::1:ff67:eb8b resp=0, pkt S=fe80::200:36ff:fe67:eb8b, on VL510 (phy
1/1/5), mld_size=24
Debug: Sep 28 00:09:51     group: 2001:DB8::1:ff67:eb8b, life = 260
Debug: Sep 28 00:09:51     forward to router fid 814
Debug: Sep 28 00:09:52 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::1:ff82:50f5
resp=0, pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:09:52     group: 2001:DB8::1:ff82:50f5, life = 260
Debug: Sep 28 00:09:52     forward to router fid 814
Debug: Sep 28 00:09:54 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::1:ff53:f9c5
resp=0, pkt S=fe80::200:36ff:fe53:f9c5, on VL510 (phy 1/1/1), mld_size=24
Debug: Sep 28 00:09:54     group: 2001:DB8::1:ff53:f9c5, life = 260
Debug: Sep 28 00:09:54     forward to router fid 814
Debug: Sep 28 00:09:56 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::1:ff20:73 resp=0,
pkt S=fe80::200:36ff:fe53:f9c5, on VL510 (phy 1/1/1), mld_size=24
Debug: Sep 28 00:09:56     group: 2001:DB8::1:ff20:73, life = 260
Debug: Sep 28 00:09:56     forward to router fid 814
Debug: Sep 28 00:09:56 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::1:ff20:77 resp=0,
pkt S=fe80::200:36ff:fe67:eb8b, on VL510 (phy 1/1/5), mld_size=24
Debug: Sep 28 00:09:56     group: 2001:DB8::1:ff20:77, life = 260
Debug: Sep 28 00:09:56     forward to router fid 814
Debug: Sep 28 00:09:56 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::1:ff20:84 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:09:56     group: 2001:DB8::1:ff20:84, life = 260
Debug: Sep 28 00:09:56     forward to router fid 814
```

```
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2400 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::24:2400, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2401 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::24:2401, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2402 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::24:2402, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2403 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::24:2403, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2404 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::24:2404, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2500 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::25:2500, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2501 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::25:2501, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2502 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::25:2502, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2503 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::25:2503, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
Debug: Sep 28 00:10:28 MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2504 resp=0,
pkt S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/3), mld_size=24
Debug: Sep 28 00:10:28     group: 2001:DB8::25:2504, life = 260
Debug: Sep 28 00:10:28     forward to router fid 814
```

debug ipv6 mld phy-port ethernet

Syntax: [no] **debug ipv6 mld phy-port ethernet** *stackid/slot/port*

This command matches the input of the physical port. The *stackid/slot/port* variable refers to the stack ID, slot number, and port number of a specific Ethernet port.

```
Brocade# debug ipv6 mld phy-port ethernet 1/1/22
Brocade# debug ipv6 mld show
debug ip6 mld physical_port 1/1/22 is enabled
Brocade# MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2400 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2501 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
```

```
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::25:2503 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
```

debug ipv6 mld prime-port ethernet

Syntax: [no] **debug ipv6 mld prime-port ethernet** *stackid/slot/port*

This command enables the activity of the physical port and primary port if trunking happened. The *stackid/slot/port* variable refers to the stack ID, slot number, and port number of a specific Ethernet port.

```
Brocade# debug ipv6 mld prime-port ethernet 1/1/21
Brocade# debug ipv6 mld show
debug ip6 mld prime-port 1/1/21 is enabled
Brocade# MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2400 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::24:2400 to
VL510(ethe 1/1/22 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2401 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/21), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::24:2401 to
VL510(ethe 1/1/21 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2402 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::24:2402 to
VL510(ethe 1/1/22 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2403 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/21), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::24:2403 to
VL510(ethe 1/1/21 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::24:2404 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::24:2404 to
VL510(ethe 1/1/22 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::25:2500 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/21), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::25:2500 to
VL510(ethe 1/1/21 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::25:2501 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::25:2501 to
VL510(ethe 1/1/22 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::25:2502 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/21), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::25:2502 to
VL510(ethe 1/1/21 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::25:2503 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/22), mld_size=24
MLD send Query(t=130) V1, s=fe80::21b:f3ff:fe84:124 2001:DB8::25:2503 to
VL510(ethe 1/1/22 ) rsp=10000 mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::25:2504 resp=0, pkt
S=fe80::200:bff:fe82:50f5, on VL510 (phy 1/1/21), mld_size=24
```

debug ipv6 mld protocol

Syntax: [no] **debug ipv6 mld protocol** [**query** | **report**]

- **query** - Displays debugging information about the IPv6 MLD queries transmitted and received.
- **report** - Displays debugging information about the IPv6 MLD reports transmitted and received.

The following is the sample output from the **debug ipv6 mld protocol query** command.

```
Brocade# debug ipv6 mld protocol query
Brocade# debug ipv6 mld enable
Brocade# MLD.VRF0: [ Port 6/1/17,v170 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe34:3c80
MLD.VRF0: [ Port 4/1/22,v78 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe34:3c80
MLD.VRF0: [ Port 6/1/22,v77 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe34:3c80
MLD.VRF0: [ Port 1/1/15,v74 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe34:3c80
```

The following is the truncated sample output from the **debug ipv6 mld protocol report** command.

```
Brocade# debug ipv6 mld protocol report
Brocade# debug ipv6 mld enable
Brocade# MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::14
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::2
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::6
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::22
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::11
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::2f
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::9
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::25
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::24
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::1c
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::2c
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::28
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::26
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::31
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::5
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::1f
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::18
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::2d
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::1
MLD.VRF0.RCV: Type V1 Report Port 3/1/12,v73 Grp ff84::2a
```

debug ipv6 mld rx

Syntax: [no] **debug ipv6 mld rx**

This command displays the IPv6 MLD packets received.

```
Brocade# debug ipv6 mld rx
MLD.VRF1.RCV: Received Query on Port 1/1/3*1/1/14,v25 Grp *

MLD.QRY: [ Port 1/1/3*1/1/14,v25. Grp :: Src fe80::207:11ff:fe11:1111].
Processing version 2 query

MLD.VRF3.RCV: Received Query on Port 1/1/3*1/1/14,v26 Grp *

MLD.QRY: [ Port 1/1/3*1/1/14,v26. Grp :: Src fe80::207:11ff:fe11:1111].
Processing version 2 query
```

debug ipv6 mld show

Syntax: debug ipv6 mld show

This command displays the IPv6 MLD debug settings. Command output resembles the following example.

```
Brocade# debug ipv6 mld show
```

```
debug ipv6 mld error
debug ipv6 mld vrf 3
```

debug ipv6 mld source

Syntax: [no] **debug ipv6 mld source** *ipaddr*

This command displays the MLD related debug information for the specified source address.

```
Brocade# debug ipv6 mld rx
Brocade# debug ipv6 mld protocol report
Brocade# debug ipv6 mld source 2001:DB8::15
=====================

MLD.VRF4.RCV: Type V2 Report Port 1/1/12,v107 Grp 2001:DB8:ff3c::7:0
MLD.RCV: Type V2 Rept Port 1/1/12,v107 #Grps 1, action BLK_OLD. #Srcs 1
MLD.VRF4.RCV: Type V2 Report Port 1/1/12,v107 Grp 2001:DB8:ff3c::7:0
MLD.RCV: Type V2 Rept Port 1/1/12,v107 #Grps 1, action BLK_OLD. #Srcs 1
MLD.VRF4.RCV: Type V2 Report Port 1/1/12,v107 Grp 2001:DB8:ff3c::7:0
MLD.RCV: Type V2 Rept Port 1/1/12,v107 #Grps 1, action ALW_NEW. #Srcs 1
MLD.VRF4.RCV: Type V2 Report Port 1/1/12,v107 Grp 2001:DB8:ff3c::7:0
MLD.RCV: Type V2 Rept Port 1/1/12,v107 #Grps 1, action ALW_NEW. #Srcs 1
MLD.VRF4.RCV: Type V2 Report Port 1/1/12,v107 Grp 2001:DB8:ff3c::7:0
```

debug ipv6 mld timer

Syntax: [no] **debug ipv6 mld timer**

This command monitors the MLD timer activity.

```
Brocade# debug ipv6 mld timer
Monitor MLD timers activity
```

debug ipv6 mld tx

Syntax: [no] **debug ipv6 mld tx**

This command displays the IPv6 MLD packets transmitted.

```
Brocade# debug ipv6 mld tx
Brocade# debug ipv6 mld protocol query

MLD.VRF1: [ Port 1/1/12,v105 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe0e:68c0

MLD.VRF3: [ Port 1/1/12,v103 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe0e:68c0

MLD.VRF4: [ Port 1/1/12,v107 ] Sent General Query version 2 using src
fe80::768e:f8ff:fe0e:68c0
```

debug ipv6 mld vlan

Syntax: [no] **debug ipv6 mld vlan** *decimal*

This command monitors the MLD VLAN activity. The *decimal* variable refers to the VLAN number.

```
Brocade# debug ipv6 mld vlan 2
monitor vlan 2
```

debug ipv6 mld vrf

Syntax: [no] **debug ipv6 mld vrf** *vrf_name*

This command displays MLD related debug information for the specified VRF.

```
Brocade# debug ipv6 mld vrf vrf1
MLD.VRF1.EVT: - Started FSM timer for 16 seconds
MLD.VRF1.EVT: Rx packet is valid. Processing packet
MLD.VRF1.EVT: - Started FSM timer for 16 seconds
MLD.VRF1: [ Port 3/1/11,3/1/11 ] General Query Timer expired. Sending Query
version 2
```

IPv6 OSPF debug commands

This section describes the debug commands used for monitoring the IPv6 OSPF environment.

debug ipv6 ospf ism

Syntax: [no] **debug ipv6 ospf ism**

This command displays debug information about the interface state machine (ISM) as shown in the following example.

```
Brocade# debug ipv6 ospf ism
OSPFv3:  ism debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:57:56 OSPFv3: ISM[644]:
InterfaceUp
Debug: Jan  1 19:57:56 OSPFv3: ISM[644]: Status change Down -> Waiting (Priority >
0)
Debug: Jan  1 19:57:56 OSPFv3: ISM[644]: Start Wait_Timer at 730122, 1458392

Disable neighbor

Debug: Jan  1 19:58:32 OSPFv3: ISM[644]: InterfaceDown
Debug: Jan  1 19:58:32 OSPFv3: ISM[644]: Status change Waiting -> Down
(Configured)
```

debug ipv6 ospf ism-events

Syntax: [no] **debug ipv6 ospf ism-events**

This command displays debug information about ISM events as shown in the following example.

```
Brocade# debug ipv6 ospf ism-events
OSPFv3:  ism-events debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 20:00:17 OSPFv3: ISM[644]:
InterfaceUp
Debug: Jan  1 20:00:17 OSPFv3: ISM[644]: Start Wait_Timer at 731552, 1461252
Debug: Jan  1 20:00:56 OSPFv3: ISM[644]:  WaitTimer at 731952, 1462051
Debug: Jan  1 20:00:56 OSPFv3: ISM[644]: NeighborChange
```

debug ipv6 ospf ism-status

Syntax: [no] **debug ipv6 ospf ism-status**

This command displays debug information about the status of ISM as shown in the following example.

```
Brocade# debug ipv6 ospf ism-status
OSPFv3:  ism-status debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 20:02:00 OSPFv3: ISM[644]: Status
change DR -> Down (Configured)
```

```
Debug: Jan  1 20:02:00 OSPFv3: ISM[644]: {dr:10.4.4.8,bdr:10.2.2.2} ->
{dr:0.0.0.0,bdr:0.0.0.0}
```

debug ipv6 ospf lsa

Syntax: [no] **debug ipv6 ospf lsa**

This command displays LSAs as shown in the following example.

```
Brocade# debug ipv6 ospf lsa
OSPFv3:  lsa debugging is on
Brocade(config-vif-9)#Debug: Jan  1 17:05:01 OSPFv3: LSA: Update Router-LSA for
area 0.0.0.0
Debug: Jan  1 17:05:01 OSPFv3: LSA: Create LSA  Type :Router Id: 0 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:01 OSPFv3: LSA: Type: 8193  ID: 0 AdvRouter: 10.4.4.8, Supress
Updating
Debug: Jan  1 17:05:01 OSPFv3: LSA: Delete LSA Type :Router Id: 0.0.0.0 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:01 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Checking
Interface 644
Debug: Jan  1 17:05:01 OSPFv3: LSA Update Intra-Area-Prefix(Stub): include
5001::/64
Debug: Jan  1 17:05:01 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:01 OSPFv3: LSA: Type: 8201  ID: 0 AdvRouter: 10.4.4.8, Supress
Updating
Debug: Jan  1 17:05:01 OSPFv3: LSA: Delete LSA Type :IntraPrefix Id: 0.0.0.0
Advrouter: 10.4.4.8
Debug: Jan  1 17:05:01 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Area 0.0.0.0
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA Header Type :Router Id: 0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: request Type =8193 ADvRtr =10.2.2.2
ID=0
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA Header Type :IntraPrefix Id: 0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: request Type =8201 ADvRtr =10.2.2.2
ID=0
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA Header Type :Link Id: 2052
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: request Type =8 ADvRtr =10.2.2.2
ID=2052
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :Router Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA HEADER Type :Router Id: 0.0.0.0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[418103392]: delayed ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA HEADER Type :IntraPrefix Id:
0.0.0.0 Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[418103392]: delayed ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :Link Id: 2052 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA HEADER Type :Link Id: 10.0.8.4
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit):Interface 644
is Stub
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Checking
Interface 644
```

```
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Stub): include
5001::/64
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:02 OSPFv3: LSA: Type: 8201  ID: 0 AdvRouter: 10.4.4.8, Supress
Updating
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :IntraPrefix Id: 0.0.0.0
Advrouter: 10.4.4.8
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Area 0.0.0.0
Debug: Jan  1 17:05:02 OSPFv3: LSA[418103392]: delayed ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :Router Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Turnover type:Router Lsa Id:0.0.0.0
AdvRouter:10.2.2.2: Debug: Jan  1 17:05:02 OSPFv3: contents changed
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :Router Id: 0.0.0.0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[418103392]: delayed ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :Router Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: direct ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :Router Id: 0.0.0.0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Turnover type:IntraPrefix Lsa Id:0.0.0.0
AdvRouter:10.2.2.2: Debug: Jan  1 17:05:02 OSPFv3: contents changed
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :IntraPrefix Id: 0.0.0.0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[418103392]: delayed ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: direct ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :IntraPrefix Id: 0.0.0.0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: direct ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :IntraPrefix Id: 0.0.0.0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 0 Advrouter:
10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA[10.2.2.2]: direct ack
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :IntraPrefix Id: 0.0.0.0
Advrouter: 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA: Update Router-LSA for area 0.0.0.0
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :Router Id: 0 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:02 OSPFv3: LSA: Turnover type:Router Lsa Id:0.0.0.0
AdvRouter:10.4.4.8: Debug: Jan  1 17:05:02 OSPFv3: contents changed
Debug: Jan  1 17:05:02 OSPFv3: LSA: Delete LSA Type :Router Id: 0.0.0.0 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:02 OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Network: Interface 644
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :Network Id: 644 Advrouter:
10.4.4.8
Debug: Jan  1 17:05:02 OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): Interface
644
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit):Checking Type
:Link Id 10.0.2.132: Adv Routr : 10.4.4.8
```

```
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): 1
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): Prefix
5001::
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): including
5001::
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit):Checking Type
:Link Id 10.0.8.4: Adv Routr : 10.2.2.2
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): 1
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): Prefix
5001::
Debug: Jan  1 17:05:02 OSPFv3: LSA Update Intra-Area-Prefix(Transit): 5001:: is
Duplicate
Debug: Jan  1 17:05:02 OSPFv3: LSA: Create LSA  Type :IntraPrefix Id: 19320
Advrouter: 10.4.4.8
Debug: Jan  1 17:05:02 OSPFv3: LSA: schedule flooding 10.2.2.2
```

debug ipv6 ospf lsa-flooding

Syntax: [no] **debug ipv6 ospf lsa-flooding**

This command displays debug information about LSA flooding activity as shown in the following example.

```
Brocade# debug ipv6 ospf lsa-flooding
OSPFv3:  lsa-flooding debugging is on
Brocade(config)# ipv6 router ospf
Brocade(config-ospf6-router)# redistribute connected
Brocade(config-ospf6-router)# redistribute connected Debug: Jan  1 17:17:49
OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:17:49 OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:17:49 OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:17:49 OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:17:49 OSPFv3: LSA: schedule flooding 10.2.2.2
Debug: Jan  1 17:17:49 OSPFv3: LSA: schedule flooding 10.2.2.2
```

debug ipv6 ospf lsa-generation

Syntax: [no] **debug ipv6 ospf lsa-generation**

This command displays debug information about LSA generation as shown in the following example.

```
Brocade# debug ipv6 ospf lsa-generation
OSPFv3:  lsa-generation debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type
:Router Id: 0 Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :Router Id: 0.0.0.0 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 1 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.1
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 2 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.2
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 3 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.3
Advrouter: 10.4.4.8
```

```
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 4 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.4
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 5 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.5
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :Router Id: 0 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :Router Id: 0.0.0.0 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 1 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.1
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 2 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.2
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 3 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.3
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 4 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.4
Advrouter: 10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Create LSA  Type :ASExternal Id: 5 Advrouter:
10.4.4.8
Debug: Jan  1 18:53:55 OSPFv3: LSA: Delete LSA Type :ASExternal Id: 10.0.0.5
Advrouter: 10.4.4.8
```

debug ipv6 ospf lsa-install

Syntax: [no] **debug ipv6 ospf lsa-install**

This command displays debug information about installed LSAs as shown in the following example.

```
Brocade# debug ipv6 ospf lsa-install
OSPFv3:  lsa-install debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 19:03:16 OSPFv3: LSA: Turnover
type:Router Lsa Id:0.0.0.0 AdvRouter:10.4.4.8: Debug: Jan  1 19:03:16 OSPFv3:
contents changed
Debug: Jan  1 19:03:59 OSPFv3: LSA: Turnover type:Router Lsa Id:0.0.0.0
AdvRouter:10.4.4.8: Debug: Jan  1 19:03:59 OSPFv3: contents changed
```

debug ipv6 ospf lsa-inter-area

Syntax: [no] **debug ipv6 ospf lsa-inter-area**

This command displays debug information about inter-area LSAs as shown in the following example.

```
Brocade# debug ipv6 ospf lsa-inter-area
OSPFv3:  lsa-inter-area debugging is on
Brocade(config-vif-7)# disable
Brocade(config-vif-7)# enable
Brocade(config-vif-7)# OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, path type = External
```

```
OSPFv3: Installing in LSDB Inter Area Prefix LSA for area 0.0.0.0 and prefix
2001:DB8::/64
OSPFv3: Inter Area LSA not generated, route is in same area.
OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, route is in same area.
OSPFv3: Installing in LSDB Inter Area Prefix LSA for area 10.0.0.1 and prefix
2001:DB8::/64
OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, path type = External
OSPFv3: Inter Area LSA not generated, route is in same area.
shOSPFv3: Inter Area LSA not generated, route is in same area.
```

debug ipv6 ospf lsa-refresh

Syntax: [no] debug ipv6 ospf lsa-refresh

This command displays LSA refresh information as shown in the following example.

```
Brocade# debug ipv6 ospf lsa-refresh
OSPFv3:  lsa-refresh debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 19:01:39 OSPFv3: LSA: Update
Router-LSA for area 0.0.0.0
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Checking
Interface 644
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Interface 644
is down
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): No prefix to
advertise for Area 0.0.0.0
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Area 0.0.0.0
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Network: Interface 644 is not DR
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Transit):Interface 644
is not DR
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Checking
Interface 644
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Interface 644
is down
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): No prefix to
advertise for Area 0.0.0.0
Debug: Jan  1 19:01:39 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Area 0.0.0.0
Debug: Jan  1 19:01:44 OSPFv3: LSA: Update Router-LSA for area 0.0.0.0
Debug: Jan  1 19:02:42 OSPFv3: LSA: Update Router-LSA for area 0.0.0.0
Debug: Jan  1 19:02:42 OSPFv3:  LSA: LSA Update AS-External: ID 10.0.0.6
Debug: Jan  1 19:02:42 OSPFv3:  LSA: LSA Update AS-External: ID 10.0.0.7debug ip
Debug: Jan  1 19:02:42 OSPFv3:  LSA: LSA Update AS-External: ID 10.0.0.8
Debug: Jan  1 19:02:42 OSPFv3:  LSA: LSA Update AS-External: ID 10.0.0.9
Debug: Jan  1 19:02:42 OSPFv3:  LSA: LSA Update AS-External: ID 10.0.0.10
Debug: Jan  1 19:02:42 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Checking
Interface 644
Debug: Jan  1 19:02:42 OSPFv3: LSA Update Intra-Area-Prefix(Stub): include
2001:DB8::/64
Debug: Jan  1 19:02:42 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Area 0.0.0.0
Debug: Jan  1 19:02:42 OSPFv3: LSA Update Link: Interface 644
Debug: Jan  1 19:02:46 OSPFv3: LSA Update Link: Interface 644
Debug: Jan  1 19:02:46 OSPFv3: LSA: Type: 8  ID: 644 AdvRouter: 10.4.4.8, Supress
Updating
Debug: Jan  1 19:02:46 OSPFv3: LSA Update Link: Interface 644
```

```
Debug: Jan  1 19:02:46 OSPFv3: LSA: Type: 8  ID: 644 AdvRouter: 10.4.4.8, Supress
Updating
Debug: Jan  1 19:02:47 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Checking
Interface 644
Debug: Jan  1 19:02:47 OSPFv3: LSA Update Intra-Area-Prefix(Stub): include
2001:DB8::/64
Debug: Jan  1 19:02:47 OSPFv3: LSA: Type: 8201  ID: 0 AdvRouter: 10.4.4.8, Supress
Updating
Debug: Jan  1 19:02:47 OSPFv3: LSA Update Intra-Area-Prefix(Stub): Area 0.0.0.0
Debug: Jan  1 19:02:47 OSPFv3: LSA: Update Router-LSA for area 0.0.0.0
Debug: Jan  1 19:02:47 OSPFv3: LSA: Type: 8193  ID: 0 AdvRouter: 10.4.4.8, Supress
Updating
```

debug ipv6 ospf nsm

Syntax: [no] **debug ipv6 ospf nsm**

This command displays debug information about the neighbor state machine (NSM) as shown in the following example.

```
Brocade# debug ipv6 ospf nsm
OSPFv3:  nsm debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 20:03:13 OSPFv3: NSM[644]: SendHello
at 733344
Debug: Jan  1 20:03:14 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733356,
1464858
Debug: Jan  1 20:03:14 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733356,
1464858
Debug: Jan  1 20:03:20 OSPFv3: NSM[644]: SendHello at 733413
Debug: Jan  1 20:03:20 OSPFv3: NSM[10.2.2.2]: AdjOK?
Debug: Jan  1 20:03:20 OSPFv3: NSM[10.2.2.2]: Status change
[2-way]->[ExStart](Need Adjacency)
Debug: Jan  1 20:03:20 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733418,
1464982
Debug: Jan  1 20:03:20 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733418,
1464982
Debug: Jan  1 20:03:25 OSPFv3: NSM[10.2.2.2]: NegotiationDone
Debug: Jan  1 20:03:25 OSPFv3: NSM[10.2.2.2]: Status change
[ExStart]->[ExChange](NegotiationDone)
Debug: Jan  1 20:03:25 OSPFv3: NSM[10.2.2.2]: ExchangeDone
Debug: Jan  1 20:03:25 OSPFv3: NSM[10.2.2.2]: Status change
[ExChange]->[Loading](Requestlist Not Empty)
Debug: Jan  1 20:03:25 OSPFv3: NSM[10.2.2.2]: LoadingDone
Debug: Jan  1 20:03:25 OSPFv3: NSM[10.2.2.2]: Status change
[Loading]->[Full](LoadingDone)
Debug: Jan  1 20:03:30 OSPFv3: NSM[644]: SendHello at 733513
Debug: Jan  1 20:03:30 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733520,
1465185
Debug: Jan  1 20:03:30 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733520,
1465185
Debug: Jan  1 20:03:40 OSPFv3: NSM[644]: SendHello at 733616
Debug: Jan  1 20:03:41 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733625,
1465396
Debug: Jan  1 20:03:41 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733625,
1465396
```

debug ipv6 ospf nsm-events

Syntax: [no] **debug ipv6 ospf nsm-events**

This command displays debug information about NSM events as shown in the following example.

```
Brocade# debug ipv6 ospf nsm-events
OSPFv3:  nsm-events debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 20:04:10 OSPFv3: NSM[10.2.2.2, 644]:
HelloReceived at 733918, 1465981
Debug: Jan  1 20:04:10 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 733918,
1465981
Debug: Jan  1 20:04:11 OSPFv3: NSM[644]: SendHello at 733932
Debug: Jan  1 20:04:19 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 734019,
1466183
Debug: Jan  1 20:04:19 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 734019,
1466183
Debug: Jan  1 20:04:20 OSPFv3: NSM[644]: SendHello at 734028
Debug: Jan  1 20:04:28 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 734109,
1466363
Debug: Jan  1 20:04:28 OSPFv3: NSM[10.2.2.2, 644]: HelloReceived at 734109,
1466363
Debug: Jan  1 20:04:31 OSPFv3: NSM[644]: SendHello at 734133
```

debug ipv6 ospf nsm-status

Syntax: [no] **debug ipv6 ospf nsm-status**

This command displays information about the status of NSM as shown in the following example.

```
Brocade# debug ipv6 ospf nsm-status
OSPFv3:  nsm-status debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 20:06:44 OSPFv3: NSM[10.2.2.2]:
Status change [Down]->[Init](HelloReceived)
Debug: Jan  1 20:06:44 OSPFv3: NSM[10.2.2.2]: Status change [Init]->[2-way](No
Need Adjacency)
Debug: Jan  1 20:07:19 OSPFv3: NSM[10.2.2.2]: Status change
[2-way]->[ExStart](Need Adjacency)
Debug: Jan  1 20:07:24 OSPFv3: NSM[10.2.2.2]: Status change
[ExStart]->[ExChange](NegotiationDone)
Debug: Jan  1 20:07:24 OSPFv3: NSM[10.2.2.2]: Status change
[ExChange]->[Loading](Requestlist Not Empty)
Debug: Jan  1 20:07:24 OSPFv3: NSM[10.2.2.2]: Status change
[Loading]->[Full](LoadingDone)
```

debug ipv6 ospf packet

Syntax: [no] **debug ipv6 ospf packet**

This command displays all OSPFv3 packets in rx or tx mode as shown in the following example.

```
Brocade# debug ipv6 ospf packet
OSPFv3:  packet debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 17:20:18 OSPFv3: Rcv Hello on ve 9
OSPFv3: (fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:18 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:23 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:29 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
```

```
Debug: Jan  1 17:20:29 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:32 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:38 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:38 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:43 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:49 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:49 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:54 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:58 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:20:58 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:21:04 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:21:09 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:21:09 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 17:21:14 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
```

debug ipv6 ospf packet-dd

Syntax: [no] **debug ipv6 ospf packet-dd**

This command displays all OSPFv3 data description packets in rx or tx mode as shown in the following example.

```
Brocade# debug ipv6 ospf packet-dd
OSPFv3:  packet-dd debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 19:06:18 OSPFv3: Rcv DbDesc on ve 9
OSPFv3: (fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
Debug: Jan  1 19:06:18   Option:00-00-13, Bits:07 SEQ:19b40
Debug: Jan  1 19:06:18 OSPFv3: DbDesc from 10.2.2.2 Ignored: state less than Init
Debug: Jan  1 19:06:19 OSPFv3: set dbdesc seqnum 000aa922 for 10.2.2.2
Debug: Jan  1 19:06:19 OSPFv3: Snd DbDesc on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
Debug: Jan  1 19:06:19   Option:00-00-13, Bits:07 SEQ:aa922
```

```
Debug: Jan  1 19:06:19 OSPFv3: Rcv DbDesc on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
Debug: Jan  1 19:06:19   Option:00-00-13, Bits:00 SEQ:aa922
Debug: Jan  1 19:06:19 OSPFv3: Snd DbDesc on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
Debug: Jan  1 19:06:19   Option:00-00-13, Bits:01 SEQ:aa923
Debug: Jan  1 19:06:19 OSPFv3: Rcv DbDesc on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
Debug: Jan  1 19:06:19   Option:00-00-13, Bits:00 SEQ:aa923
```

debug ipv6 ospf packet-hello

Syntax: [no] **debug ipv6 ospf packet-hello**

This command displays all OSPFv3 hello packets in rx or tx mode as shown in the following example.

```
Brocade# debug ipv6 ospf packet-hello
OSPFv3:  packet-hello debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 18:52:05 OSPFv3: Snd Hello on ve 9
OSPFv3: (fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 18:52:07 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 18:52:07 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 18:52:16 OSPFv3: Snd Hello on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  RtrID:10.4.4.8 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 18:52:16 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
Debug: Jan  1 18:52:16 OSPFv3: Rcv Hello on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  RtrID:10.2.2.2 DR:10.4.4.8 BDR:10.2.2.2
```

debug ipv6 ospf packet-lsa-ack

Syntax: [no] **debug ipv6 ospf packet-lsa-ack**

This command displays all OSPFv3 LSA acknowledgment packets in rx or tx mode as shown in the following example.

```
Brocade# debug ipv6 ospf packet-lsa-ack
OSPFv3:  packet-lsa-ack debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 20:14:33 OSPFv3: Rcv LSAck on ve 9
OSPFv3: (fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:8000000f AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:2001, LSID:0 Adv:10.2.2.2 SEQ:8000003c AGE:1
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
```

```
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Rcv LSAck on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
  Type:2009, LSID:0 Adv:10.4.4.8 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Rcv LSAck on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:4005, LSID:76 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:77 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:78 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:79 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:80 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:0008, LSID:644 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:2001, LSID:0 Adv:10.4.4.8 SEQ:8000003c AGE:1
  Type:2002, LSID:644 Adv:10.4.4.8 SEQ:80000010 AGE:1
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:80000010 AGE:1
  Type:2009, LSID:0 Adv:10.4.4.8 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4c 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4d 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4e 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4f 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 50 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 8 284 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 9 0 4040408 ACKed by Neighbor 10.2.2.2 is not in
retranslist
Debug: Jan  1 20:14:33 OSPFv3: Rcv LSAck on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:4005, LSID:76 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:77 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:78 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:79 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:4005, LSID:80 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:0008, LSID:644 Adv:10.4.4.8 SEQ:80000001 AGE:46
  Type:2001, LSID:0 Adv:10.4.4.8 SEQ:8000003c AGE:1
  Type:2002, LSID:644 Adv:10.4.4.8 SEQ:80000010 AGE:1
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:80000010 AGE:1
  Type:2009, LSID:0 Adv:10.4.4.8 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4c 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4d 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4e 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 4f 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 5 50 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 8 284 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 1 0 4040408 ACKed by Neighbor 10.2.2.2 is not in
retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 2 284 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
```

```
Debug: Jan  1 20:14:33 OSPFv3: LS 9 4b78 4040408 ACKed by Neighbor 10.2.2.2 is not
in retranslist
Debug: Jan  1 20:14:33 OSPFv3: LS 9 0 4040408 ACKed by Neighbor 10.2.2.2 is not in
retranslist
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:0008, LSID:2052 Adv:10.2.2.2 SEQ:80000001 AGE:45
  Type:4005, LSID:271 Adv:10.2.2.2 SEQ:80000001 AGE:45
  Type:4005, LSID:272 Adv:10.2.2.2 SEQ:80000001 AGE:45
  Type:4005, LSID:273 Adv:10.2.2.2 SEQ:80000001 AGE:45
  Type:4005, LSID:274 Adv:10.2.2.2 SEQ:80000001 AGE:45
  Type:4005, LSID:275 Adv:10.2.2.2 SEQ:80000001 AGE:45
  Type:4005, LSID:276 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:277 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:278 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:279 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:280 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:281 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:282 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:283 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:284 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:285 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:286 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:287 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:4005, LSID:288 Adv:10.2.2.2 SEQ:80000001 AGE:44
  Type:2001, LSID:0 Adv:10.2.2.2 SEQ:8000003c AGE:1
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:4005, LSID:272 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:4005, LSID:273 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 20:14:33 OSPFv3: Snd LSAck on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:4005, LSID:272 Adv:10.2.2.2 SEQ:80000001 AGE:3600
  Type:4005, LSID:273 Adv:10.2.2.2 SEQ:80000001 AGE:3600
```

debug ipv6 ospf packet-lsa-req

Syntax: [no] **debug ipv6 ospf packet-lsa-req**

This command displays all OSPFv3 LSA request packets in rx or tx mode as shown in the following example.

```
Brocade# debug ipv6 ospf packet-lsa-req
OSPFv3:  packet-lsa-req debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:12:00 OSPFv3: Rcv LSReq on ve 9
OSPFv3: (fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
  Type:2001, LSID:0 Adv-Router:10.4.4.8
  Type:2009, LSID:0 Adv-Router:10.4.4.8
  Type:4005, LSID:26 Adv-Router:10.4.4.8
  Type:4005, LSID:27 Adv-Router:10.4.4.8
  Type:4005, LSID:28 Adv-Router:10.4.4.8
  Type:4005, LSID:29 Adv-Router:10.4.4.8
  Type:4005, LSID:30 Adv-Router:10.4.4.8
  Type:0008, LSID:644 Adv-Router:10.4.4.8
Debug: Jan  1 19:12:00 OSPFv3: Snd LSReq on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:2009, LSID:19320 Adv-Router:10.4.4.8
```

```
  Type:2001, LSID:0 Adv-Router:10.2.2.2
  Type:2009, LSID:0 Adv-Router:10.2.2.2
  Type:0008, LSID:2052 Adv-Router:10.2.2.2
  Type:4005, LSID:91 Adv-Router:10.2.2.2
  Type:4005, LSID:92 Adv-Router:10.2.2.2
  Type:4005, LSID:93 Adv-Router:10.2.2.2
  Type:4005, LSID:94 Adv-Router:10.2.2.2
  Type:4005, LSID:95 Adv-Router:10.2.2.2
  Type:4005, LSID:96 Adv-Router:10.2.2.2
  Type:4005, LSID:97 Adv-Router:10.2.2.2
  Type:4005, LSID:98 Adv-Router:10.2.2.2
  Type:4005, LSID:99 Adv-Router:10.2.2.2
  Type:4005, LSID:100 Adv-Router:10.2.2.2
  Type:4005, LSID:101 Adv-Router:10.2.2.2
  Type:4005, LSID:102 Adv-Router:10.2.2.2
  Type:4005, LSID:103 Adv-Router:10.2.2.2
  Type:4005, LSID:104 Adv-Router:10.2.2.2
  Type:4005, LSID:105 Adv-Router:10.2.2.2
  Type:4005, LSID:106 Adv-Router:10.2.2.2
  Type:4005, LSID:107 Adv-Router:10.2.2.2
  Type:4005, LSID:108 Adv-Router:10.2.2.2
  Type:2002, LSID:644 Adv-Router:10.4.4.8
```

debug ipv6 ospf packet-lsa-update

Syntax: [no] **debug ipv6 ospf packet-lsa-update**

This command displays all OSPFv3 LSA update packets in rx or tx mode as shown in the following example.

```
Brocade# debug ipv6 ospf packet-lsa-update
OSPFv3:  packet-lsa-update debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9
OSPFv3: (fe80::224:38ff:fe21:6400->OSPFv3: fe80::224:38ff:fed6:7800)
  Type:2001, LSID:0 Adv:10.4.4.8 SEQ:80000019 AGE:41
  Type:2009, LSID:0 Adv:10.4.4.8 SEQ:80000001 AGE:41
  Type:4005, LSID:31 Adv:10.4.4.8 SEQ:80000001 AGE:41
  Type:4005, LSID:32 Adv:10.4.4.8 SEQ:80000001 AGE:41
  Type:4005, LSID:33 Adv:10.4.4.8 SEQ:80000001 AGE:41
  Type:4005, LSID:34 Adv:10.4.4.8 SEQ:80000001 AGE:41
  Type:4005, LSID:35 Adv:10.4.4.8 SEQ:80000001 AGE:41
  Type:0008, LSID:644 Adv:10.4.4.8 SEQ:80000001 AGE:41
Debug: Jan  1 19:14:43 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: fe80::224:38ff:fe21:6400)
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:80000007 AGE:168
  Type:2001, LSID:0 Adv:10.2.2.2 SEQ:80000019 AGE:42
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:40
  Type:0008, LSID:2052 Adv:10.2.2.2 SEQ:80000001 AGE:42
  Type:4005, LSID:109 Adv:10.2.2.2 SEQ:80000001 AGE:42
  Type:4005, LSID:110 Adv:10.2.2.2 SEQ:80000001 AGE:42
  Type:4005, LSID:111 Adv:10.2.2.2 SEQ:80000001 AGE:42
  Type:4005, LSID:112 Adv:10.2.2.2 SEQ:80000001 AGE:42
  Type:4005, LSID:113 Adv:10.2.2.2 SEQ:80000001 AGE:42
  Type:4005, LSID:114 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:115 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:116 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:117 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:118 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:119 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:120 Adv:10.2.2.2 SEQ:80000001 AGE:41
```

```
  Type:4005, LSID:121 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:122 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:123 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:124 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:125 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:4005, LSID:126 Adv:10.2.2.2 SEQ:80000001 AGE:41
  Type:2002, LSID:644 Adv:10.4.4.8 SEQ:80000007 AGE:168
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:80000007 AGE:3600
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2002, LSID:644 Adv:10.4.4.8 SEQ:80000007 AGE:3600
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:80000007 AGE:3600
Debug: Jan  1 19:14:43 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:2001, LSID:0 Adv:10.2.2.2 SEQ:8000001a AGE:1
Debug: Jan  1 19:14:43 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:2001, LSID:0 Adv:10.2.2.2 SEQ:8000001a AGE:1
Debug: Jan  1 19:14:43 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 19:14:43 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
  Type:2009, LSID:0 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2001, LSID:0 Adv:10.4.4.8 SEQ:8000001a AGE:1
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2002, LSID:644 Adv:10.4.4.8 SEQ:80000008 AGE:1
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2009, LSID:19320 Adv:10.4.4.8 SEQ:80000008 AGE:1
Debug: Jan  1 19:14:43 OSPFv3: Snd LSUpdate on ve 9 OSPFv3:
(fe80::224:38ff:fe21:6400->OSPFv3: ff02::5)
  Type:2009, LSID:0 Adv:10.4.4.8 SEQ:80000001 AGE:3600
  Type:2009, LSID:0 Adv:10.4.4.8 SEQ:80000001 AGE:3600
Debug: Jan  1 19:14:48 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:4005, LSID:110 Adv:10.2.2.2 SEQ:80000001 AGE:3600
  Type:4005, LSID:111 Adv:10.2.2.2 SEQ:80000001 AGE:3600
Debug: Jan  1 19:14:48 OSPFv3: Rcv LSUpdate on ve 9  OSPFv3:
(fe80::224:38ff:fed6:7800->OSPFv3: ff02::5)
  Type:4005, LSID:110 Adv:10.2.2.2 SEQ:80000001 AGE:3600
  Type:4005, LSID:111 Adv:10.2.2.2 SEQ:80000001 AGE:3600
```

debug ipv6 ospf route

Syntax: [no] **debug ipv6 ospf route**

This command displays all OSPFv3 routes as shown in the following example.

```
Brocade# debug ipv6 ospf route
OSPFv3:  route debugging is on
```

```
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:19:50 OSPFv3: add to spf list Inst
vrf id 0, flag 0
Debug: Jan  1 19:19:50 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:19:50 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:19:50 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:19:50 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:19:50 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:19:51 OSPFv3: delete from spf list Inst vrf id 0, flag 1
Debug: Jan  1 19:19:51 OSPFv3 ROUTE: release semaphore 1
Debug: Jan  1 19:19:51 OSPFv3: SPF: Calculation for area 0.0.0.0
Debug: Jan  1 19:19:51 OSPFv3: SPF: installing vertex 10.4.4.8
Debug: Jan  1 19:19:51 OSPFv3: SPF: Calculation for area 0.0.0.0 done
Debug: Jan  1 19:19:51 OSPFv3: ROUTE: Router (e:1, b:0) 10.2.2.2 removed from area
0.0.0.0
Debug: Jan  1 19:19:51 OSPFv3: ROUTE: Router (e:0, b:0) 10.4.4.8 removed from area
0.0.0.0
Debug: Jan  1 19:19:51 OSPFv3: SPF finished  at 0
Debug: Jan  1 19:19:51 OSPFv3: Calculating Intra Area routes for area 0.0.0.0
Debug: Jan  1 19:19:51 OSPFv3: INTRA AREA ROUTE: Calculating Intra Area Stub
Routes
Debug: Jan  1 19:19:51 OSPFv3: INTRA AREA ROUTE: Can't find Prefix LSA for id
0.0.0.0 AdvRouter 10.4.4.8
Debug: Jan  1 19:19:51 OSPFv3: Intra area route calculation finished  at 706906
Debug: Jan  1 19:19:51 OSPFv3: TRANSIT ROUTE: Discarding routes with nexthop
unresolved
Debug: Jan  1 19:19:51 OSPFv3: Transit route calculation finished at 706906
Debug: Jan  1 19:19:51 OSPFv3: External route calculation finished at 706906
Debug: Jan  1 19:19:51 OSPFv3: ROUTE: Validate routing table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
```

```
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2000::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2000::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2000::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:19:51 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:19:51 OSPFv3: SPF_cleanup: FINISHED. Next spf can start at
707006, last end at 706906
Debug: Jan  1 19:19:51 OSPFv3: Route calculation (26) finished at 706906
Debug: Jan  1 19:19:55 OSPFv3: add to spf list Inst vrf id 0, flag 0
```

debug ipv6 ospf route-calc-external

Syntax: [no] debug ipv6 ospf route-calc-external

This command displays external route calculations as shown in the following example.

```
Brocade# debug ipv6 ospf route-calc-external
OSPFv3:  route-calc-external debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE:
External LSA(ID= 37) is Self-originated:
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: External LSA(ID= 39) is
Self-originated:
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
```

```
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: External LSA(ID= 36) is
Self-originated:
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: External LSA(ID= 38) is
Self-originated:
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: External LSA(ID= 40) is
Self-originated:
Debug: Jan  1 19:21:07 OSPFv3: EXTERNAL ROUTE: Forwarding flag specified
```

debug ipv6 ospf route-calc-inter-area

Syntax: [no] **debug ipv6 ospf route-calc-inter-area**

This command displays inter-area route calculations as shown in the following example.

```
Brocade# debug ipv6 ospf route-calc-inter-area
OSPFv3:  route-calc-inter-area debugging is on
Brocade(config-vif-7)# ipv6 ospf area 1
Brocade(config-vif-7)# sOSPFv3: INTER AREA ROUTE: Inter Area Prefix LSA(ID= 2,
prefix 2001:DB8::/64) is Self-originated. Area id 0.0.0.0
OSPFv3: INTER AREA ROUTE: Inter Area Prefix LSA(ID= 2, prefix 2001:DB8::/64) is
Self-originated. Area id 0.0.0.0
Brocade(config-vif-7)# disable
Brocade(config-vif-7)# OSPFv3: INTER AREA ROUTE: Inter Area Prefix LSA(ID= 2,
prefix 2001:DB8::/64) is Self-originated. Area id 0.0.0.0
Brocade(config-vif-7)# enable
Brocade(config-vif-7)# OSPFv3: INTER AREA ROUTE: Inter Area Prefix LSA(ID= 11,
prefix 2001:DB8::/64) is Self-originated. Area id 0.0.0.0
Brocade(config)# Debug: Jan  1 00:32:22 OSPFv3: INTER AREA ROUTE: Inter Area route
for prefix 2001:DB8::/64 created
```

debug ipv6 ospf route-calc-intra-area

Syntax: [no] **debug ipv6 ospf route-calc-intra-area**

This command displays intra-area route calculations as shown in the following example.

```
Brocade# debug ipv6 ospf route-calc-intra-area
OSPFv3:  route-calc-intra-area debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:22:33 OSPFv3: Calculating Intra
Area routes for area 0.0.0.0
Debug: Jan  1 19:22:33 OSPFv3: INTRA AREA ROUTE: Calculating Intra Area Stub
Routes
Debug: Jan  1 19:22:33 OSPFv3: INTRA AREA ROUTE: Can't find Prefix LSA for id
0.0.0.0 AdvRouter 10.4.4.8
Debug: Jan  1 19:22:43 OSPFv3: Calculating Intra Area routes for area 0.0.0.0
Debug: Jan  1 19:22:43 OSPFv3: INTRA AREA ROUTE: Calculating Intra Area Stub
Routes
Debug: Jan  1 19:22:43 OSPFv3: INTRA AREA ROUTE: Can't find Prefix LSA for id
0.0.0.0 AdvRouter 10.4.4.8
```

debug ipv6 ospf route-calc-spf

Syntax: [no] **debug ipv6 ospf route-calc-spf**

This command displays SPF route calculations as shown in the following example.

```
Brocade# debug ipv6 ospf route-calc-spf
OSPFv3:  route-calc-spf debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:26:13 OSPFv3: Schedule routing
table build
Debug: Jan  1 19:26:13 OSPFv3: add to spf list Inst vrf id 0, flag 0
Debug: Jan  1 19:26:19 OSPFv3: delete from spf list Inst vrf id 0, flag 1
Debug: Jan  1 19:26:19 OSPFv3: SPF: Calculation for area 0.0.0.0
Debug: Jan  1 19:26:19 OSPFv3: SPF: installing vertex 10.4.4.8
Debug: Jan  1 19:26:19 OSPFv3: SPF: Calculation for area 0.0.0.0 done
Debug: Jan  1 19:26:19  OSPFv3 SPF_cleanup: Waiting for holdtime to elapse
Debug: Jan  1 19:26:19 OSPFv3: SPF_cleanup: FINISHED. Next spf can start at
710947, last end at 710847
Debug: Jan  1 19:26:53 OSPFv3: Schedule routing table build
Debug: Jan  1 19:26:53 OSPFv3: add to spf list Inst vrf id 0, flag 0
Debug: Jan  1 19:26:58 OSPFv3: delete from spf list Inst vrf id 0, flag 1
Debug: Jan  1 19:26:58 OSPFv3: SPF: Calculation for area 0.0.0.0
Debug: Jan  1 19:26:58 OSPFv3: SPF: installing vertex 10.4.4.8
Debug: Jan  1 19:26:58 OSPFv3: SPF: 10.4.4.8:644 is the first hop
Debug: Jan  1 19:26:58 OSPFv3: SPF : 10.4.4.8:644 nexthop :: ifindex 644
Debug: Jan  1 19:26:58 OSPFv3: SPF: Examining Vertex: 10.4.4.8:644
Debug: Jan  1 19:26:58 OSPFv3: SPF: new node added to candidate list: 10.4.4.8:644
Debug: Jan  1 19:26:58 OSPFv3: SPF: installing vertex 10.4.4.8:644
Debug: Jan  1 19:26:58 OSPFv3: SPF: Ignore link description to myself
Debug: Jan  1 19:26:58 OSPFv3: SPF : 10.2.2.2:0 nexthop fe80::224:38ff:fed6:7800
ifindex 644
Debug: Jan  1 19:26:58 OSPFv3: SPF: Examining Vertex: 10.2.2.2:0
Debug: Jan  1 19:26:58 OSPFv3: SPF: new node added to candidate list: 10.2.2.2:0
Debug: Jan  1 19:26:58 OSPFv3: SPF: installing vertex 10.2.2.2:0
Debug: Jan  1 19:26:58 OSPFv3: SPF: 10.4.4.8:644 inherits 10.2.2.2:0's
nexthop_list
Debug: Jan  1 19:26:58 OSPFv3: SPF: Examining Vertex: 10.4.4.8:644
Debug: Jan  1 19:26:58 OSPFv3: SPF: already in SPF tree: 10.4.4.8:644
Debug: Jan  1 19:26:58 OSPFv3: SPF: Calculation for area 0.0.0.0 done
Debug: Jan  1 19:26:58  OSPFv3 SPF_cleanup: Waiting for holdtime to elapse
```

debug ipv6 ospf route-install

Syntax: [no] **debug ipv6 ospf route-install**

This command displays all OSPFv3 installed routes as shown in the following example.

```
Brocade# debug ipv6 ospf route-install
OSPFv3:  route-install debugging is on
Brocade(config-if-e1000-8/9)# Debug: Jan  1 19:48:12 OSPFv3: Redistribute:remove
prefix 2001:DB8::/64
Debug: Jan  1 19:48:12 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:48:12 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:48:12 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:48:12 OSPFv3: Redistribute:remove prefix 2001:DB8::/64
Debug: Jan  1 19:48:12 OSPFv3: ROUTE: Router (e:1, b:0) 10.2.2.2 removed from area
0.0.0.0
Debug: Jan  1 19:48:12 OSPFv3: ROUTE: Router (e:0, b:0) 10.4.4.8 removed from area
0.0.0.0
Debug: Jan  1 19:48:12 OSPFv3: ROUTE: Validate routing table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
```

```
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2001:DB8::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2000::/64, nexthop cnt 1,
route_info->flag 0x1
Debug: Jan  1 19:48:12 OSPFv3: Route 2000::/64 removed from RIB
Debug: Jan  1 19:48:12 OSPFv3: Route 2000::/64 deleted from OSPF Table
Debug: Jan  1 19:48:12 OSPFv3: Validating route 2001:DB8::/64, nexthop cnt 1,
route_info->flag 0x1
```

DHCPv6 debug commands

This section describes the debug commands used for troubleshooting issues related to Dynamic Host Configuration Protocol for IPv6 (DHCPv6) relay agent and snooping activity.

debug ipv6 dhcp relay

Syntax: [no] **debug ipv6 dhcp relay**

This command displays debug information related to DCHPv6 relay agent. The following ouput will be displayed when you enable the DHCPv6 relay agent on the Ethernet interface 8/2.

```
Brocade# debug ipv6 dhcp relay
DHCP6: dhcp6_relay enabled on port ethernet 8/2
```

debug ipv6 dhcp snooping

Syntax: [no] **debug ipv6 dhcp snooping**

This command enables debugging of the DHCPv6 snooping activity.

```
Brocade# debug ipv6 dhcp snooping
Brocade(config-vif-11)# DHCPv6: snooping mapped to outgoing port 2/1/24 vlan 14
VRF 1
DHCPv6: snooping on trusted port 3/1/23,  VRF 1, 2001:DB8::b->0000.007d.7a3e
```

debug ipv6 dhcp pd all

Syntax: [no] **debug ipv6 dhcp pd all**

This command enables all IPv6 Prefix Delegation (PD) debugging. Command output resembles the following example.

```
Brocade# debug ipv6 dhcp pd all
DHCP6 PD: Adding route to DHCP DP table, 2001:DB8::4/64 with nexthop
fe80::100:120:30 interface e 1/1/15
DHCP6: dhcp6_sync_dhcp6_msg_with_standby called
DHCP6: dhcp6_sync_node_pack_dhcp6_msg_to_standby called
DHCP6: dhcp6_sync_node_ack_dhcp6_msg_from_standby called
DHCP6 PD: Removing route from DHCP DP table, lifetime expired, 2001:DB8::4/64 with
nexthop fe80::100:120:30 interface e 1/1/15
DHCP6: dhcp6_sync_dhcp6_msg_with_standby called
DHCP6: dhcp6_sync_node_pack_dhcp6_msg_to_standby called
DHCP6: dhcp6_sync_node_ack_dhcp6_msg_from_standby called
```

debug ipv6 dhcp pd flash

Syntax: [no] **debug ipv6 dhcp pd flash**

This command enables PD debugging for read and write to flash file. Command output resembles the following example.

```
Brocade# debug ipv6 dhcp pd flash
DHCP6 PD: writing data to file dhcp6_delegated_prefixes_data
DHCP6 PD: writing data to file dhcp6_delegated_prefixes_data
DHCP6 PD: Removed file dhcp6_delegated_prefixes_data
```

debug ipv6 dhcp pd pd-option

Syntax: [no] **debug ipv6 dhcp pd pd-option**

This command enables debugging for processing of PD options in DHCPv6 messages. Command output resembles the following example.

```
Brocade# debug ipv6 dhcp pd pd-option
DHCP6 PD: Adding route to DHCP DP table, 2001:DB8::4/64 with nexthop
fe80::100:120:30 interface e 1/1/15
DHCP6 PD: Removing route from DHCP DP table, lifetime expired, 2001:DB8::4/64 with
nexthop fe80::100:120:30 interface e 1/1/15
```

debug ipv6 dhcp pd static

Syntax: [no] **debug ipv6 dhcp pd static**

This command enables debugging of DHCPv6 interface with IPv6 static route module. Command output resembles the following example.

```
Brocade# debug ipv6 dhcp pd static
DHCP6 PD: Adding static route 2001:DB8::4/64 with nexthop fe80::100:120:30
interface e 1/1/15
DHCP6 PD: Removing static route 2001:DB8::4/64 with nexthop fe80::100:120:30
interface e 1/1/15
```

debug ipv6 dhcp sync

Syntax: [no] **debug ipv6 dhcp sync**

This command enables debugging of DHCPv6 synchronization between active and standby MPs. Command output resembles the following example.

```
Brocade# debug ipv6 dhcp sync
DHCP6: dhcp6_sync_node_pack_dhcp6_msg_to_standby called
DHCP6: dhcp6_sync_node_ack_dhcp6_msg_from_standby called
```

DHCPv6 show commands

This section describes the show commands that display debuuging information related to DHCPv6 relay agent.

show ipv6 dhcp-relay debug

Syntax: show ipv6 dhcp-relay debug

This command displays DCHPv6 statistics as shown in the following example.

```
Brocade# show ipv6 dhcp-relay debug
DHCP6 Error Counters:
   rx_packet_dropped_dhcp6_relay_disabled: 0
   rx_packet_dropped_no_relay_option: 0
   rx_packet_dropped_no_buffer: 0
   send_packet_fail_no_address: 0
   send_packet_fail_no_buffer: 0
   send_packet_fail_udp: 0
DHCP6 Other Counters:
   dhcp6_relay_enabled_cnt: 1
```

IPv6 PIM debug commands

This section describes the debug commands used for troubleshooting issues related to IPv6 Protocol Independent Multicast (PIM).

debug ipv6 pim bootstrap

Syntax: [no] **debug ipv6 pim bootstrap**

This command enables IPv6 PIM bootstrap debugging and the output resembles the following example.

```
Brocade# debug ipv6 pim bootstrap
PIM6:  bootstrap debugging is on
Brocade# PIM-BSR.VRF0: Prefer BSR 2001:DB8::12(Pr 120) over current BSR
2001:DB8::12(Pr 120)
PIM-BSR.VRF0: Intf v1844 - accept BSM from BSR 2001:DB8::12(Pr 120), local state
CandBSR, curr BSR 2001:DB8::12 (Pr 120)
```

debug ipv6 pim clear

Syntax: [no] **debug ipv6 pim clear**

This command clears all the IPv6 PIM debug settings. The output will be similar to the following if no settings are enabled.

```
Brocade# debug ip pim clear
no debug ip pim is enabled
```

debug ipv6 pim event

Syntax: [no] **debug ipv6 pim event**

This command enables debugging of IPv6 PIM events and the output resembles the following example.

```
Brocade# debug ipv6 pim event
PIM6:  event debugging is on
PIM-EVT.VRF0: Receive intf v1844 Dn state_notify
Mcast6v6 receive event 16
PIM6-EVT: Received vport v1844/1/1/11 Dn state notify
Mcast6v6 receive event 16
PIM6-EVT: Received vport v1844/1/1/12 Dn state notify
Mcast6v6 receive event 16
PIM6-EVT: Received vport v1844/1/1/13 Dn state notify
Mcast6v6 receive event 16
Mcast6v6 receive event 13
PIM6-EVT: Received vport v1844/1/1/14 Dn state notify
MC-EVT: Port v1844 state changed to DOWN
PIM-EVT.VRF0: Receive intf v1844 Dn state_notify
PIM6-EVT: Received vport v1844/1/1/14 Dn state notify
```

debug ipv6 pim group

Syntax: [no] **debug ipv6 pim group** *group_addr*

This command displays the IPv6 PIM related debugging information for the specified PIM group. The following is the truncated output.

```
Brocade# debug ipv6 pim group ff0::8
        PIM6:  group ff0::8 debugging is on
Brocade# PIM.VRF0: Deallocated hw resources for (7400::200 ff0::8), parent:
NIL,Nil. olist not empty
VRF0 Flow Entry Delete (2001:DB8::200 ff0::1) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::2) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::3) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::4) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::5) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::6) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::7) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::8) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::9) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::a) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::b) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::c) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::d) HW bits: 1 1
VRF0 Flow Entry Delete (2001:DB8::200 ff0::e) HW bits: 1 1
```

debug ipv6 pim join-prune

Syntax: [no] **debug ipv6 pim join-prune**

This command controls join prune processing and displays debugging information related to the IPv6 join prune messages. The following is the truncated sample output from the **debug ipv6 pim join-prune** command.

```
Brocade# debug ipv6 pim join-prune
PIM6:  join-prune debugging is on
Brocade# PIMSM.0: BEGIN J/P proc: rpf_nbr fe80::768e:f8ff:fe34:3c80, to_me=1 from
fe80::768e:f8ff:fe3e:9600, intf v1844 ----
  Group=ff0::1. Join list: 1 srcs
    J-Src=2001:DB8::200, wc=0, rpt=0, SM=1
      PIM.VRF0: Processing (S,G) Join (2001:DB8::200 ff84::1) from
fe80::768e:f8ff:fe3e:9600, intf v1844,1/1/11
  Group=ff0::1. Prune list: 0 srcs
  Group=ff0::2. Join list: 1 srcs
    J-Src=2001:DB8::200, wc=0, rpt=0, SM=1
      PIM.VRF0: Processing (S,G) Join (2001:DB8::200 ff84::2) from
fe80::768e:f8ff:fe3e:9600, intf v1844,1/1/11
  Group=ff0::2. Prune list: 0 srcs
  Group=ff0::3. Join list: 1 srcs
    J-Src=2001:DB8::200, wc=0, rpt=0, SM=1
      PIM.VRF0: Processing (S,G) Join (2001:DB8::200 ff84::3) from
fe80::768e:f8ff:fe3e:9600, intf v1844,1/1/11
  Group=ff0::3. Prune list: 0 srcs
  Group=ff0::4. Join list: 1 srcs
    J-Src=2001:DB8::200, wc=0, rpt=0, SM=1
      PIM.VRF0: Processing (S,G) Join (2001:DB8::200 ff84::4) from
fe80::768e:f8ff:fe3e:9600, intf v1844,1/1/11
  Group=ff0::4. Prune list: 0 srcs
  Group=ff0::5. Join list: 1 srcs
    J-Src=2001:DB8::200, wc=0, rpt=0, SM=1
      PIM.VRF0: Processing (S,G) Join (2001:DB8::200 ff84::5) from
fe80::768e:f8ff:fe3e:9600, intf v1844,1/1/11
  Group=ff0::5. Prune list: 0 srcs
  Group=ff0::6. Join list: 1 srcs
    J-Src=2001:DB8::200, wc=0, rpt=0, SM=1
      PIM.VRF0: Processing (S,G) Join (2001:DB8::200 ff84::6) from
fe80::768e:f8ff:fe3e:9600, intf v1844,1/1/11
```

debug ipv6 pim nbr-change

Syntax: [no] **debug ipv6 pim nbr-change** [*nbr_address*]

The *nbr_address* variable specifies the neighbor address.

This command controls the IPv6 PIM neighbor changes and the command output resembles the following example.

```
Brocade# debug ipv6 pim nbr-change
PIM6:  nbr-change  debugging is on
Brocade# PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe3e:9600 on intf v1844,
1/1/11-1/1/14
Brocade(config-if-e1000-1/1/11)# disable
Brocade(config-if-e1000-1/1/11)# PIM-NBR.VRF0: Neighbor fe80::768e:f8ff:fe3e:9600
on interface v1844, 1/1/11 deleted on port-down
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe46:5580 on intf v170, 6/1/17
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe46:5580 on intf v170, 6/1/17

Brocade(config-if-e1000-1/1/11)# enable
Brocade(config-if-e1000-1/1/11)# PIM.VRF0: Rx Hello msg from
fe80::768e:f8ff:fe46:5580 on intf v170, 6/1/17
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe46:5580 on intf v170, 6/1/17
PIMV2 rcvd from fe80::768e:f8ff:fe3e:9600 on v1844 (phy e1/1/11), not pim
neighbour
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe3e:9600 on intf v1844, 1/1/11-1/1/14
PIM-NBR.VRF0: Neighbor fe80::768e:f8ff:fe3e:9600 on interface v1844, 1/1/11 genid
changed from 4294953471 to 13824
PIM-NBR.VRF0: GenId changed for Neighbor fe80::768e:f8ff:fe3e:9600 on interface
v1844, 1/1/11 ; new GenId 13824
PIM-NBR.VRF0: Neighbor fe80::768e:f8ff:fe3e:9600 on interface v1844, 1/1/11 added
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe46:5580 on intf v170, 6/1/17
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe46:5580 on intf v170, 6/1/17
PIM.VRF0: Rx Hello msg from fe80::768e:f8ff:fe3e:9600 on intf v1844, 1/1/11-1/1/14
```

debug ipv6 pim packet

Syntax: [no] **debug ipv6 pim packet** *source_address group_address* [**rate**]

- **source_address** - Specifies the IPv6 source address of the PIM packet
- **group_address** - Specifies the IPv6 group address of the PIM packet.
- **rate** - Specifies the hardware rate.

This command displays debugging information about the IPv6 PIM packets. Command output resembles the following example.

```
Brocade# debug ipv6 pim packet
Rx Pkt (2001:DB8::200, ff0::18) on v1244(1/11).
   Rx flow on intf/port diff than entry's intf v1244 or port INVALID
  send MCAST_FIRST_DATA: entry phy port: INVALID, pkt input port: 1/11-1/14.
SPT-F, FAST-F
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 1. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::19) on v1244(1/12).
   Rx flow on intf/port diff than entry's intf v1244 or port INVALID
  send MCAST_FIRST_DATA: entry phy port: INVALID, pkt input port: 1/11-1/14.
SPT-F, FAST-F
* *: 0 pkt,from v1244(e1/12), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
```

```
   switch_to_spt: No. Thrshld: 1, Cnt: 1. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::1a) on v1244(1/13).
   Rx flow on intf/port diff than entry's intf v1244 or port INVALID
  send MCAST_FIRST_DATA: entry phy port: INVALID, pkt input port: 1/11-1/14.
SPT-F, FAST-F
* *: 0 pkt,from v1244(e1/13), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 1. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::1b) on v1244(1/14).
   Rx flow on intf/port diff than entry's intf v1244 or port INVALID
  send MCAST_FIRST_DATA: entry phy port: INVALID, pkt input port: 1/11-1/14.
SPT-F, FAST-F
* *: 0 pkt,from v1244(e1/14), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 1. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::1c) on v1244(1/11).
   Rx flow on intf/port diff than entry's intf v1244 or port INVALID
  send MCAST_FIRST_DATA: entry phy port: INVALID, pkt input port: 1/11-1/14.
SPT-F, FAST-F
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 1. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::14) on v1244(1/11).
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 52. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::15) on v1244(1/12).
* *: 0 pkt,from v1244(e1/12), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 52. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::16) on v1244(1/13).
* *: 0 pkt,from v1244(e1/13), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 51. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::17) on v1244(1/14).
* *: 0 pkt,from v1244(e1/14), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 51. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::14) on v1244(1/11).
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 53. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::14) on v1244(1/11).
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 107. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::15) on v1244(1/12).
* *: 0 pkt,from v1244(e1/12), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 107. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::16) on v1244(1/13).
* *: 0 pkt,from v1244(e1/13), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 106. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::17) on v1244(1/14).
* *: 0 pkt,from v1244(e1/14), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 106. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::14) on v1244(1/11).
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
```

```
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 108. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::14) on v1244(1/11).
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 162. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::15) on v1244(1/12).
* *: 0 pkt,from v1244(e1/12), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 162. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::16) on v1244(1/13).
* *: 0 pkt,from v1244(e1/13), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 161. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::17) on v1244(1/14).
* *: 0 pkt,from v1244(e1/14), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 161. RPT=0, Local-Rcvr=0
Rx Pkt (2001:DB8::200, ff0::14) on v1244(1/11).
* *: 0 pkt,from v1244(e1/11), dup. pkt to v55 phy e1/1, l2=0
   SW forwarded pkt to 1 oif(s)
   switch_to_spt: No. Thrshld: 1, Cnt: 163. RPT=0, Local-Rcvr=0
```

debug ipv6 pim regproc

Syntax: [no] **debug ipv6 pim regproc**

This command controls IPv6 PIM register processing and the command output resembles the following example.

```
Brocade# debug ipv6 pim regproc
PIM6:  regproc debugging is on
Brocade# PIMSM-REG.VRF0: (2001:DB8::200 ff0::1) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::2) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::3) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::4) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::5) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::6) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::7) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::8) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::9) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::a) Sending L2-Null Reg on iface v74
PIMSM-REG.VRF0: (2001:DB8::200 ff0::b) Sending L2-Null Reg on iface v74
```

debug ipv6 pim rp

Syntax: [no] **debug ipv6 pim rp**

This command controls the Rendezvouz Point (RP) processing for IPv6 PIM events and the command output resembles the following example.

```
Brocade# debug ipv6 pim rp
PIM6:  rp  debugging is on
Brocade# PIMv6-CRP.VRF0: Sending Candidate RP Msg (size 46)
PIMv6-CRP.VRF0: Sending Candidate RP Msg (size 46)
```

debug ipv6 pim source

Syntax: [no] **debug ipv6 pim source** *ipaddr*

This command displays the IPv6 PIM related debugging information for the specified source address. The following is the truncated output.

```
Brocade# debug ipv6 pim source 2001:DB8::200
        PIM6:  source 2001:DB8::200 debugging is on
Brocade# PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::1), parent:
NIL,Nil. olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::2), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::3), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::4), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::5), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::6), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::7), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::8), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::9), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::a), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::b), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::c), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::d), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::e), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::f), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::10), parent: NIL,Nil.
olist not empty
PIM.VRF0: Deallocated hw resources for (2001:DB8::200 ff0::11), parent: NIL,Nil.
olist not empty
```

debug ipv6 pim show

Syntax: debug ipv6 pim show

This command displays the IPv6 PIM debug settings.

```
Brocade# debug ipv6 pim show
pim6 debug-enable-any = 1
```

IPv6 RIP debug commands

The following debug commands display debugging information for the IPv6 Routing Information Protocol (RIP).

debug ipv6 rip events

Syntax: [no] debug ipv6 rip events

This command displays debugging information about the IPv6 RIP events.

```
Brocade# debug ipv6 rip events
Brocade(config-vif-11)# RIPng: update timer expired
clear ipv routeRIPng: triggered update
RIPng: garbage prefix 2001:DB8::/64 timer 1, metric 0, tag 0
         from :: on interface NULL
RIPng: garbage prefix 2001:DB8::/64 timer 1, metric 0, tag 0
         from :: on interface NULL
RIPng: Adding local connected route 2001:DB8::1/64 on interface v11
RIPng: Adding local connected route 2001:DB8::1/64 on interface v12
RIPng: update timer expired
RIPng: Redistribute add route 2001:DB8::/64, type CONNECTED (1/0)
RIPng: Redistribute add route 2001:DB8::/64, type CONNECTED (1/0)
```

debug ipv6 rip receive

Syntax: [no] debug ipv6 rip receive

This command displays debugging information about the IPv6 RIP packets received.

```
Brocade# debug ipv6 rip receive
Brocade(config-vif-11)# RIPng: received packet from fe80::224:38ff:fe9f:7500 port
521 on interface v11
       command response version 1 packet size 64
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
RIPng: received packet from fe80::224:38ff:fe9f:7500 port 521 on interface v12
       command response version 1 packet size 64
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
RIPng: received packet from fe80::224:38ff:fe9f:7500 port 521 on interface v11
       command response version 1 packet size 64
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
RIPng: received packet from fe80::224:38ff:fe9f:7500 port 521 on interface v11
       command response version 1 packet size 64
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
RIPng: received packet from fe80::224:38ff:fe9f:7500 port 521 on interface v11
       command response version 1 packet size 64
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0
RIPng: received packet from fe80::224:38ff:fe9f:7500 port 521 on interface v11
       command response version 1 packet size 64
       prefix 2001:DB8::/64 metric 1 tag 0
       prefix 2001:DB8::/64 metric 1 tag 0

       prefix 2001:DB8::/64 metric 1 tag 0
```

Chapter

Multicast Diagnostics 6

This chapter contains the following sections:

MSDP debug commands

Multicast Source Discovery Protocol (MSDP) is used by Protocol-Independent Multicast (PIM) Sparse routers to exchange routing information for PIM Sparse multicast groups across PIM Sparse domains. Devices running MSDP can discover PIM Sparse sources that are in other PIM Sparse domains.

The following **debug ip msdp** commands generate MSDP information.

debug ip msdp

Syntax: [no] **debug ip msdp** [**alarms** | **events** | **message**]

- **alarms** - Displays information about the MSDP alarms.
- **events** - Displays information about the MSDP events.
- **message** - Displays information about the MSDP messages.

The **debug ip msdp** command generates information about MSDP alarms, events, and messages.

debug ip msdp alarms

Syntax: [no] **debug ip msdp alarms**

This command generates information about the MSDP processing alarms.

```
Brocade# debug ip msdp alarms
: MSDP: S=xxxxxxx P=0 Initiate Transport Connection to MSDP peer
: MSDP: S=xxxxxxx P=0 Initiate Transport Connection to MSDP peer
```

debug ip msdp events

Syntax: [no] **debug ip msdp events**

This command tracks originating SA-advertisements, major peer events, and peer-keepalive timer events.

```
Brocade# debug ip msdp events
: MSDP: 10.120.120.45: Process START event, local = 10.120.120.31
: MSDP: S=xxxxxxx P=0 Initiate Transport Connection to MSDP peer
: MSDP: 10.120.120.45: TCP Connection to Remote Peer is Open
: MSDP: 10.120.120.45: MSDP-TCP Connection opened
: MSDP: 10.120.120.45: TCP_OPEN DONE, State 4
: MSDP: Remote Peer closed TCP connection
: MSDP: 10.120.120.45 Remote Peer closed TCP connection
: MSDP: Originating SA
: MSDP: Originating SA
: MSDP: Originating SA
: MSDP: Originating SA
: MSDP: Originating SA
```

debug ip msdp message

Syntax: [no] **debug ip msdp message**

This command generates information about MSDP messages received, transmitted and forwarded, and flag errors in the MSDP messages.

```
Brocade# debug ip msdp message
: MSDP: 10.0.0.32: Xmt KA
: MSDP: 10.0.0.32: State=4, Rcv SA 20 bytes
  RP: 10.0.0.32, Num SA: 1 10.17.17.1,237.14.18.3
: MSDP: 10.0.0.32: State=4, Rcv KA
: MSDP: 10.0.0.32: State=4, Rcv SA 20 bytes
  RP: 10.0.0.32, Num SA: 1 10.17.17.1,237.14.18.3
: MSDP: 10.0.0.32: State=4, Rcv KA
: MSDP: 10.0.0.32: Xmt SA 32 bytes
  RP: 10.0.0.45, Num SA: 2 10.14.14.14,237.14.18.3 10.17.17.1,237.14.18.3
: MSDP: 10.0.0.32: Xmt KA
: MSDP: 10.0.0.32: State=4, Rcv SA 20 bytes
  RP: 10.0.0.32, Num SA: 1 10.17.17.1,237.14.18.3
```

IGMP debug commands

The Internet Group Management Protocol (IGMP) allows an IPv4 system to communicate IP multicast group membership information to its neighboring routers. The routers in turn limit the multicast of IP packets with multicast destination addresses to only those interfaces on the router that are identified as IP multicast group members.

The following commands display the information related to IGMP debugging.

debug ip igmp add-del-oif

Syntax: [no] **debug ip igmp add-del-oif**

This command displays information about the addition or deletion of the outgoing interfaces (OIFs).

```
Brocade# debug ip igmp add-del-oif
add-del-oif enabled
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp add-del-oif is enabled
Brocade# debug ip igmp
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.147 resp=100, pkt S=10.17.17.222 to
239.0.0.147, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.148 resp=100, pkt S=10.17.17.222 to
239.0.0.148, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.149 resp=100, pkt S=10.17.17.222 to
239.0.0.149, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.150 resp=100, pkt S=10.17.17.222 to
239.0.0.150, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.151 resp=100, pkt S=10.17.17.222 to
239.0.0.151, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.152 resp=100, pkt S=10.17.17.222 to
239.0.0.152, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.153 resp=100, pkt S=10.17.17.222 to
239.0.0.153, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.154 resp=100, pkt S=10.17.17.222 to
239.0.0.154, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.155 resp=100, pkt S=10.17.17.222 to
239.0.0.155, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.156 resp=100, pkt S=10.17.17.222 to
239.0.0.156, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.157 resp=100, pkt S=10.17.17.222 to
239.0.0.157, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.158 resp=100, pkt S=10.17.17.222 to
239.0.0.158, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.159 resp=100, pkt S=10.17.17.222 to
239.0.0.159, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.160 resp=100, pkt S=10.17.17.222 to
239.0.0.160, on v17 (phy 2/1/17), igmp_size=8

Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.161 resp=100, pkt S=10.17.17.222 to
239.0.0.161, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.162 resp=100, pkt S=10.17.17.222 to
239.0.0.162, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.163 resp=100, pkt S=10.17.17.222 to
239.0.0.163, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.164 resp=100, pkt S=10.17.17.222 to
239.0.0.164, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.165 resp=100, pkt S=10.17.17.222 to
239.0.0.165, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.166 resp=100, pkt S=10.17.17.222 to
239.0.0.166, on v17 (phy 2/1/17), igmp_size=8
nIGMP: rcvd Report-V2(t=22)  g=239.0.0.167 resp=100, pkt S=10.17.17.222 to
239.0.0.167, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.168 resp=100, pkt S=10.17.17.222 to
239.0.0.168, on v17 (phy 2/1/17), igmp_size=8
o IGMP: rcvd Report-V2(t=22)  g=239.0.0.169 resp=100, pkt S=10.17.17.222 to
239.0.0.169, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.170 resp=100, pkt S=10.17.17.222 to
239.0.0.170, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.171 resp=100, pkt S=10.17.17.222 to
239.0.0.171, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.172 resp=100, pkt S=10.17.17.222 to
239.0.0.172, on v17 (phy 2/1/17), igmp_size=8
dIGMP: rcvd Report-V2(t=22)  g=239.0.0.173 resp=100, pkt S=10.17.17.222 to
239.0.0.173, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.174 resp=100, pkt S=10.17.17.222 to
239.0.0.174, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.175 resp=100, pkt S=10.17.17.222 to
239.0.0.175, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.176 resp=100, pkt S=10.17.17.222 to
239.0.0.176, on v17 (phy 2/1/17), igmp_size=8
```

```
eIGMP: rcvd Report-V2(t=22)  g=239.0.0.177 resp=100, pkt S=10.17.17.222 to
239.0.0.177, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.178 resp=100, pkt S=10.17.17.222 to
239.0.0.178, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.179 resp=100, pkt S=10.17.17.222 to
239.0.0.179, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.180 resp=100, pkt S=10.17.17.222 to
239.0.0.180, on v17 (phy 2/1/17), igmp_size=8
gIGMP: rcvd Report-V2(t=22)  g=239.0.0.181 resp=100, pkt S=10.17.17.222 to
239.0.0.181, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.182 resp=100, pkt S=10.17.17.222 to
239.0.0.182, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.183 resp=100, pkt S=10.17.17.222 to
239.0.0.183, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.184 resp=100, pkt S=10.17.17.222 to
239.0.0.184, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.185 resp=100, pkt S=10.17.17.222 to
239.0.0.185, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.186 resp=100, pkt S=10.17.17.222 to
239.0.0.186, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.187 resp=100, pkt S=10.17.17.222 to
239.0.0.187, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.188 resp=100, pkt S=10.17.17.222 to
239.0.0.188, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.189 resp=100, pkt S=10.17.17.222 to
239.0.0.189, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.190 resp=100, pkt S=10.17.17.222 to
239.0.0.190, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.191 resp=100, pkt S=10.17.17.222 to
239.0.0.191, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.192 resp=100, pkt S=10.17.17.222 to
239.0.0.192, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.193 resp=100, pkt S=10.17.17.222 to
239.0.0.193, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.194 resp=100, pkt S=10.17.17.222 to
239.0.0.194, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.195 resp=100, pkt S=10.17.17.222 to
239.0.0.195, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.196 resp=100, pkt S=10.17.17.222 to
239.0.0.196, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.197 resp=100, pkt S=10.17.17.222 to
239.0.0.197, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.198 resp=100, pkt S=10.17.17.222 to
239.0.0.198, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.199 resp=100, pkt S=10.17.17.222 to
239.0.0.199, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.200 resp=100, pkt S=10.17.17.222 to
239.0.0.200, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.201 resp=100, pkt S=10.17.17.222 to
239.0.0.201, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.202 resp=100, pkt S=10.17.17.222 to
239.0.0.202, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.203 resp=100, pkt S=10.17.17.222 to
239.0.0.203, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.204 resp=100, pkt S=10.17.17.222 to
239.0.0.204, on v17 (phy 2/1/17), igmp_size=8
bIGMP: rcvd Report-V2(t=22)  g=239.0.0.205 resp=100, pkt S=10.17.17.222 to
239.0.0.205, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.206 resp=100, pkt S=10.17.17.222 to
239.0.0.206, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.207 resp=100, pkt S=10.17.17.222 to
239.0.0.207, on v17 (phy 2/1/17), igmp_size=8
uIGMP: rcvd Report-V2(t=22)  g=239.0.0.208 resp=100, pkt S=10.17.17.222 to
239.0.0.208, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.209 resp=100, pkt S=10.17.17.222 to
239.0.0.209, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.210 resp=100, pkt S=10.17.17.222 to
239.0.0.210, on v17 (phy 2/1/17), igmp_size=8
gIGMP: rcvd Report-V2(t=22)  g=239.0.0.211 resp=100, pkt S=10.17.17.222 to
239.0.0.211, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.212 resp=100, pkt S=10.17.17.222 to
239.0.0.212, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.213 resp=100, pkt S=10.17.17.222 to
239.0.0.213, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.214 resp=100, pkt S=10.17.17.222 to
239.0.0.214, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.215 resp=100, pkt S=10.17.17.222 to
239.0.0.215, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.216 resp=100, pkt S=10.17.17.222 to
239.0.0.216, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.217 resp=100, pkt S=10.17.17.222 to
239.0.0.217, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.218 resp=100, pkt S=10.17.17.222 to
239.0.0.218, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.219 resp=100, pkt S=10.17.17.222 to
239.0.0.219, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.220 resp=100, pkt S=10.17.17.222 to
239.0.0.220, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.221 resp=100, pkt S=10.17.17.222 to
239.0.0.221, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.222 resp=100, pkt S=10.17.17.222 to
239.0.0.222, on v17 (phy 2/1/17), igmp_size=8
aIGMP: rcvd Report-V2(t=22)  g=239.0.0.223 resp=100, pkt S=10.17.17.222 to
239.0.0.223, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.224 resp=100, pkt S=10.17.17.222 to
239.0.0.224, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.225 resp=100, pkt S=10.17.17.222 to
239.0.0.225, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.226 resp=100, pkt S=10.17.17.222 to
239.0.0.226, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.227 resp=100, pkt S=10.17.17.222 to
239.0.0.227, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.228 resp=100, pkt S=10.17.17.222 to
239.0.0.228, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.229 resp=100, pkt S=10.17.17.222 to
239.0.0.229, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.230 resp=100, pkt S=10.17.17.222 to
239.0.0.230, on v17 (phy 2/1/17), igmp_size=8

Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Turn off stack trace of write table
Disable shadow register write will disable all writes from shadow to hardware
```

debug ip igmp add-del-oif all

Syntax: [no] **debug ip igmp add-del-oif all**

This command displays information about the addition or deletion of all the OIFs that are IGMP-enabled.

```
Brocade# debug ip igmp add-del-oif all
add-del-oif all enabled
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp add-del-oif all is enabled
Brocade# debug ip igmp
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.99 resp=100, pkt S=10.17.17.222 to
239.0.0.99, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.100 resp=100, pkt S=10.17.17.222 to
239.0.0.100, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.101 resp=100, pkt S=10.17.17.222 to
239.0.0.101, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.102 resp=100, pkt S=10.17.17.222 to
239.0.0.102, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.103 resp=100, pkt S=10.17.17.222 to
239.0.0.103, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.104 resp=100, pkt S=10.17.17.222 to
239.0.0.104, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.105 resp=100, pkt S=10.17.17.222 to
239.0.0.105, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.106 resp=100, pkt S=10.17.17.222 to
239.0.0.106, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.107 resp=100, pkt S=10.17.17.222 to
239.0.0.107, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.108 resp=100, pkt S=10.17.17.222 to
239.0.0.108, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.109 resp=100, pkt S=10.17.17.222 to
239.0.0.109, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.110 resp=100, pkt S=10.17.17.222 to
239.0.0.110, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.111 resp=100, pkt S=10.17.17.222 to
239.0.0.111, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.112 resp=100, pkt S=10.17.17.222 to
239.0.0.112, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.113 resp=100, pkt S=10.17.17.222 to
239.0.0.113, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.114 resp=100, pkt S=10.17.17.222 to
239.0.0.114, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.115 resp=100, pkt S=10.17.17.222 to
239.0.0.115, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.116 resp=100, pkt S=10.17.17.222 to
239.0.0.116, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.117 resp=100, pkt S=10.17.17.222 to
239.0.0.117, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.118 resp=100, pkt S=10.17.17.222 to
239.0.0.118, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.119 resp=100, pkt S=10.17.17.222 to
239.0.0.119, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.120 resp=100, pkt S=10.17.17.222 to
239.0.0.120, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.121 resp=100, pkt S=10.17.17.222 to
239.0.0.121, on v17 (phy 2/1/17), igmp_size=8

Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.122 resp=100, pkt S=10.17.17.222 to
239.0.0.122, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.123 resp=100, pkt S=10.17.17.222 to
239.0.0.123, on v17 (phy 2/1/17), igmp_size=8
```

```
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.124 resp=100, pkt S=10.17.17.222 to
239.0.0.124, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.125 resp=100, pkt S=10.17.17.222 to
239.0.0.125, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.126 resp=100, pkt S=10.17.17.222 to
239.0.0.126, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.127 resp=100, pkt S=10.17.17.222 to
239.0.0.127, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.128 resp=100, pkt S=10.17.17.222 to
239.0.0.128, on v17 (phy 2/1/17), igmp_size=8
nIGMP: rcvd Report-V2(t=22)  g=239.0.0.129 resp=100, pkt S=10.17.17.222 to
239.0.0.129, on v17 (phy 2/1/17), igmp_size=8
oIGMP: rcvd Report-V2(t=22)  g=239.0.0.130 resp=100, pkt S=10.17.17.222 to
239.0.0.130, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.131 resp=100, pkt S=10.17.17.222 to
239.0.0.131, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.132 resp=100, pkt S=10.17.17.222 to
239.0.0.132, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.133 resp=100, pkt S=10.17.17.222 to
239.0.0.133, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.134 resp=100, pkt S=10.17.17.222 to
239.0.0.134, on v17 (phy 2/1/17), igmp_size=8
dIGMP: rcvd Report-V2(t=22)  g=239.0.0.135 resp=100, pkt S=10.17.17.222 to
239.0.0.135, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.136 resp=100, pkt S=10.17.17.222 to
239.0.0.136, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.137 resp=100, pkt S=10.17.17.222 to
239.0.0.137, on v17 (phy 2/1/17), igmp_size=8
eIGMP: rcvd Report-V2(t=22)  g=239.0.0.138 resp=100, pkt S=10.17.17.222 to
239.0.0.138, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.139 resp=100, pkt S=10.17.17.222 to
239.0.0.139, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.140 resp=100, pkt S=10.17.17.222 to
239.0.0.140, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.141 resp=100, pkt S=10.17.17.222 to
239.0.0.141, on v17 (phy 2/1/17), igmp_size=8
bIGMP: rcvd Report-V2(t=22)  g=239.0.0.142 resp=100, pkt S=10.17.17.222 to
239.0.0.142, on v17 (phy 2/1/17), igmp_size=8
uIGMP: rcvd Report-V2(t=22)  g=239.0.0.143 resp=100, pkt S=10.17.17.222 to
239.0.0.143, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.144 resp=100, pkt S=10.17.17.222 to
239.0.0.144, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.145 resp=100, pkt S=10.17.17.222 to
239.0.0.145, on v17 (phy 2/1/17), igmp_size=8
gIGMP: rcvd Report-V2(t=22)  g=239.0.0.146 resp=100, pkt S=10.17.17.222 to
239.0.0.146, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.147 resp=100, pkt S=10.17.17.222 to
239.0.0.147, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.148 resp=100, pkt S=10.17.17.222 to
239.0.0.148, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.149 resp=100, pkt S=10.17.17.222 to
239.0.0.149, on v17 (phy 2/1/17), igmp_size=8
aIGMP: rcvd Report-V2(t=22)  g=239.0.0.150 resp=100, pkt S=10.17.17.222 to
239.0.0.150, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.151 resp=100, pkt S=10.17.17.222 to
239.0.0.151, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.152 resp=100, pkt S=10.17.17.222 to
239.0.0.152, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.153 resp=100, pkt S=10.17.17.222 to
239.0.0.153, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.154 resp=100, pkt S=10.17.17.222 to
239.0.0.154, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.155 resp=100, pkt S=10.17.17.222 to
239.0.0.155, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.156 resp=100, pkt S=10.17.17.222 to
239.0.0.156, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.157 resp=100, pkt S=10.17.17.222 to
239.0.0.157, on v17 (phy 2/1/17), igmp_size=8

Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Turn off stack trace of write table
Disable shadow register write will disable all writes from shadow to hardware
```

debug ip igmp add-del-oif stack

Syntax: [no] debug ip igmp add-del-oif stack

This command monitors and displays instances of multicast cache activity, such as OIF additions or deletions, and generates a stack trace of the add or delete event.

```
Brocade# debug ip igmp add-del-oif stack
add-del-oif stack enabled
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp add-del-oif stack is enabled
Brocade# debug ip ig
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.1.8 resp=100, pkt S=10.17.17.222 to
239.0.1.8, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.9 resp=100, pkt S=10.17.17.222 to 239.0.1.9,
on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.10 resp=100, pkt S=10.17.17.222 to
239.0.1.10, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.11 resp=100, pkt S=10.17.17.222 to
239.0.1.11, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.12 resp=100, pkt S=10.17.17.222 to
239.0.1.12, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.13 resp=100, pkt S=10.17.17.222 to
239.0.1.13, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.14 resp=100, pkt S=10.17.17.222 to
239.0.1.14, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.15 resp=100, pkt S=10.17.17.222 to
239.0.1.15, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.16 resp=100, pkt S=10.17.17.222 to
239.0.1.16, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.17 resp=100, pkt S=10.17.17.222 to
239.0.1.17, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.18 resp=100, pkt S=10.17.17.222 to
239.0.1.18, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.19 resp=100, pkt S=10.17.17.222 to
239.0.1.19, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.20 resp=100, pkt S=10.17.17.222 to
239.0.1.20, on v17 (phy 2/1/17), igmp_size=8
nIGMP: rcvd Report-V2(t=22)  g=239.0.1.21 resp=100, pkt S=10.17.17.222 to
239.0.1.21, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.22 resp=100, pkt S=10.17.17.222 to
239.0.1.22, on v17 (phy 2/1/17), igmp_size=8
```

```
oIGMP: rcvd Report-V2(t=22)  g=239.0.1.23 resp=100, pkt S=10.17.17.222 to
239.0.1.23, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.24 resp=100, pkt S=10.17.17.222 to
239.0.1.24, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.25 resp=100, pkt S=10.17.17.222 to
239.0.1.25, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.26 resp=100, pkt S=10.17.17.222 to
239.0.1.26, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.27 resp=100, pkt S=10.17.17.222 to
239.0.1.27, on v17 (phy 2/1/17), igmp_size=8
dIGMP: rcvd Report-V2(t=22)  g=239.0.1.28 resp=100, pkt S=10.17.17.222 to
239.0.1.28, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.29 resp=100, pkt S=10.17.17.222 to
239.0.1.29, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.30 resp=100, pkt S=10.17.17.222 to
239.0.1.30, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.31 resp=100, pkt S=10.17.17.222 to
239.0.1.31, on v17 (phy 2/1/17), igmp_size=8
eIGMP: rcvd Report-V2(t=22)  g=239.0.1.32 resp=100, pkt S=10.17.17.222 to
239.0.1.32, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.33 resp=100, pkt S=10.17.17.222 to
239.0.1.33, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.34 resp=100, pkt S=10.17.17.222 to
239.0.1.34, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.35 resp=100, pkt S=10.17.17.222 to
239.0.1.35, on v17 (phy 2/1/17), igmp_size=8
bIGMP: rcvd Report-V2(t=22)  g=239.0.1.36 resp=100, pkt S=10.17.17.222 to
239.0.1.36, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.37 resp=100, pkt S=10.17.17.222 to
239.0.1.37, on v17 (phy 2/1/17), igmp_size=8
uIGMP: rcvd Report-V2(t=22)  g=239.0.1.38 resp=100, pkt S=10.17.17.222 to
239.0.1.38, on v17 (phy 2/1/17), igmp_size=8
gIGMP: rcvd Report-V2(t=22)  g=239.0.1.39 resp=100, pkt S=10.17.17.222 to
239.0.1.39, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.40 resp=100, pkt S=10.17.17.222 to
239.0.1.40, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.41 resp=100, pkt S=10.17.17.222 to
239.0.1.41, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.42 resp=100, pkt S=10.17.17.222 to
239.0.1.42, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.43 resp=100, pkt S=10.17.17.222 to
239.0.1.43, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.44 resp=100, pkt S=10.17.17.222 to
239.0.1.44, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.45 resp=100, pkt S=10.17.17.222 to
239.0.1.45, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.46 resp=100, pkt S=10.17.17.222 to
239.0.1.46, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.47 resp=100, pkt S=10.17.17.222 to
239.0.1.47, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.48 resp=100, pkt S=10.17.17.222 to
239.0.1.48, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.49 resp=100, pkt S=10.17.17.222 to
239.0.1.49, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.50 resp=100, pkt S=10.17.17.222 to
239.0.1.50, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.51 resp=100, pkt S=10.17.17.222 to
239.0.1.51, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.52 resp=100, pkt S=10.17.17.222 to
239.0.1.52, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.1.53 resp=100, pkt S=10.17.17.222 to
239.0.1.53, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.54 resp=100, pkt S=10.17.17.222 to
239.0.1.54, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.55 resp=100, pkt S=10.17.17.222 to
239.0.1.55, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.56 resp=100, pkt S=10.17.17.222 to
239.0.1.56, on v17 (phy 2/1/17), igmp_size=8
aIGMP: rcvd Report-V2(t=22)  g=239.0.1.57 resp=100, pkt S=10.17.17.222 to
239.0.1.57, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.58 resp=100, pkt S=10.17.17.222 to
239.0.1.58, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.1.59 resp=100, pkt S=10.17.17.222 to
239.0.1.59, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.60 resp=100, pkt S=10.17.17.222 to
239.0.1.60, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.61 resp=100, pkt S=10.17.17.222 to
239.0.1.61, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.1.62 resp=100, pkt S=10.17.17.222 to
239.0.1.62, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.63 resp=100, pkt S=10.17.17.222 to
239.0.1.63, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.64 resp=100, pkt S=10.17.17.222 to
239.0.1.64, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.65 resp=100, pkt S=10.17.17.222 to
239.0.1.65, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.66 resp=100, pkt S=10.17.17.222 to
239.0.1.66, on v17 (phy 2/1/17), igmp_size=8

Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Turn off stack trace of write table
Disable shadow register write will disable all writes from shadow to hardware
```

debug ip igmp clear

Syntax: [no] **debug ip igmp clear**

This command clears all the IGMP debug settings.

```
Brocade# debug ip igmp clear
no debug ip igmp is enabled
```

debug ip igmp down-port

Syntax: [no] **debug ip igmp down-port**

This command monitors the port that is down.

```
Brocade# debug ip igmp down-port
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp port-down is enabled
Brocade# debug ip igmp
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.84 resp=100, pkt S=10.17.17.222 to
239.0.0.84, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.85 resp=100, pkt S=10.17.17.222 to
239.0.0.85, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.86 resp=100, pkt S=10.17.17.222 to
239.0.0.86, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.87 resp=100, pkt S=10.17.17.222 to
239.0.0.87, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.88 resp=100, pkt S=10.17.17.222 to
239.0.0.88, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.89 resp=100, pkt S=10.17.17.222 to
239.0.0.89, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.90 resp=100, pkt S=10.17.17.222 to
239.0.0.90, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.91 resp=100, pkt S=10.17.17.222 to
239.0.0.91, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.92 resp=100, pkt S=10.17.17.222 to
239.0.0.92, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.93 resp=100, pkt S=10.17.17.222 to
239.0.0.93, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.94 resp=100, pkt S=10.17.17.222 to
239.0.0.94, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.95 resp=100, pkt S=10.17.17.222 to
239.0.0.95, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.96 resp=100, pkt S=10.17.17.222 to
239.0.0.96, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.97 resp=100, pkt S=10.17.17.222 to
239.0.0.97, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.98 resp=100, pkt S=10.17.17.222 to
239.0.0.98, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.99 resp=100, pkt S=10.17.17.222 to
239.0.0.99, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.100 resp=100, pkt S=10.17.17.222 to
239.0.0.100, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.101 resp=100, pkt S=10.17.17.222 to
239.0.0.101, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.102 resp=100, pkt S=10.17.17.222 to
239.0.0.102, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.103 resp=100, pkt S=10.17.17.222 to
239.0.0.103, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.104 resp=100, pkt S=10.17.17.222 to
239.0.0.104, on v17 (phy 2/1/17), igmp_size=8
nIGMP: rcvd Report-V2(t=22)  g=239.0.0.105 resp=100, pkt S=10.17.17.222 to
239.0.0.105, on v17 (phy 2/1/17), igmp_size=8
oIGMP: rcvd Report-V2(t=22)  g=239.0.0.106 resp=100, pkt S=10.17.17.222 to
239.0.0.106, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.107 resp=100, pkt S=10.17.17.222 to
239.0.0.107, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.108 resp=100, pkt S=10.17.17.222 to
239.0.0.108, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.109 resp=100, pkt S=10.17.17.222 to
239.0.0.109, on v17 (phy 2/1/17), igmp_size=8
dIGMP: rcvd Report-V2(t=22)  g=239.0.0.110 resp=100, pkt S=10.17.17.222 to
239.0.0.110, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.111 resp=100, pkt S=10.17.17.222 to
239.0.0.111, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.112 resp=100, pkt S=10.17.17.222 to
239.0.0.112, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.113 resp=100, pkt S=10.17.17.222 to
239.0.0.113, on v17 (phy 2/1/17), igmp_size=8
eIGMP: rcvd Report-V2(t=22)  g=239.0.0.114 resp=100, pkt S=10.17.17.222 to
239.0.0.114, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.115 resp=100, pkt S=10.17.17.222 to
239.0.0.115, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.116 resp=100, pkt S=10.17.17.222 to
239.0.0.116, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.117 resp=100, pkt S=10.17.17.222 to
239.0.0.117, on v17 (phy 2/1/17), igmp_size=8
bIGMP: rcvd Report-V2(t=22)  g=239.0.0.118 resp=100, pkt S=10.17.17.222 to
239.0.0.118, on v17 (phy 2/1/17), igmp_size=8
uIGMP: rcvd Report-V2(t=22)  g=239.0.0.119 resp=100, pkt S=10.17.17.222 to
239.0.0.119, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.120 resp=100, pkt S=10.17.17.222 to
239.0.0.120, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.121 resp=100, pkt S=10.17.17.222 to
239.0.0.121, on v17 (phy 2/1/17), igmp_size=8
g IGMP: rcvd Report-V2(t=22)  g=239.0.0.122 resp=100, pkt S=10.17.17.222 to
239.0.0.122, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.123 resp=100, pkt S=10.17.17.222 to
239.0.0.123, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.124 resp=100, pkt S=10.17.17.222 to
239.0.0.124, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.125 resp=100, pkt S=10.17.17.222 to
239.0.0.125, on v17 (phy 2/1/17), igmp_size=8
aIGMP: rcvd Report-V2(t=22)  g=239.0.0.126 resp=100, pkt S=10.17.17.222 to
239.0.0.126, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.127 resp=100, pkt S=10.17.17.222 to
239.0.0.127, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.128 resp=100, pkt S=10.17.17.222 to
239.0.0.128, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.129 resp=100, pkt S=10.17.17.222 to
239.0.0.129, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.130 resp=100, pkt S=10.17.17.222 to
239.0.0.130, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.131 resp=100, pkt S=10.17.17.222 to
239.0.0.131, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.132 resp=100, pkt S=10.17.17.222 to
239.0.0.132, on v17 (phy 2/1/17), igmp_size=8
Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Turn off stack trace of write table
Disable shadow register write will disable all writes from shadow to hardware
```

debug ip igmp enable

Syntax: [no] **debug ip igmp enable**

This command enables the IGMP debugging.

```
Brocade# debug ip igmp enable
debug ip igmp is enabled
```

debug ip igmp error

Syntax: [no] **debug ip igmp error**

This command displays the IGMP multicast error messages.

```
Brocade# debug ip igmp error
IGMP.VRF3.ERR: Rx packet ttl 255 not 1. Dropping packet
IGMP.VRF3.ERR: Rx packet ttl 255 not 1. Dropping packet
IGMP.VRF3.ERR: Rx packet ttl 255 not 1. Dropping packet
```

debug ip igmp group

Syntax: [no] **debug ip igmp group** *ipaddr*

This command matches the IGMP-enabled group based on the IP address. The *ipaddr* variable refers to the IP address of the IGMP group.

```
Brocade# debug ip igmp group 10.11.66.62
IGMP: No L3 mcast, pkt S=10.11.66.62 to 225.1.1.167, on v8 (phy 1/2/4),
igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=225.1.1.167 resp=0, pkt S=10.11.66.62 to
225.1.1.167, on VL8 (phy 1/2/4), igmp_size=8
```

debug ip igmp level

Syntax: [no] **debug ip igmp level** *decimal*

This command sets the debug level of the IGMP. The *decimal* variable specifies the level of the IGMP. The valid values are from 1 through 3.

```
Brocade# debug ip igmp level  1
debug level = 1
Brocade# debug ip ig show
igmp debug-enable-any = 1
debug ip igmp level 1 is enabled
Brocade# debug ip igmp
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.1.240 resp=100, pkt S=10.17.17.222 to
239.0.1.240, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.241 resp=100, pkt S=10.17.17.222 to
239.0.1.241, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.242 resp=100, pkt S=10.17.17.222 to
239.0.1.242, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.243 resp=100, pkt S=10.17.17.222 to
239.0.1.243, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.244 resp=100, pkt S=10.17.17.222 to
239.0.1.244, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.245 resp=100, pkt S=10.17.17.222 to
239.0.1.245, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.246 resp=100, pkt S=10.17.17.222 to
239.0.1.246, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.247 resp=100, pkt S=10.17.17.222 to
239.0.1.247, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.1.248 resp=100, pkt S=10.17.17.222 to
239.0.1.248, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.5 resp=100, pkt S=10.17.17.222 to 239.0.0.5,
on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.6 resp=100, pkt S=10.17.17.222 to 239.0.0.6,
on v17 (phy 2/1/17), igmp_size=8
    consume pkt
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.7 resp=100, pkt S=10.17.17.222 to 239.0.0.7,
on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.8 resp=100, pkt S=10.17.17.222 to 239.0.0.8,
on v17 (phy 2/1/17), igmp_size=8
    consume pkt
nIGMP: rcvd Report-V2(t=22)  g=239.0.0.9 resp=100, pkt S=10.17.17.222 to
239.0.0.9, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
oIGMP: rcvd Report-V2(t=22)  g=239.0.0.10 resp=100, pkt S=10.17.17.222 to
239.0.0.10, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.11 resp=100, pkt S=10.17.17.222 to
239.0.0.11, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.12 resp=100, pkt S=10.17.17.222 to
239.0.0.12, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.13 resp=100, pkt S=10.17.17.222 to
239.0.0.13, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.14 resp=100, pkt S=10.17.17.222 to
239.0.0.14, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.15 resp=100, pkt S=10.17.17.222 to
239.0.0.15, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.16 resp=100, pkt S=10.17.17.222 to
239.0.0.16, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.17 resp=100, pkt S=10.17.17.222 to
239.0.0.17, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.18 resp=100, pkt S=10.17.17.222 to
239.0.0.18, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.19 resp=100, pkt S=10.17.17.222 to
239.0.0.19, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.20 resp=100, pkt S=10.17.17.222 to
239.0.0.20, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.21 resp=100, pkt S=10.17.17.222 to
239.0.0.21, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.22 resp=100, pkt S=10.17.17.222 to
239.0.0.22, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.23 resp=100, pkt S=10.17.17.222 to
239.0.0.23, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.24 resp=100, pkt S=10.17.17.222 to
239.0.0.24, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.25 resp=100, pkt S=10.17.17.222 to
239.0.0.25, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.26 resp=100, pkt S=10.17.17.222 to
239.0.0.26, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.27 resp=100, pkt S=10.17.17.222 to
239.0.0.27, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.28 resp=100, pkt S=10.17.17.222 to
239.0.0.28, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.29 resp=100, pkt S=10.17.17.222 to
239.0.0.29, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
dIGMP: rcvd Report-V2(t=22)  g=239.0.0.30 resp=100, pkt S=10.17.17.222 to
239.0.0.30, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.31 resp=100, pkt S=10.17.17.222 to
239.0.0.31, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.32 resp=100, pkt S=10.17.17.222 to
239.0.0.32, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
eIGMP: rcvd Report-V2(t=22)  g=239.0.0.33 resp=100, pkt S=10.17.17.222 to
239.0.0.33, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.34 resp=100, pkt S=10.17.17.222 to
239.0.0.34, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.35 resp=100, pkt S=10.17.17.222 to
239.0.0.35, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.36 resp=100, pkt S=10.17.17.222 to
239.0.0.36, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
bIGMP: rcvd Report-V2(t=22)  g=239.0.0.37 resp=100, pkt S=10.17.17.222 to
239.0.0.37, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.38 resp=100, pkt S=10.17.17.222 to
239.0.0.38, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
uIGMP: rcvd Report-V2(t=22)  g=239.0.0.39 resp=100, pkt S=10.17.17.222 to
239.0.0.39, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.40 resp=100, pkt S=10.17.17.222 to
239.0.0.40, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
gIGMP: rcvd Report-V2(t=22)  g=239.0.0.41 resp=100, pkt S=10.17.17.222 to
239.0.0.41, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.42 resp=100, pkt S=10.17.17.222 to
239.0.0.42, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.43 resp=100, pkt S=10.17.17.222 to
239.0.0.43, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.44 resp=100, pkt S=10.17.17.222 to
239.0.0.44, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.45 resp=100, pkt S=10.17.17.222 to
239.0.0.45, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
aIGMP: rcvd Report-V2(t=22)  g=239.0.0.46 resp=100, pkt S=10.17.17.222 to
239.0.0.46, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.0.47 resp=100, pkt S=10.17.17.222 to
239.0.0.47, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
lIGMP: rcvd Report-V2(t=22)  g=239.0.0.48 resp=100, pkt S=10.17.17.222 to
239.0.0.48, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.49 resp=100, pkt S=10.17.17.222 to
239.0.0.49, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
lIGMP send Query(t=17) V2, s=10.17.17.52 0.0.0.0 to v17(all) rsp=100 igmp=8B,
pkt=46B
IGMP send Query(t=17) V2, s=19.19.19.52 0.0.0.0 to v19(all) rsp=100 igmp=8B,
pkt=46B
IGMP: rcvd Report-V2(t=22)  g=239.0.0.50 resp=100, pkt S=10.17.17.222 to
239.0.0.50, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.51 resp=100, pkt S=10.17.17.222 to
239.0.0.51, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.52 resp=100, pkt S=10.17.17.222 to
239.0.0.52, on v17 (phy 2/1/17), igmp_size=8
    consume pkt
IGMP: rcvd Report-V2(t=22)  g=239.0.0.53 resp=100, pkt S=10.17.17.222 to
239.0.0.53, on v17 (phy 2/1/17), igmp_size=8
    consume pkt

Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Turn off stack trace of write table
Disable shadow register write will disable all writes from shadow to hardware
```

debug ip igmp packet

Syntax: [no] **debug ip igmp packet** *ipaddr*

This command traces the IGMP packets. The *ipaddr* variable specifies the IP address of the IGMP packet.

```
Brocade# debug ip igmp packet 10.17.17.222 239.0.0.155
debug ip igmp packet S=10.17.17.222 G=239.0.0.155
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp packet 10.17.17.222 239.0.0.155 is enabled
Brocade# debug ip igmp
Brocade#IGMP: rcvd Report-V2(t=22)  g=239.0.1.160 resp=100, pkt S=10.17.17.222 to
239.0.1.160, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.161 resp=100, pkt S=10.17.17.222 to
239.0.1.161, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.162 resp=100, pkt S=10.17.17.222 to
239.0.1.162, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.163 resp=100, pkt S=10.17.17.222 to
239.0.1.163, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.164 resp=100, pkt S=10.17.17.222 to
239.0.1.164, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.165 resp=100, pkt S=10.17.17.222 to
239.0.1.165, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.166 resp=100, pkt S=10.17.17.222 to
239.0.1.166, on v17 (phy 2/1/17), igmp_size=8
```

```
IGMP: rcvd Report-V2(t=22)  g=239.0.1.167 resp=100, pkt S=10.17.17.222 to
239.0.1.167, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.168 resp=100, pkt S=10.17.17.222 to
239.0.1.168, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.169 resp=100, pkt S=10.17.17.222 to
239.0.1.169, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.170 resp=100, pkt S=10.17.17.222 to
239.0.1.170, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.171 resp=100, pkt S=10.17.17.222 to
239.0.1.171, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.172 resp=100, pkt S=10.17.17.222 to
239.0.1.172, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.173 resp=100, pkt S=10.17.17.222 to
239.0.1.173, on v17 (phy 2/1/17), igmp_size=8

Brocade#IGMP: rcvd Report-V2(t=22)  g=239.0.1.174 resp=100, pkt S=10.17.17.222 to
239.0.1.174, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.175 resp=100, pkt S=10.17.17.222 to
239.0.1.175, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.176 resp=100, pkt S=10.17.17.222 to
239.0.1.176, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.177 resp=100, pkt S=10.17.17.222 to
239.0.1.177, on v17 (phy 2/1/17), igmp_size=8

Brocade#IGMP: rcvd Report-V2(t=22)  g=239.0.1.178 resp=100, pkt S=10.17.17.222 to
239.0.1.178, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.179 resp=100, pkt S=10.17.17.222 to
239.0.1.179, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.180 resp=100, pkt S=10.17.17.222 to
239.0.1.180, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.181 resp=100, pkt S=10.17.17.222 to
239.0.1.181, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.182 resp=100, pkt S=10.17.17.222 to
239.0.1.182, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.183 resp=100, pkt S=10.17.17.222 to
239.0.1.183, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.184 resp=100, pkt S=10.17.17.222 to
239.0.1.184, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.185 resp=100, pkt S=10.17.17.222 to
239.0.1.185, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.186 resp=100, pkt S=10.17.17.222 to
239.0.1.186, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.187 resp=100, pkt S=10.17.17.222 to
239.0.1.187, on v17 (phy 2/1/17), igmp_size=8
nIGMP: rcvd Report-V2(t=22)  g=239.0.1.188 resp=100, pkt S=10.17.17.222 to
239.0.1.188, on v17 (phy 2/1/17), igmp_size=8
oIGMP: rcvd Report-V2(t=22)  g=239.0.1.189 resp=100, pkt S=10.17.17.222 to
239.0.1.189, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.190 resp=100, pkt S=10.17.17.222 to
239.0.1.190, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.191 resp=100, pkt S=10.17.17.222 to
239.0.1.191, on v17 (phy 2/1/17), igmp_size=8
dIGMP: rcvd Report-V2(t=22)  g=239.0.1.192 resp=100, pkt S=10.17.17.222 to
239.0.1.192, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.193 resp=100, pkt S=10.17.17.222 to
239.0.1.193, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.194 resp=100, pkt S=10.17.17.222 to
239.0.1.194, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.195 resp=100, pkt S=10.17.17.222 to
239.0.1.195, on v17 (phy 2/1/17), igmp_size=8
```

```
eIGMP: rcvd Report-V2(t=22)  g=239.0.1.196 resp=100, pkt S=10.17.17.222 to
239.0.1.196, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.197 resp=100, pkt S=10.17.17.222 to
239.0.1.197, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.198 resp=100, pkt S=10.17.17.222 to
239.0.1.198, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.199 resp=100, pkt S=10.17.17.222 to
239.0.1.199, on v17 (phy 2/1/17), igmp_size=8
buIGMP: rcvd Report-V2(t=22)  g=239.0.1.200 resp=100, pkt S=10.17.17.222 to
239.0.1.200, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.201 resp=100, pkt S=10.17.17.222 to
239.0.1.201, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.202 resp=100, pkt S=10.17.17.222 to
239.0.1.202, on v17 (phy 2/1/17), igmp_size=8
gIGMP: rcvd Report-V2(t=22)  g=239.0.1.203 resp=100, pkt S=10.17.17.222 to
239.0.1.203, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.204 resp=100, pkt S=10.17.17.222 to
239.0.1.204, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.205 resp=100, pkt S=10.17.17.222 to
239.0.1.205, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.206 resp=100, pkt S=10.17.17.222 to
239.0.1.206, on v17 (phy 2/1/17), igmp_size=8
aIGMP: rcvd Report-V2(t=22)  g=239.0.1.207 resp=100, pkt S=10.17.17.222 to
239.0.1.207, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.1.208 resp=100, pkt S=10.17.17.222 to
239.0.1.208, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.209 resp=100, pkt S=10.17.17.222 to
239.0.1.209, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.210 resp=100, pkt S=10.17.17.222 to
239.0.1.210, on v17 (phy 2/1/17), igmp_size=8
lIGMP: rcvd Report-V2(t=22)  g=239.0.1.211 resp=100, pkt S=10.17.17.222 to
239.0.1.211, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.212 resp=100, pkt S=10.17.17.222 to
239.0.1.212, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.213 resp=100, pkt S=10.17.17.222 to
239.0.1.213, on v17 (phy 2/1/17), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.1.214 resp=100, pkt S=10.17.17.222 to
239.0.1.214, on v17 (phy 2/1/17), igmp_size=8

Debug message destination:  default (console)
All possible debuggings have been turned off
tracking is off and all results are cleared
Turn off stack trace of write table
Disable shadow register write will disable all writes from shadow to hardware
```

debug ip igmp physical-port ethernet

Syntax: [no] **debug ip igmp physical-port ethernet** *stackid/slot/port*

This command matches the physical port that is connected. The *stackid/slot/port* variable refers to the stack ID, slot number, and port number of a specific Ethernet port.

```
Brocade# debug ip igmp physical-port ethernet 1/2/4
IGMP: rcvd Leave(t=23)  g=225.1.1.31 resp=0, pkt S=10.11.66.62 to 224.0.0.2, on
VL8 (phy 1/2/4), igmp_size=8
IGMP: No L3 mcast, pkt S=10.11.66.62 to 224.0.0.2, on v8 (phy 1/2/4), igmp_size=8
IGMP: rcvd Leave(t=23)  g=225.1.1.32 resp=0, pkt S=10.11.66.62 to 224.0.0.2, on
VL8 (phy 1/2/4), igmp_size=8
IGMP: No L3 mcast, pkt S=10.11.66.62 to 224.0.0.2, on v8 (phy 1/2/4), igmp_size=8
```

```
IGMP: rcvd Leave(t=23)  g=225.1.1.33 resp=0, pkt S=10.11.66.62 to 224.0.0.2, on
VL8 (phy 1/2/4), igmp_size=8
```

debug ip igmp prime-port ethernet

Syntax: [no] **debug ip igmp prime-port ethernet** *stackid/slot/port*

This command monitors the activity of the physical port if the trunking is carried out on the primary port. The <*stackid/slot/port*> variable refers to the stack ID, slot number, and port number of a specific Ethernet port.

```
Brocade# debug ip igmp prime-port ethernet 1/1/8
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp prime-port 1/1/8 is enabled
Brocade#
Brocade# debug ip igmp
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.5 resp=100, pkt S=10.16.16.222 to
239.0.0.5, on v16 (phy 1/1/8), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=239.0.0.5 resp=100, pkt S=10.16.16.222 to 239.0.0.5,
on v16 (phy 1/1/8), igmp_size=8
```

debug ip igmp protocol

Syntax: [no] **debug ip igmp protocol** [**query** | **report**]

- **query** - Displays debugging information about the IGMP queries transmitted and received.
- **report** - Displays debugging information about the IGMP reports transmitted and received.

The following is the sample output from the **debug ip igmp protocol query** command.

```
Brocade# debug ip igmp protocol query
Brocade# IGMP.VRF0: [ Port 1/1/15,v74. Grp 0.0.0.0 ] Sent version 3 Query. size
12. Src 10.0.0.18
IGMP.VRF0: [ Port 1/1/15,v74 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 2/1/15,v75. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 2/1/15,v75 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 3/1/39,v76. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 3/1/39,v76 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 6/1/22,v77. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 6/1/22,v77 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 4/1/22,v78. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 4/1/22,v78 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 5/1/22,v79. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 5/1/22,v79 ] Sent General Query version 3 using src 10.0.0.18
```

The following is the truncated sample output from the **debug ip igmp protocol report** command.

```
Brocade# debug ip igmp protocol report
Brocade# IGMP.VRF0: [ Port 1/1/15,v74. Grp 0.0.0.0 ] Sent version 3 Query. size
12. Src 10.0.0.18
IGMP.VRF0: [ Port 1/1/15,v74 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 2/1/15,v75. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 2/1/15,v75 ] Sent General Query version 3 using src 10.0.0.18
```

```
IGMP.VRF0: [ Port 3/1/39,v76. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 3/1/39,v76 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 6/1/22,v77. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 6/1/22,v77 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 4/1/22,v78. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 4/1/22,v78 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: [ Port 5/1/22,v79. Grp 0.0.0.0 ] Sent version 3 Query. size 12. Src
10.0.0.18
IGMP.VRF0: [ Port 5/1/22,v79 ] Sent General Query version 3 using src 10.0.0.18
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.52
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.43
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.22
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.16
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.34
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.54
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.4
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.90
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.33
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.15
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.61
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.37
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.59
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.8
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.1
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.46
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.7
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.1
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.94
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.99
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.98
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.96
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.35
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.91
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.64
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.5
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.92
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.20
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.41
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.80
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.76
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.36
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.39
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.93
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.31
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.38
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.33
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.72
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.57
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.79
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.2
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.99
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.73
IGMP.VRF0: Type V2 Report Port 6/1/22,v77 Grp 228.0.0.59
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.40
IGMP.VRF0: Type V2 Report Port 5/1/22,v79 Grp 228.0.0.84
```

debug ip igmp rx

Syntax: [no] **debug ip igmp rx**

This command displays the IGMP packets received.

```
Brocade# debug ip igmp rx
Brocade# debug ip igmp protocol query
IGMP.VRF1: Query Port 1/1/3*1/1/14,v25 Grp 0.0.0.0
IGMP.QRY: [ Port 1/1/3*1/1/14,v25. Grp 0.0.0.0 ]. Processing version 2 query
IGMP.VRF1: Query Port 1/1/3*1/1/14,v25 Grp 0.0.0.0
IGMP.QRY: [ Port 1/1/3*1/1/14,v25. Grp 0.0.0.0 ]. Processing version 2 query
und allIGMP.VRF3: Query Port 1/1/3*1/1/14,v26 Grp 0.0.0.0
IGMP.QRY: [ Port 1/1/3*1/1/14,v26. Grp 0.0.0.0 ]. Processing version 2 query
```

debug ip igmp show

Syntax: debug ip igmp show

This command displays the IGMP debug settings. Do not use the **show debug** command for this purpose. The **show debug** command displays general debug information.

```
Brocade# debug ip igmp show
igmp debug-enable-any = 1
debug ip igmp port-down is enabled
debug ip igmp detail is enabled
debug ip igmp level 2 is enabled
```

debug ip igmp source

Syntax: [no] **debug ip igmp source** *ipaddr*

This command matches the source of the control packet or IGMP V3 source packets. The *ipaddr* variable specifies the IP address of the IGMP source.

```
Brocade# debug ip igmp source 16.16.16.222
Brocade# IGMP: rcvd Report-V2(t=22)  g=239.0.0.5 resp=100, pkt S=16.16.16.222 to
239.0.0.5, on v16 (phy 1/1/8), igmp_size=8
```

debug ip igmp timer

Syntax: [no] **debug ip igmp timer**

This command monitors the IGMP timer activity.

```
Brocade# debug ip igmp timer
monitor timer activity
```

debug ip igmp tx

Syntax: [no] **debug ip igmp tx**

This command displays the IGMP packets transmitted.

```
Brocade# debug ip igmp tx
Brocade# debug ip igmp protocol query
IGMP.VRF2: [ Port 1/1/3*1/1/14,v21. Grp 0.0.0.0 ] Sent version 2 Query. size 8.
Src 21.0.0.6
IGMP.VRF2: [ Port 1/1/3*1/1/14,v21 ] Sent General Query version 2 using src
21.0.0.6
```

```
IGMP.VRF0: [ Port 1/1/7-1/1/8,tn11. Grp 0.0.0.0 ] Sent version 2 Query. size 8.
Src 61.0.0.6
IGMP.VRF0: [ Port 1/1/7-1/1/8,tn11 ] Sent General Query version 2 using src
61.0.0.6
IGMP.VRF0: [ Port 1/1/1-1/1/2,v91. Grp 0.0.0.0 ] Sent version 2 Query. size 8. Src
91.0.0.6
IGMP.VRF0: [ Port 1/1/1-1/1/2,v91 ] Sent General Query version 2 using src
91.0.0.6
```

debug ip igmp vrf

Syntax: [no] **debug ip igmp vrf** *vrf_name*

This command displays IGMP related debug information for the specified VRF.

```
Brocade# debug ip igmp vrf vrf1
Debug: Jan  1 22:24:59 IGMP.VRF1: IGMP Leave, Type V2 Report Port 6/5,v65 Grp
229.2.1.1
Debug: Jan  1 22:25:05 IGMP.VRF1: Type V2 Report Port 6/5,v65 Grp 229.2.1.1
Debug: Jan  1 22:25:08 IGMP.VRF1: Type V2 Report Port 6/5,v65 Grp 229.2.1.1
```

debug ip igmp proxy clear

Syntax: debug ip igmp proxy clear

This command clears all the IGMP proxy debug settings.

debug ip igmp proxy error

Syntax: [no] **debug ip igmp proxy error**

This command displays the IGMP proxy error messages.

```
Brocade# debug ip igmp proxy error
Debug: Jan  2 03:26:25 1d 3:26:43 - MCPROXY_ERROR: ref count 1 for group entry
235.5.4.145
Debug: Jan  2 03:46:35 - MCPROXY_ERROR: ref count 1 for group entry 235.5.4.241
Debug: Jan  2 03:48:55 - MCPROXY_ERROR: ref count 1 for group entry 235.5.4.49
```

debug ip igmp proxy event

Syntax: [no] **debug ip igmp proxy event**

This command enables debugging of IGMP proxy events and the output resembles the following example.

```
Brocade# debug ip igmp proxy event
Debug: Jan  1 20:53:56 20:54:14 - MCPROXY_EVENT: [IPv4:default-vrf] receive group
229.2.1.1 state change
Debug: Jan  1 20:54:11 20:54:29 - MCPROXY_EVENT: [IPv4:default-vrf] receive group
229.2.1.1 state change
Debug: Jan  1 20:54:33 20:54:51 - MCPROXY_EVENT: [IPv4:default-vrf] receive group
229.2.1.1 state change
Debug: Jan  1 20:54:40 20:54:58 - MCPROXY_EVENT: [IPv4:default-vrf] receive group
229.2.1.1 state change
```

debug ip igmp proxy packet

Syntax: [no] **debug ip igmp proxy packet**

This command displays debugging information related to the IGMP proxy packets as shown in the following example.

```
Brocade# debug ip igmp proxy packet
Debug: Jan  1 20:55:28 20:55:46 - MCPROXY_TX: [IPv4:default-vrf] sending v1v2 grp
229.2.1.1 rprt to vir port v85 phy port 2/13
Debug: Jan  1 20:55:28 20:55:46 - MCPROXY_TX: [IPv4:default-vrf] receive group
query 229.2.1.1 on port e2/13 max resp code 10
Debug: Jan  1 20:55:29 20:55:47 - MCPROXY_TX: [IPv4:default-vrf] receive group
query 229.2.1.1 on port e2/13 max resp code 10
Debug: Jan  1 20:55:40 20:55:58 - MCPROXY_TX: [IPv4:default-vrf] sending v1v2 grp
229.2.1.1 rprt to vir port v85 phy port 2/13
Debug: Jan  1 20:55:42 20:55:59 - MCPROXY_TX: [IPv4:default-vrf] sending v1v2 grp
229.2.1.1 rprt to vir port v85 phy port 2/13
```

debug ip igmp proxy rx

Syntax: [no] **debug ip igmp proxy rx**

This command displays the IGMP proxy packets received as shown in the following example.

```
Brocade# debug ip igmp proxy rx
debug ip igmp proxy Debug: Jan  1 20:58:06 20:58:23 - MCPROXY_RX:
[IPv4:default-vrf] receive general query on port e2/13 max resp code 100
Debug: Jan  1 21:00:11 21:0:28 - MCPROXY_RX: [IPv4:default-vrf] receive general
query on port e2/13 max resp code 100
Debug: Jan  1 21:04:21 21:4:38 - MCPROXY_RX: [IPv4:default-vrf] receive general
query on port e2/13 max resp code 100
Debug: Jan  1 21:06:26 21:6:43 - MCPROXY_RX: [IPv4:default-vrf] receive general
query on port e2/13 max resp code 100
Debug: Jan  1 21:08:31 21:8:48 - MCPROXY_RX: [IPv4:default-vrf] receive general
query on port e2/13 max resp code 100
Debug: Jan  1 21:10:36 21:10:54 - MCPROXY_RX: [IPv4:default-vrf] receive general
query on port e2/13 max resp code 100
Debug: Jan  1 21:12:41 21:12:59 - MCPROXY_RX: [IPv4:default-vrf] receive general
query on port e2/13 max resp code 100
```

debug ip igmp proxy show

Syntax: [no] **debug ip igmp proxy show**

This command displays the IGMP proxy debug settings as shown in the following example.

```
Brocade# debug ip igmp proxy show
Mcast Proxy debug for IPv4:
 Receive debug is enable
 Transmit debug is enable
 Error debug is enable
```

debug ip igmp proxy tx

Syntax: [no] **debug ip igmp proxy tx**

This command displays the IGMP proxy packets transmitted as shown in the following example.

```
Brocade# debug ip igmp proxy tx
Debug: Jan  1 20:55:28 20:55:46 - MCPROXY_TX: [IPv4:default-vrf] receive group
query 229.2.1.1 on port e2/13 max resp code 10
Debug: Jan  1 20:55:29 20:55:47 - MCPROXY_TX: [IPv4:default-vrf] receive group
query 229.2.1.1 on port e2/13 max resp code 10
```

```
Debug: Jan  1 20:55:40 20:55:58 - MCPROXY_TX: [IPv4:default-vrf] sending v1v2 grp
229.2.1.1 rprt to vir port v85 phy port 2/13
Debug: Jan  1 20:55:42 20:55:59 - MCPROXY_TX: [IPv4:default-vrf] sending v1v2 grp
229.2.1.1 rprt to vir port v85 phy port 2/13
```

debug ip igmp proxy vrf

Syntax: [no] **debug ip igmp proxy vrf** *vrf_name*

This command displays IGMP proxy related debug information for the specified VRF shown in the following example.

```
Brocade# debug ip igmp proxy vrf vrf1
Debug: Jan  1 21:43:08 21:43:26 - MCPROXY_TX: [IPv4:1] sending v1v2 grp 229.2.1.1
rprt to vir port v85 phy port 2/13
Debug: Jan  1 21:43:08 21:43:26 - MCPROXY_TX: [IPv4:1] receive group query
229.2.1.1 on port e2/13 max resp code 10
Debug: Jan  1 21:43:09 21:43:27 - MCPROXY_TX: [IPv4:1] receive group query
229.2.1.1 on port e2/13 max resp code 10
Debug: Jan  1 21:43:22 21:43:39 - MCPROXY_TX: [IPv4:1] sending v1v2 grp 229.2.1.1
rprt to vir port v85 phy port 2/13
Debug: Jan  1 21:43:23 21:43:41 - MCPROXY_TX: [IPv4:1] sending v1v2 grp 229.2.1.1
rprt to vir port v85 phy port 2/13
```

IGMP snooping debug commands

The following commands display information related to the IGMP snoop debugging.

debug igmp-snoop add-del-oif all

Syntax: debug igmp-snoop add-del-oif all

This command displays information about the addition or deletion of all outgoing interfaces (OIFs). Command output resembles the following example.

```
Brocade# debug igmp-snoop add-del-oif all
Del e1/1/23  from (0.0.0.0, 228.1.1.10) vlan 130
Del e1/1/23  from (0.0.0.0, 228.1.1.9) vlan 130
Del e1/1/23  from (0.0.0.0, 228.1.1.8) vlan 130
Add e1/1/23  to (0.0.0.0, 228.1.1.8) vlan 130
Add e1/1/23  to (0.0.0.0, 228.1.1.9) vlan 130
Add e1/1/23  to (0.0.0.0, 228.1.1.10) vlan 130
```

debug igmp-snoop clear

Syntax: [no] **debug igmp-snoop clear**

This command clears all the IGMP snooping debug settings.

debug igmp-snoop error

Syntax: [no] **debug igmp-snoop error**

This command displays the IGMP snooping multicast error messages. Command output resembles the following example.

```
Brocade# debug igmp-snoop error
```

```
IGMP: Error : Ignoring IGMP version 3 Report pkt due to version mismatch , pkt
S=10.13.13.2 to 224.0.0.22, on VL130 (phy e1/1/2), igmp_size=16
IGMP: Error : Ignoring IGMP version 3 Report pkt due to version mismatch , pkt
S=10.13.13.2 to 224.0.0.22, on VL130 (phy e1/1/2), igmp_size=16
IGMP: Error : Ignoring IGMP version 3 Report pkt due to version mismatch , pkt
S=10.13.13.2 to 224.0.0.22, on VL130 (phy e1/1/2), igmp_size=16
```

debug igmp-snoop group

Syntax: [no] **debug igmp-snoop group** *group_address*

The *group_address* is the address of the IGMP snoop multicast group.

This command displays the IGMP snooping multicast group information. Command output resembles the following example.

```
Brocade# debug igmp-snoop group 230.1.1.1
IGMP: rcvd Report-V2(t=22)  g=230.1.1.1 resp=0, pkt S=10.85.1.3 to 230.1.1.1, on
VL100 (phy e1/1/48), igmp_size=8
```

debug igmp-snoop level

Syntax: [no] **debug igmp-snoop level** *decimal*

The *decimal variable* specifies the debugging level of IGMP snooping activity and it can take values from 1 through 3.

This command sets the debuging level of IGMP snooping activity. Level 3 displays more information than level 2, and level 0 displays no information. Command output resembles the following example.

```
Brocade# debug igmp-snoop level 1
igmpsnoop debug level = 1
IGMP: rcvd Query(t=17) V2 g=0.0.0.0 resp=100, pkt S=10.1.15.1 to 224.0.0.1, on
VL100 (phy e3/1/6), igmp_size=8
```

```
Brocade# debug igmp-snoop level 3
igmpsnoop debug level = 3
Brocade# Add e1/1/2  to (0.0.0.0, 228.228.0.1) vlan 275
Add e1/1/2  to (0.0.0.0, 228.228.0.2) vlan 275
Add e1/1/2  to (0.0.0.0, 228.228.0.3) vlan 275
Brocade# Del e1/1/2  from (0.0.0.0, 228.228.0.3) vlan 275
Del e1/1/2  from (0.0.0.0, 228.228.0.2) vlan 275
Del e1/1/2  from (0.0.0.0, 228.228.0.1) vlan 275
Del e1/1/12  from (0.0.0.0, 228.228.0.3) vlan 275
Del e1/1/2  from (0.0.0.0, 228.228.0.1) vlan 275
```

debug igmp-snoop packet

Syntax: [no] **debug igmp-snoop packet** *source_address group_address*

The *source_address specifies the source address* of the IGMP snoop packet.

The *group_address* is the address of the IGMP snoop multicast group.

This command displays debugging information related to the IGMP snoop packets. Command output resembles the following example.

```
Brocade# debug igmp-snoop packet 10.13.13.3 228.1.1.2
```

```
debug ip igmpsnoop  packet S=10.13.13.3 G=228.1.1.2
```

```
rtr_l2mcast_fwd_data (pkts:1), (10.13.13.3 228.1.1.2), from e1/1/1 vlan 130, has
fwd_entry
rtr_l2mcast_fwd_data (pkts:1), (10.13.13.3 228.1.1.2), from e1/1/1 vlan 130,
clear hw_aging flag
```

debug igmp-snoop port ethernet

Syntax: [no] **debug igmp-snoop port ethernet** *num*

The *num* variable refers to the stack ID, slot number, and port number of a specific Ethernet port.

This command displays the IGMP snooping related debug information for the specified Ethernet interface. Command output resembles the following example.

```
Brocade# debug igmp-snoop port ethernet 1/1/2
IGMP: rcvd Report-V2(t=22)  g=228.1.1.6 resp=0, pkt S=10.13.13.2 to 228.1.1.6, on
VL130 (phy e1/1/2), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=228.1.1.8 resp=0, pkt S=10.13.13.2 to 228.1.1.8, on
VL130 (phy e1/1/2), igmp_size=8
```

debug igmp-snoop show

Syntax: [no] **debug igmp-snoop show**

This command displays all the IGMP snooping debug settings. Command output resembles the following example.

```
Brocade# debug igmp-snoop show
igmp debug-enable-any = 1
debug igmpsnoop  is enabled
IGMP: rcvd Query(t=17) V2 g=0.0.0.0 resp=100, pkt S=10.1.15.1 to 224.0.0.1, on
VL100 (phy e3/1/6), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.1.12 resp=0, pkt S=10.85.1.3 to 230.1.1.12, on
VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.4.148 resp=0, pkt S=10.85.1.3 to 230.1.4.148,
on VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.2.22 resp=0, pkt S=10.85.1.3 to 230.1.2.22, on
VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.1.92 resp=0, pkt S=10.85.1.3 to 230.1.1.92, on
VL100 (phy e1/1/48), igmp_size=8
```

debug igmp-snoop source

Syntax: [no] **debug igmp-snoop source** *source_address*

This command displays the IGMP snooping related debug information for the specified source address. Command output resembles the following example.

```
Brocade# debug igmp-snoop source 10.13.13.2
IGMP: rcvd Report-V2(t=22)  g=228.1.1.6 resp=0, pkt S=10.13.13.2 to 228.1.1.6, on
VL130 (phy e1/1/2), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=228.1.1.8 resp=0, pkt S=10.13.13.2 to 228.1.1.8, on
VL130 (phy e1/1/2), igmp_size=8
```

debug igmp-snoop timer

Syntax: debug igmp-snoop timer

This command displays information about the IGMP snooping related timer events. Command output resembles the following example.

```
Brocade# debug igmp-snoop timer
Brocade# debug igmp-snoop prime-port eth 1/1/2
igmp timer, VL130(e1/1/2) change V1 client age out
igmp timer, E401010A VL130(e1/1/2) switch back to include mode
igmp timer, E4010109 VL130(e1/1/2) switch back to include mode
igmp timer, E4010108 VL130(e1/1/2) switch back to include mode
```

debug igmp-snoop vlan

Syntax: [no] **debug igmp-snoop vlan** *vlan_id*

This command displays the IGMP snooping related debug information for the specified VLAN ID. Command output resembles the following example.

```
Brocade# debug igmp-snoop vlan 100
igmpsnoop monitor vlan 100
IGMP: rcvd Report-V2(t=22)  g=230.1.1.110 resp=0, pkt S=10.85.1.3 to 230.1.1.110,
on VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.2.52 resp=0, pkt S=10.85.1.3 to 230.1.2.52, on
VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.3.133 resp=0, pkt S=10.85.1.3 to 230.1.3.133,
on VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.2.203 resp=0, pkt S=10.85.1.3 to 230.1.2.203,
on VL100 (phy e1/1/48), igmp_size=8
IGMP: rcvd Report-V2(t=22)  g=230.1.1.151 resp=0, pkt S=10.85.1.3 to 230.1.1.151,
on VL100 (phy e1/1/48), igmp_size=8
```

PIM debug commands

Protocol Independent Multicast (PIM) is used to simplify some of the complexity of the routing protocol at the cost of additional overhead tied with a greater replication of forwarded multicast packets. PIM is similar to DVMRP in that PIM builds source-routed multicast delivery trees and employs reverse path check when forwarding the multicast packets.

debug ip pim

Syntax: [no] **debug ip pim**

This command activates the PIM debugging.

```
Brocade# debug ip pim
PIM:  pim debugging is on
```

debug ip pim add-del-oif

Syntax: [no] **debug ip pim add-del-oif**

This command shows mcache activity and will work with the **debug ip pim group** or **debug ip pim source** commands.

```
Brocade# show ip pim neighbors
Total number of neighbors: 4 on 4 ports
Port   Phy_p      Neighbor        Holdtime Age   UpTime  GenID
v19    e16/11     10.19.19.52     180      0     58692   0x5EF61EDA
Port   Phy_p      Neighbor        Holdtime Age   UpTime  GenID
tn2    e16/27     10.220.220.21  180      0     10556   0x28C97FB3
Port   Phy_p      Neighbor        Holdtime Age   UpTime  GenID
tn9    e7/7       10.139.139.32  180      0     0       0x50B77D32
```

```
Port   Phy_p      Neighbor        Holdtime Age   UpTime  GenID
tn24   e7/1       10.120.120.31  180      60    840     0x15EB0B50
Brocade# debug ip pim add-del-oif
add-del-oif enabled
Brocade# debug ip pim show
pim debug-enable-any = 1
debug ip pim add-del-oif is enabled
Brocade# debug ip pim
          PIM:  pim debugging is on
Brocade# vport_state_notify v19 (phy e7/3) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
pim_dvmrp_process_port_down (ve port) v19
    update oif, del nbr, del prune_list, update fwd
    igmp_del_group, del_grp, igmp_del_g_grp, reset vif cnt
pim_process_port_up (ve port) v19, send query
vport_state_notify v19 (phy e7/3) up=1
vport_state_notify v19 (phy e16/12) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e16/11) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()

Brocade#vport_state_notify v19 (phy e7/3) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e7/4) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e7/3) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e7/3) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e7/3) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e16/11) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
pim_dvmrp_process_port_down (ve port) v19
    update oif, del nbr, del prune_list, update fwd
    igmp_del_group, del_grp, igmp_del_g_grp, reset vif cnt
pim_process_port_up (ve port) v19, send query
vport_state_notify v19 (phy e7/3) up=1

Brocade#vport_state_notify v19 (phy e7/3) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
vport_state_notify v19 (phy e16/12) up=0
    update_nbr_phy_mask_for_portdown()
    remove_vport_forward_entry()
pim_dvmrp_process_port_down (ve port) v19
    update oif, del nbr, del prune_list, update fwd
    igmp_del_group, del_grp, igmp_del_g_grp, reset vif cnt
pim_process_port_up (ve port) v19, send query
vport_state_notify v19 (phy e7/3) up=1
```

debug ip pim bootstrap

Syntax: [no] **debug ip pim bootstrap**

This command enables bootstrap debugging and the output resembles the following example.

```
Brocade# debug ip pim bootstrap
PIM:  bootstrap debugging is on

Brocade# PIM-BSR.VRF0: Prefer BSR 10.0.0.43(Pr 43) over current BSR 10.0.0.43(Pr
43)

PIM-BSR.VRF0: Intf tn4 - accept BSM from BSR 10.0.0.43(Pr 43), local state
CandBSR, curr BSR 10.0.0.43 (Pr 43)
```

debug ip pim clear

Syntax: [no] **debug ip pim clear**

This command clears all the PIM debug settings. The output will be similar to the following if no settings are enabled.

```
Brocade# debug ip pim clear
no debug ip pim is enabled
```

debug ip pim control-source

Syntax: [no] **debug ip pim control-source** *ipaddr*

This command monitors the control packet sent by the given router. The <*ipaddr*> variable refers to a particular route.

```
Brocade# debug ip pim control-source 10.10.10.129
Brocade# IGMP: rcvd Query(t=17) V2 g=0.0.0.0 resp=100, pkt S=10.10.10.129 to
224.0.0.1, on v100 (phy 8/1/22), igmp_size=8

Brocade# IGMP: rcvd Query(t=17) V2 g=0.0.0.0 resp=100, pkt S=10.10.10.129 to
224.0.0.1, on v100 (phy 8/1/22), igmp_size=8
```

debug ip pim event

Syntax: [no] **debug ip pim event**

This command enables debugging of PIM events and the output resembles the following example.

```
Brocade# debug ip pim event
PIM:  event debugging is on
Brocade(config)# int ethernet 1/1/11
Brocade(config-if-e1000-1/1/11)# disable
Brocade(config-if-e1000-1/1/11)# Mcastv4 receive event 16
Mcastv4 receive event 16
Mcastv4 receive event 16
Mcastv4 receive event 16
Mcastv4 receive event 13
MC-EVT: Port tn4 state changed to DOWN
PIM-EVT.VRF0: Receive intf tn4 Dn state_notify
Mcastv4 receive event 6
Brocade(config-if-e1000-1/1/11)# enable
Mcastv4 receive event 15
Mcastv4 receive event 15
Mcastv4 receive event 15
Mcastv4 receive event 15
```

debug ip pim group

Syntax: [no] **debug ip pim group** *ipaddr*

This command displays the PIM related activities related to the particular IP address of the PIM group.

```
Brocade# debug ip pim group 227.1.1.1
Brocade# Proc IGMP join g=227.1.1.1 from v300(2/1/17) 10.11.55.55
PIM: send prune v100, source 10.11.99.99 group 227.1.1.1 nbr 10.10.10.129
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 227.1.1.1
Prune list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=34 to v100
PIM: send prune v100, source 0.0.0.0 group 227.1.1.1 nbr 10.10.10.129
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 227.1.1.1
Prune list: (0.0.0.0 227.1.1.1) wc=0 rpt=0 sparse=1
Prune list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=42 to v100
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
  group address: 227.1.1.1
Prune list: (10.10.10.129 227.1.1.1) wc=1 rpt=1 sparse=1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=34 to v100
join_prune_timer, (10.11.99.99 227.1.1.1) num=0, RPT=0, group_on_dr_and_sg=0, PR
UNE
join_prune_timer, (10.11.99.99 227.1.1.1) ->num=0, rpt=0, spt=1, SRC=v100, RP=v1
00
join_prune_timer, (10.11.99.99 227.1.1.1) fail RPT PRUNE cond.
join_prune_timer, (0.0.0.0 227.1.1.1) rp not local, send join or prune, wc=1, rp
t=1
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 227.1.1.1
Prune list: (10.10.10.129 227.1.1.1) wc=1 rpt=1 sparse=1
Prune list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=406 to v100

Brocade# show ip pim group | i 227.1.1.1
Brocade# join_prune_timer, (10.11.99.99 227.1.1.1) num=0, RPT=0, group_on_
dr_and_sg=0, PRUNE
join_prune_timer, (10.11.99.99 227.1.1.1) ->num=0, rpt=0, spt=1, SRC=v100, RP=v1
00
join_prune_timer, (10.11.99.99 227.1.1.1) fail RPT PRUNE cond.
join_prune_timer, (0.0.0.0 227.1.1.1) rp not local, send join or prune, wc=1, rp
t=1
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 227.1.1.1
Prune list: (10.10.10.129 227.1.1.1) wc=1 rpt=1 sparse=1
Prune list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=406 to v100

Brocade# Proc IGMP join g=227.1.1.1 from v300(2/1/17) 10.11.55.55
join_prune_timer, (10.11.99.99 227.1.1.1) ->num=1, rpt=0, am_rp_and_proxy=0, JOI
N
join_prune_timer, (10.11.99.99 227.1.1.1) ->num=1, rpt=0, spt=1, SRC=v100, RP=v1
00
join_prune_timer, (10.11.99.99 227.1.1.1) fail RPT PRUNE cond.
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 227.1.1.1
    Join list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
    Join list: (10.10.10.129 227.1.1.1) wc=1 rpt=1 sparse=1
```

```
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=42 to v100
Brocade# Proc IGMP join g=227.1.1.1 from v300(2/1/17) 10.11.55.55

Brocade# join_prune_timer, (10.11.99.99 227.1.1.1) ->num=1, rpt=0,
am_rp_and_proxy=0, JOIN
join_prune_timer, (10.11.99.99 227.1.1.1) ->num=1, rpt=0, spt=1, SRC=v100, RP=v1
00
join_prune_timer, (10.11.99.99 227.1.1.1) fail RPT PRUNE cond.
join_prune_timer, (0.0.0.0 227.1.1.1) rp not local, send join or prune, wc=1, rp
t=1
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 227.1.1.1
    Join list: (10.10.10.129 227.1.1.1) wc=1 rpt=1 sparse=1
    Join list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=406 to v100
reset age (10.11.99.99, 227.1.1.1)

Brocade# show ip pim group | i 227.1.1.1
Index 7 Group 227.1.1.1
```

debug ip pim join-prune

Syntax: [no] **debug ip pim join-prune**

This command controls join prune processing and displays debugging information related to the join prune messages. The following is the truncated sample output from the **debug ip pim join-prune** command.

```
Brocade# debug ip pim join-prune
PIM:  join-prune debugging is on
Brocade#  PIM.VRF0: (10.0.0.178 232.0.0.6) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.6) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.7) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.7) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.8) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.8) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.9) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.9) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.10) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.10) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.11) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.11) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.12) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.12) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.13) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.13) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.14) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.14) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.15) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.15) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.16) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.16) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.17) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.17) RPFIf v76. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.178 232.0.0.18) RPFIf v75. RPFNbr -. Send Join/Prune
  PIM.VRF0: (10.0.0.168 232.0.0.18) RPFIf v76. RPFNbr -. Send Join/Prune
```

debug ip pim level

Syntax: [no] **debug ip pim level** *decimal*

This command sets the PIM level. The *decimal* variable refers to the number of the PIM level. The valid values are from 1 through 3.

```
Brocade# debug ip pim level
    DECIMAL 1-3 (most detailed)
Brocade# debug ip pim level 1
debug level = 1
Brocade# rcvd PIM_V2 Hello, from 10.10.10.129 to 224.0.0.13 Len 34 on v10
0 (8/1/22)
rcvd PIM_V2 Hello, from 10.10.10.129 to 224.0.0.13 Len 34 on v100 (8/1/22)

Brocade# debug ip pim level 2
debug level = 2
Brocade# 00:21:13 Send PIM_V2 Hello Src 10.10.10.119 Dst 224.0.0.13 Len 3
4 on v100
00:21:13 Send PIM_V2 Hello Src 10.1.20.22 Dst 224.0.0.13 Len 34 on v200
00:21:13 Send PIM_V2 Hello Src 10.1.30.22 Dst 224.0.0.13 Len 34 on v300
rcvd PIM_V2 Bootstrap, from 10.10.10.129 to 224.0.0.13 Len 36 on v100 (8/1/22)
00:21:16 Send PIM_V2 Hello Src 10.10.10.119 Dst 224.0.0.13 Len 34 on v100
00:21:16 Send PIM_V2 Hello Src 10.1.20.22 Dst 224.0.0.13 Len 34 on v200
00:21:16 Send PIM_V2 Hello Src 10.1.30.22 Dst 224.0.0.13 Len 34 on v300
PIMSM: BEGIN Periodic join-prune msgs
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.10.129,
group address: 225.1.1.10
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=42 to v100
group address: 225.1.1.9
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=70 to v100
group address: 225.1.1.8
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=98 to v100
group address: 225.1.1.7
send_J/P_msg(), lcl_adr=00.10.10.119 dest=224.0.0.13 pkt_size=126 to v100
group address: 225.1.1.6
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=154 to v100
group address: 225.1.1.5
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=182 to v100
group address: 225.1.1.4
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=210 to v100
group address: 226.1.1.1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=238 to v100
group address: 225.1.1.3
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=266 to v100
group address: 225.1.1.2
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=294 to v100
group address: 225.1.1.1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=322 to v100
group address: 226.1.1.2
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=350 to v100
group address: 227.1.1.2
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=378 to v100
group address: 227.1.1.1
send_J/P_msg(), lcl_adr=10.10.10.119 dest=224.0.0.13 pkt_size=406 to v100
00:21:16 Send PIM_V2 Join/Prune Src 10.10.10.119 Dst 224.0.0.13 Len 406 on v100
PIMSM: END Periodic join-prune msgs

Brocade# debug ip pim level 3
debug level = 3
Brocade# rcvd PIM_V2 Hello, from 10.10.10.129 to 224.0.0.13 Len 34 on v10
0 (8/1/22)
Hello msg v100, source 10.10.10.129 group 224.0.0.13
rcvd PIM_V2 Hello, from 10.10.10.129 to 224.0.0.13 Len 34 on v100 (8/1/22)
Hello msg v100, source 10.10.10.129 group 224.0.0.13
```

```
Proc IGMP join g=226.1.1.2 from v200(5/1/13) 10.11.66.66
Proc IGMP join g=225.1.1.1 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.2 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.3 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=226.1.1.1 from v200(8/1/17)10.11.22.22
Proc IGMP join g=225.1.1.4 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.5 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.6 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.7 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.8 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=227.1.1.1 from v300(2/1/17)10.11.55.55
Proc IGMP join g=225.1.1.9 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=225.1.1.10 from v200(5/1/9) 10.11.10.77
Proc IGMP join g=227.1.1.2 from v300(2/1/17) 10.11.55.55
00:22:13 Send PIM_V2 Hello Src 10.10.119 Dst 224.0.0.13 Len 34 on v100
00:22:13 Send PIM_V2 Hello Src 10.1.20.22 Dst 224.0.0.13 Len 34 on v200
00:22:13 Send PIM_V2 Hello Src 10.1.30.22 Dst 224.0.0.13 Len 34 on v300
rcvd PIM_V2 Bootstrap, from 10.10.129 to 224.0.0.13 Len 36 on v100 (8/1/22)
00:22:16 Send PIM_V2 Hello Src 10.10.119 Dst 224.0.0.13 Len 34 on v100
00:22:16 Send PIM_V2 Hello Src 10.1.20.22 Dst 224.0.0.13 Len 34 on v200
00:22:16 Send PIM_V2 Hello Src 10.1.30.22 Dst 224.0.0.13 Len 34 on v300
PIMSM: BEGIN Periodic join-prune msgs
Begin sending Join/Prune msg to v100, rpf_nbr=10.10.129,
  group address: 225.1.1.10
    Join list: (10.10.129 225.1.1.10) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.10) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=42 to v100
  group address: 225.1.1.9
    Join list: (10.10.129 225.1.1.9) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.9) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=70 to v100
  group address: 225.1.1.8
    Join list: (10.10.129 225.1.1.8) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.8) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=98 to v100
  group address: 225.1.1.7
    Join list: (10.10.129 225.1.1.7) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.7) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=126 to v100
  group address: 225.1.1.6
    Join list: (10.10.129 225.1.1.6) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.6) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=154 to v100
  group address: 225.1.1.5
    Join list: (10.10.129 225.1.1.5) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.5) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=182 to v100
  group address: 225.1.1.4
    Join list: (10.10.129 225.1.1.4) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.4) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=210 to v100
  group address: 226.1.1.1
    Join list: (10.10.129 226.1.1.1) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 226.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=238 to v100
  group address: 225.1.1.3
    Join list: (10.10.129 225.1.1.3) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.3) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=266 to v100
  group address: 225.1.1.2
```

```
    Join list: (10.10.129 225.1.1.2) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.2) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=294 to v100
  group address: 225.1.1.1
    Join list: (10.10.129 225.1.1.1) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 225.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=322 to v100
  group address: 226.1.1.2
    Join list: (10.10.129 226.1.1.2) wc=1 rpt=1 sparse=1
    Join list: (10.11.10.101 226.1.1.2) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=350 to v100
  group address: 227.1.1.2
    Join list: (10.10.129 227.1.1.2) wc=1 rpt=1 sparse=1
    Join list: (10.11.99.99 227.1.1.2) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=378 to v100
  group address: 227.1.1.1
    Join list: (10.10.129 227.1.1.1) wc=1 rpt=1 sparse=1
    Join list: (10.11.99.99 227.1.1.1) wc=0 rpt=0 sparse=1
send_J/P_msg(), lcl_adr=10.10.119 dest=224.0.0.13 pkt_size=406 to v100
00:22:16 Send PIM_V2 Join/Prune Src 10.10.119 Dst 224.0.0.13 Len 406 on v100
PIMSM: END Periodic join-prune msgs
```

debug ip pim nbr-change

Syntax: [no] **debug ip pim nbr-change** *nbr_address*

The *nbr_address* variable specifies the IP address of the neighbor.

This command controls the PIM IPv4 neighbor changes and the command output resembles the following example.

```
Brocade# debug ip pim nbr-change 10.144.144.12
PIM:  nbr-change  debugging is on
Brocade(config-if-e1000-1/1/11)# PIM-NBR.VRF0: Neighbor 10.144.144.12 on
interface tn4, 1/1/11 deleted on port-down
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.11) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.12) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.13) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.14) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.15) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.16) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.17) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.18) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.19) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.20) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.21) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.22) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.23) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.24) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.25) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.26) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.27) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.28) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.29) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.30) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.31) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.32) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.33) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.34) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.35) HW bits: 0 0
```

```
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.36) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.37) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.38) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.39) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.40) HW bits: 0 0
```

debug ip pim packet

Syntax: [no] **debug ip pim packet** *source_address group_address* [**rate**]

- **source_address** - Specifies the source address of the PIM packet
- **group_address** - Specifies the group address of the PIM packet.
- **rate** - Specifies the hardware rate.

This command displays debugging information about the PIM packets. Command output resembles the following example.

```
Brocade# debug ip pim packet 10.0.0.178 232.0.0.25
debug packet S=10.0.0.178 G=232.0.0.25
Brocade# VRF0 Flow Entry Delete (10.0.0.178 232.0.0.25) HW bits: 1 1

Brocade# debug ip pim packet 10.0.0.178 232.0.0.25 10
debug packet S=10.0.0.178 G=232.0.0.25 rate 10
Brocade# VRF0 Flow Entry Delete (10.0.0.178 232.0.0.25) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.178 232.0.0.25) HW bits: 0 0
```

debug ip pim physical-port ethernet

Syntax: [no] **debug ip pim physical-port ethernet** *stackid/slot/port*

This command displays information of the PIM physical ports that are connected. The *stackid/slot/port* variable refers to the stack ID, slot number, and port number.

```
Brocade# debug ip pim physical-port ethernet 2/1/5
Brocade# IGMP: IGMP: rcvd Report-V3(t=34) #rec=1, pkt S=10.2.1.6 to 224.0.0.22, on
v800 (phy 2
/1/5), igmp_size=16
IGMP: IGMP: rcvd Report-V3(t=34) #rec=1, pkt S=10.2.1.6 to 224.0.0.22, on v800
(phy 2/1/5), igmp_size=16
```

debug ip pim regproc

Syntax: [no] **debug ip pim regproc**

This command controls register processing and the command output resembles the following example.

```
Brocade# debug ip pim regproc
Brocade(config-if-e1000-1/1/11)# debug ip pim regproc
        PIM:  regproc debugging is on
Brocade(config-if-e1000-1/1/11)# enable
Brocade(config-if-e1000-1/1/11)# PIMSM-REG.VRF0: (10.0.0.168 228.0.0.1) Sending
L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.2) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.3) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.4) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.5) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.6) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.7) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.8) Sending L2-Null Reg on iface v78
```

```
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.9) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.10) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.11) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.12) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.13) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.14) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.15) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.16) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.17) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.18) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.19) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.20) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.21) Sending L2-Null Reg on iface v78
PIMSM-REG.VRF0: (10.0.0.168 228.0.0.22) Sending L2-Null Reg on iface v78
```

debug ip pim rp

Syntax: [no] **debug ip pim rp**

This command controls the Rendezvouz Point (RP) processing and the command output resembles the following example.

```
Brocade# debug ip pim rp
PIM:  rp  debugging is on
Brocade(config-if-e1000-1/1/11)# enable
Brocade(config-if-e1000-1/1/11)# PIMv4-CRP.VRF0: Sending Candidate RP Msg (size
22)
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.11) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.12) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.13) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.14) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.15) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.16) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.17) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.18) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.19) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.20) HW bits: 0 0
VRF0 Flow Entry Delete (10.0.0.168 232.0.0.21) HW bits: 0 0
```

debug ip pim source

Syntax: [no] **debug ip pim source** *ipaddr*

This command monitors the content related to a specific PIM server. The *ipaddr* variable refers to the IP address of the PIM server source.

```
Brocade# debug ip pim source 10.11.10.101
Brocade# join_prune_timer, (10.11.10.101 225.1.1.10) ->num=1, rpt=0,
am_rp_and_proxy=0, JOIN
join_prune_timer, (10.11.10.101 225.1.1.10) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.10) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.9) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.9) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.9) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.8) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
```

```
join_prune_timer, (10.11.10.101 225.1.1.8) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.8) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.7) ->num=1, rpt=0, am_rp_and_proxy=0, J
OIN
join_prune_timer, (10.11.10.101 225.1.1.7) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.7) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.6) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.6) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.6) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.5) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.5) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.5) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.4) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.4) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.4) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 226.1.1.1) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 226.1.1.1) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 226.1.1.1) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.3) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.3) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.3) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.2) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.2) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.2) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 225.1.1.1) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 225.1.1.1) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 225.1.1.1) fail RPT PRUNE cond.
join_prune_timer, (10.11.10.101 226.1.1.2) ->num=1, rpt=0, am_rp_and_proxy=0,
JOIN
join_prune_timer, (10.11.10.101 226.1.1.2) ->num=1, rpt=0, spt=1, SRC=v100,
RP=v100
join_prune_timer, (10.11.10.101 226.1.1.2) fail RPT PRUNE cond.
    Join list: (10.11.10.101 225.1.1.10) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.9) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.8) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.7) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.6) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.5) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.4) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 226.1.1.1) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.3) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.2) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 225.1.1.1) wc=0 rpt=0 sparse=1
    Join list: (10.11.10.101 226.1.1.2) wc=0 rpt=0 sparse=1
```

debug ip pim show

Syntax: debug ip pim show

This command displays the PIM debug settings. Do not use the **show debug** command for this purpose. The **show debug** command displays general debug information.

```
Brocade# debug ip pim show
pim debug-enable-any = 1
debug ip pim level 2 is enabled
```

debug ip pim vrf

Syntax: debug ip pim vrf *vrf_name*

This command displays the PIM related debugging information for the specified VRF.

```
Brocade# debug ip pim vrf vrf1
Debug: Jan  1 21:46:55 PIM_CFG.VRF1: port v85, ena_mode Dis
Debug: Jan  1 21:46:59 PIM_CFG.VRF1: port v85, ena_mode Ena
Debug: Jan  1 21:48:14 PIM_CFG.VRF1: port v85, ena_mode Dis
Debug: Jan  1 21:48:17 PIM_CFG.VRF1: port v85, ena_mode Ena
```

PIM-SM snooping debug commands

The following commands display information related to the Protocol Independent Multicast Sparse mode (PIM-SM) snoop debugging.

debug pimsm-snoop clear

Syntax: [no] **debug pimsm-snoop clear**

This command clears all the PIM-SM snooping debug settings.

debug pimsm-snoop control-source

Syntax: [no] **debug pimsm-snoop control-source** *source_address*

This command displays the PIM-SM snooping related debug information for the specified control source address. Command output resembles the following example.

```
Brocade# debug pimsm-snoop control-source 10.27.27.2
Brocade# PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Join/Prune from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Join/Prune from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
```

debug pimsm-snoop enable

Syntax: [no] **debug pimsm-snoop enable**

This command enables debugging of the PIM-SM snooping activity. Command output resembles the following example.

```
Brocade# debug pimsm-snoop enable
```

```
pimsm-snoop debug is enabled
```

debug pimsm-snoop group

Syntax: [no] **debug pimsm-snoop group** *group_address*

The *group_address* is the address of the PIM-SM snoop multicast group.

This command displays the PIM-SM snooping multicast group information. Command output resembles the following example.

```
Brocade# debug pim-sm-snoop group 228.228.0.1
Brocade# PIM from 10.27.27.2, vlan 275, e1/1/12 has G=228.228.0.1 with 1 JOIN, 0
PRUNE
    pim_snoop join (10.27.27.5 228.228.0.1), from e1/1/12
    PIM from 10.27.27.2, vlan 275, port 1/1/12: G=228.228.0.1 Src=10.0.0.1  (1
JOIN, 0 PRUNE)
PIM from 10.27.27.2, vlan 275, e1/1/12 has G=228.228.0.1 with 0 JOIN, 1 PRUNE
    pim_snoop prune 1/1/12 from (10.27.27.5, 228.228.0.1),     PIM from
10.27.27.2, vlan 275, port 1/1/12: G=228.228.0.1 Src=10.0.0.1  (0 JOIN, 2 PRUNE)
Pim snoop, age oif (* 228.228.0.1) remove e1/1/12, pim_prune=0
```

debug pimsm-snoop level

Syntax: [no] **debug pimsm-snoop level** *decimal*

The *decimal variable* specifies the debugging level of PIM-SM snooping activity and it can take values from 1 through 3.

This command sets the debuging level of PIM-SM snooping activity. Level 3 displays more information than level 2, and level 0 displays no information. Command output resembles the following example.

```
Brocade# debug pimsm-snoop level 1
pimsm-snoop debug level is set to 1
Brocade# PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.1 port 1/1/26 on vlan 275
```

debug pimsm-snoop physical-port ethernet

Syntax: [no] **debug pimsm-snoop physical-port ethernet** *num*

The *num* variable refers to the stack ID, slot number, and port number of a specific Ethernet port.

This command displays the PIM-SM snooping related debug information for the specified Ethernet interface. Command output resembles the following example.

```
Brocade# debug pimsm-snoop physical-port ethernet 1/1/12
Brocade# PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Join/Prune from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Join/Prune from 10.27.27.2 port 1/1/12 on vlan 275
PIMv2 Hello from 10.27.27.2 port 1/1/12 on vlan 275
```

debug pimsm-snoop show

Syntax: [no] **debug pimsm-snoop show**

This command displays all the PIM-SM snooping debug settings. Command output resembles the following example.

```
Brocade# debug pimsm-snoop show
pimsm-snoop debug-enable-any = 1
debug pimsm-snoop  is enabled
debug pimsm-snoop physical_port e4/48 is enabled
debug pimsm-snoop error is enabled
debug pimsm-snoop control-source 10.85.1.3 is enabled
```

debug pimsm-snoop source

Syntax: [no] **debug pimsm-snoop source** *source_address*

This command displays the PIM-SM snooping related debug information for the specified source address. Command output resembles the following example.

```
Brocade# debug pimsm-snoop source 10.27.27.5
Brocade# pim_snoop join (10.27.27.5 228.228.0.1), from e1/1/12
    pim_snoop join (10.27.27.5 228.228.0.2), from e1/1/12
    pim_snoop join (10.27.27.5 228.228.0.3), from e1/1/12
    PIM from 10.27.27.2, vlan 275, port 1/1/12: Src=10.27.27.5 with 0 JOIN with 0
PRUNE
    pim_snoop prune 1/1/12 from (10.27.27.5, 228.228.0.1),     pim_snoop prune
1/1/12 from (10.27.27.5, 228.228.0.2),     pim_snoop prune 1/1/12 from
(10.27.27.5, 228.228.0.3),     PIM from 10.27.27.2, vlan 275, port 1/1/12:
Src=10.27.27.5 with 0 JOIN with 3 PRUNE, 1st Grp=228.228.0.1
    pim_snoop join (10.27.27.5 228.228.0.1), from e1/1/12
    pim_snoop join (10.27.27.5 228.228.0.2), from e1/1/12
    pim_snoop join (10.27.27.5 228.228.0.3), from e1/1/12
    PIM from 10.27.27.2, vlan 275, port 1/1/12: Src=10.27.27.5 with 0 JOIN with 0
PRUNE
```

MLD snooping debug commands

Multicast Listener Discovery (MLD) discovers multicast listeners, or nodes that want to receive multicast packets on directly attached links. MLDv2 supports source filtering, the ability of a node to send reports on traffic that is from a specific address source or from all multicast addresses except the specified address sources. The information is then provided to the source-specific multicast (SSM) routing protocols such as PIM-SSM.

The following commands display information related to the MLD snoop debugging.

debug mld-snoop add-del-oif all

Syntax: [no] **debug mld-snoop add-del-oif all**

This command displays information about the addition or deletion of all the OIFs that are MLD snooping-enabled.

```
Brocade# debug mld-snoop add-del-oif all
mldsnoop add-del-oif all enabled
Brocade# Del e1/1/23  from (0x0 0x230005) vlan 130
Del e1/1/23  from (0x0 0x230004) vlan 130
Del e1/1/23  from (0x0 0x230003) vlan 130
Del e1/1/23  from (0x0 0x230002) vlan 130
Del e1/1/23  from (0x0 0x230001) vlan 130

Add e1/1/23  to (0x0 0x230001) vlan 130
Add e1/1/23  to (0x0 0x230002) vlan 130
Add e1/1/23  to (0x0 0x230003) vlan 130
```

```
Add e1/1/23  to (0x0 0x230004) vlan 130
Add e1/1/23  to (0x0 0x230005) vlan 130
```

debug mld-snoop clear

Syntax: [no] **debug mld-snoop clear**

This command clears all the MLD snooping debug settings.

debug mld-snoop error

Syntax: [no] **debug mld-snoop error**

This command displays the MLD snooping multicast error messages. Command output resembles the following example.

```
Brocade# debug mld-snoop error
monitor mldsnoop  error
Brocade# MLD:  Error : Ignoring MLD version 2 Report pkt due to version mismatch ,
pkt S=fe80::1, on VL130 (phy e1/1/2), mld_size=28
MLD:  Error : Ignoring MLD version 2 Report pkt due to version mismatch , pkt
S=fe80::1, on VL130 (phy e1/1/2), mld_size=28
MLD:  Error : Ignoring MLD version 2 Report pkt due to version mismatch , pkt
S=fe80::1, on VL130 (phy e1/1/2), mld_size=28
MLD:  Error : Ignoring MLD version 2 Report pkt due to version mismatch , pkt
S=fe80::1, on VL130 (phy e1/1/2), mld_size=28
MLD:  Error : Ignoring MLD version 2 Report pkt due to version mismatch , pkt
S=fe80::1, on VL130 (phy e1/1/2), mld_size=28
```

debug mld-snoop group

Syntax: [no] **debug mld-snoop group** *group_address*

The *group_address* is the address of the MLD snoop multicast group.

This command displays the MLD snooping multicast group information. Command output resembles the following example.

```
Brocade# debug mld-snoop group 2001:DB8::23:1
                 source fe80::1
                phy-port eth 1/1/23
                vlan 130

MLD send Query(t=130) V1, s=fe80::224:38ff:fe76:4bc0 :: to VL130(all) rsp=10000
mld=24B, pkt=86B
MLD: rcvd Leave(ty=132)  g=2001:DB8::23:1 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Leave(ty=132)  g=2001:DB8::23:2 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Leave(ty=132)  g=2001:DB8::23:3 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Leave(ty=132)  g=2001:DB8::23:4 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Leave(ty=132)  g=2001:DB8::23:5 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::23:1 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
```

```
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::23:2 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::23:3 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::23:4 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
MLD: rcvd Report-V1(ty=131)  g=2001:DB8::23:5 resp=0, pkt S=fe80::1, on VL130 (phy
e1/1/23), mld_size=24
```

debug mld-snoop mcache-group

Syntax: [no] **debug mld-snoop mcache-group** *group_address*

The *group_address* is the 32-bit address of the MLD snoop multicast group.

This command displays MLD snoop debugging information related to mcache activity for the specified group address.

```
Brocade# debug mld-snoop mcache-group 230001
Brocade(config)# Del e1/1/23  from (0x0 0x230001) vlan 130
Brocade(config)# Add e1/1/23  to (0x0 0x230001) vlan 130
```

debug mld-snoop packet

Syntax: [no] **debug mld-snoop packet** *source_address group_address*

The *source_address specifies the 32-bit source address* of the MLD snoop packet.

The *group_address* is the 32-bit address of the MLD snoop multicast group.

This command displays debugging information related to the MLD snoop packets. Command output resembles the following example.

```
Brocade# debug mld-snoop packet 3 230001
debug mldsnoop  packet (0x3 0x230001)
// dm raw: RX [a0a]2322::3                        ->ff1e::23:1
port: 1/1/1
rtr_l2mcast_fwd_data (pkts:2), (0x3 0x230001), from e1/1/1 vlan 130, has fwd_entry
rtr_l2mcast_fwd_data (pkts:1), (0x3 0x230001), from e1/1/1 vlan 130, clear
hw_aging flag
```

debug mld-snoop show

Syntax: [no] **debug mld-snoop show**

This command displays all the MLD snooping debug settings. Command output resembles the following example.

```
Brocade# debug mld-snoop show
debug mldsnoop  vlan 100 is enabled
```

debug mld-snoop timer

Syntax: [no] **debug mld-snoop timer**

This command displays the MLD snooping related timer events. Command output resembles the following example.

```
Brocade# debug mld-snoop timer
Brocade# mld timer, VL130(e1/1/2) change V3 client age out
```

Chapter

7

Security Diagnostics

This chapter contains the following sections:

ACL debug commands

The IP Access Control List (ACL) feature filters traffic based on the information in the IP packet header. The following command displays the ACL log information.

debug acl log

Syntax: [no] **debug acl log**

This command sends the ACL log to the CPU for all the packets.

```
Brocade# debug acl log
Enabling ACL log
```

802.1x debug commands

The following commands displays information about 802.1x authentication events, activities, and settings.

debug dot1x events

Syntax: [no] **debug dot1x events**

This command displays the authentications failed or succeeded and the application of VLANs or ACLs requested by the Remote Authentication Dial In User Service (RADIUS) server. This command works globally across all the ports.

```
Brocade# debug dot1x events
dot1x:  Events debugging is on
```

debug dot1x filter

Syntax: [no] **debug dot1x filter**

This command enables the 802.1x filter debugging.

```
Brocade# debug dot1x filter
dot1x:  Filter debugging is on
```

debug dot1x misc

Syntax: [no] **debug dot1x misc**

This command enables the 802.1x miscellaneous debugging.

```
Brocade# debug dot1x misc
dot1x:  Misc debugging is on
```

debug dot1x packets

Syntax: [no] **debug dot1x packets**

This command displays information about 802.1x packets.

```
Brocade# debug dot1x packets
dot1x:  Packets debugging is on
```

debug dot1x timers

Syntax: [no] **debug dot1x timers**

This command displaysinformation about 802.1x timers.

```
Brocade# debug dot1x timers
dot1x:  Timers debugging is on
```

MAC authentication debug commands

The following command displays MAC authentication information.

debug mac-authentication

Syntax: [no] **debug mac-authentication** *hex*

This command enables the MAC authentication debugging for a single port. The *hex* variable specifies the hexadecimal number of the port.

NOTE
This debug command displays output that is useful for the technical support personnel. Use the **ptrace aaa** command instead of the **mac authentication** command to display information about the authentication of packets between the switch and the RADIUS server.

```
Brocade# ptrace aaa
specified trace was turned ON
Brocade# clear auth
Brocade# Reseting RADIUS Client structure
RADIUS: Create client 15, Total number of active clients=1
```

```
Tracing the outgoing Radius Authentication packet..
UDP packet source IP=198.19.9.1, port=1645, destination IP=64.12.12.5, port=1745
Radius Header (hex): Code=0x01 Identifier=0x11 Length=0x0065
Authenticator (hex):000078d6000028c00000699d00004bf3
 Attribute Type(hex)=0x01 Len(hex)=0x0e Value (ASCII)=0 0 3 0 4 8 7 4 3 1 8 1
 Attribute Type(hex)=0x02 Len(hex)=0x12 Value (hex)=414084e32550daec45ef8129b7c6
aed9
 Attribute Type(hex)=0x06 Len(hex)=0x06 Value (hex)=00000002
 Attribute Type(hex)=0x0c Len(hex)=0x06 Value (hex)=000005dc
 Attribute Type(hex)=0x04 Len(hex)=0x06 Value (hex)=c6130901
 Attribute Type(hex)=0x3d Len(hex)=0x06 Value (hex)=0000000f
 Attribute Type(hex)=0x05 Len(hex)=0x06 Value (hex)=0000038c
 Attribute Type(hex)=0x1f Len(hex)=0x13 Value (ASCII)=0 0 - 3 0 - 4 8 - 7 4 - 3
1 - 8 1

Radius message received from server.
Tracing the received Radius packet..
Radius Header (hex): Code=0x02 Identifier=0x11 Length=0x0051
Authenticator (hex):c98e259aa7b6aeb66baf79af9a77e81e
 Attribute Type(hex)=0x41 Len(hex)=0x06 Value (hex)=00000006
 Attribute Type(hex)=0x51 Len(hex)=0x05 Value (hex)=313939
 Attribute Type(hex)=0x40 Len(hex)=0x06 Value (hex)=0000000d
 Attribute Type(hex)=0x07 Len(hex)=0x06 Value (hex)=00000001
 Attribute Type(hex)=0x06 Len(hex)=0x06 Value (hex)=00000002
 Attribute Type(hex)=0x19 Len(hex)=0x20 Value (hex)=3cdc04f60000013700010a144086
01cb1fa09e8d214e00000000000001b2

RADIUS: Unsupported RADIUS message 7 with code 2
RADIUS: Unsupported RADIUS message 6 with code 2
RADIUS: Unsupported RADIUS message 25 with code 2
+++++ Timer cancelled for client 15.
Radius server returns status PASS.
AAA:DYNAMIC-AUTHEN:Authentication successful.

SYSLOG: <13>0d00h25m33s:SDX1600 MAC Authentication succeeded for [0000.0074.3181
 198.19.9.105] on port 15/12
RADIUS: Reset client 15, Total number of active clients=0

Reseting RADIUS Client structure
```

sFlow debug commands

sFlow is a system for observing traffic flow patterns and quantities within and among a set of Brocade devices. The following command displays debug information related to sFlow.

debug sflow problems

Syntax: [no] **debug sflow problems**

This command enables debuggging of the internal sFlow.

```
Brocade# debug sflow problems
sflow:  problem debugging is on
```

IP security debug commands

The following command displays debug information related to Internet Protocol security (IPsec) operation for OSPFv3.

debug ipsec all

Syntax: [no] **debug ipsec all**

This command displays all debuggging information related to IPsec operation as shown in the following example.

```
Brocade# debug ipsec all
IPSec:  all debugging is on
Brocade(config-ospf6-router)# no area 0 auth ipsec spi 400 esp sha1
abcef12345678901234fedcba098765432109876IPSEC,SA: ipipsec_pfkeyv2_input() ::
receiving 'DELETE' command
IPSEC,SA: Removing SA: ESP  in spi=0x190 dst=FE80::
IPSEC,Policy: Removing flow [input use 'prot=OSPF src=FE80::/10:0 dst=::/0:0' ->
SA: ESP  in spi=0x190 dst=FE80::] : ok
IPSEC,SA: ipipsec_pfkeyv2_input() :: succeeded
```

debug ipsec esp

Syntax: [no] **debug ipsec esp**

This command enables debugging of Encapsulating Security Payload (ESP) and displays an output as shown in the following example.

```
Brocade# debug ipsec esp
IPSec:  esp debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 02:03:27 IPSEC,ESP: decrypt ok, seq=0
(SA: ESP  in spi=0x190 dst=FE80::)
Debug: Jan  1 02:03:27 IPSEC,ESP: decrypt ok, seq=0  (SA: ESP  in spi=0x190
dst=FE80::)
Debug: Jan  1 02:03:36 IPSEC,ESP: decrypt ok, seq=0  (SA: ESP  in spi=0x190
dst=FE80::)
Debug: Jan  1 02:03:36 IPSEC,ESP: decrypt ok, seq=0  (SA: ESP  in spi=0x190
dst=FE80::)
Debug: Jan  1 02:03:45 IPSEC,ESP: decrypt ok, seq=0  (SA: ESP  in spi=0x190
dst=FE80::)
Debug: Jan  1 02:03:45 IPSEC,ESP: decrypt ok, seq=0  (SA: ESP  in spi=0x190
dst=FE80::)
```

debug ipsec in

Syntax: [no] **debug ipsec in**

This command enables the display of debugging information related to inbound OSPFv3 packets with IPsec.

```
Brocade# debug ipsec in
IPSec:  in debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 02:04:15 IPSEC,IN:  ESP  spi=400 (pkt
'ESP  FE80:: -> FE80::') payloadlength =64
Debug: Jan  1 02:04:15 IPSEC,IN: Incoming packet matches Policy :  input use
'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::
Debug: Jan  1 02:04:15 IPSEC,IN:  ESP  spi=400 (pkt 'ESP  FE80:: -> FE80::')
payloadlength =64
```

```
Debug: Jan  1 02:04:15 IPSEC,IN: Incoming packet matches Policy :  input use
'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::
Debug: Jan  1 02:04:26 IPSEC,IN:  ESP  spi=400 (pkt 'ESP  FE80:: -> FE80::')
payloadlength =64
Debug: Jan  1 02:04:26 IPSEC,IN: Incoming packet matches Policy :  input use
'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::
Debug: Jan  1 02:04:26 IPSEC,IN:  ESP  spi=400 (pkt 'ESP  FE80:: -> FE80::')
payloadlength =64
Debug: Jan  1 02:04:26 IPSEC,IN: Incoming packet matches Policy :  input use
'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::
Debug: Jan  1 02:04:36 IPSEC,IN:  ESP  spi=400 (pkt 'ESP  FE80:: -> FE80::')
payloadlength =64
Debug: Jan  1 02:04:36 IPSEC,IN: Incoming packet matches Policy :  input use
'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::
Debug: Jan  1 02:04:36 IPSEC,IN:  ESP  spi=400 (pkt 'ESP  FE80:: -> FE80::')
payloadlength =64
Debug: Jan  1 02:04:36 IPSEC,IN: Incoming packet matches Policy :  input use
'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::
```

debug ipsec out

Syntax: [no] **debug ipsec out**

This command enables the display of debugging information related to outbound OSPFv3 packets with IPsec.

```
Brocade# debug ipsec out
IPSec:  out debugging is on
Brocade(config-ospf6-router)# Debug: Jan  1 02:04:55 IPSEC,OUT:    Matching Flow:
output use 'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  out spi=0x190 dst=::
Debug: Jan  1 02:04:55 IPSEC,OUT: SA ESP  out spi=0x190 dst=:: payloadlength =64
Debug: Jan  1 02:04:56 IPSEC,OUT: OSPF FE80:: -> FE80::, payloadlength =40
Debug: Jan  1 02:04:56 IPSEC,OUT: OSPF FE80:: -> FE80::, payloadlength =40
Debug: Jan  1 02:05:06 IPSEC,OUT:    Matching Flow:  output use 'prot=OSPF
src=FE80::/10:0 dst=::/0:0' -> SA: ESP  out spi=0x190 dst=::
Debug: Jan  1 02:05:06 IPSEC,OUT: SA ESP  out spi=0x190 dst=:: payloadlength =64
Debug: Jan  1 02:05:07 IPSEC,OUT: OSPF FE80:: -> FE80::, payloadlength =40
Debug: Jan  1 02:05:07 IPSEC,OUT: OSPF FE80:: -> FE80::, payloadlength =40
Debug: Jan  1 02:05:15 IPSEC,OUT:    Matching Flow:  output use 'prot=OSPF
src=FE80::/10:0 dst=::/0:0' -> SA: ESP  out spi=0x190 dst=::
Debug: Jan  1 02:05:15 IPSEC,OUT: SA ESP  out spi=0x190 dst=:: payloadlength =64
Debug: Jan  1 02:05:16 IPSEC,OUT: OSPF FE80:: -> FE80::, payloadlength =40
Debug: Jan  1 02:05:16 IPSEC,OUT: OSPF FE80:: -> FE80::, payloadlength =40
```

debug ipsec policy

Syntax: [no] **debug ipsec policy**

This command enables the display of debugging information for IPsec policy.

```
Brocade# debug ipsec policy
IPSec:  policy debugging is on

Brocade(config-ospf6-router)#no area 0 auth ipsec spi 400 esp sha1
abcef12345678901234fedcba098765432109876
Debug: Jan  1 01:57:05 IPSEC,Policy: Removing flow [input use 'prot=OSPF
src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::] : ok

Brocade(config-ospf6-router)#area 0 auth ipsec spi 400 esp sha1
abcef12345678901234fedcba098765432109876
```

```
Debug: Jan  1 01:57:26 IPSEC,Policy: Creating flow [input use 'prot=OSPF
src=FE80::/10:0 dst=::/0:0' -> SA: ESP  in spi=0x190 dst=FE80::] : ok

Brocade(config-ospf6-router)#Debug: Jan  1 02:02:21 IPSEC,Policy: Creating flow
[output use 'prot=OSPF src=FE80::/10:0 dst=::/0:0' -> SA: ESP  out spi=0x190
dst=::] : ok
```

debug ipv6 ospf ipsec

Syntax: [no] **debug ipv6 ospf ipsec**

TThis command displays information about IPsec events. This command can show if IPsec is actually providing its services to IPv6 OSPFv3. The following output shows success in the attempts to provide various IPsec services to OSPFv3.

```
Brocade# debug ipv6 ospf ipsec
OSPFv3:  ipsec debugging is on
Brocade(config-ospf6-router)# no area 0 auth ipsec spi 400 esp sha1
abcef12345678901234fedcba098765432109876Debug: Jan  1 02:22:09 OSPFv3:
ITC_AUTHENTICATION_CONFIG message received
Debug: Jan  1 02:22:09 OSPFv3: Auth timer started
Debug: Jan  1 02:22:09 OSPFv3: Key Rollover, for area 0.0.0.0, state change
NOT_ACTIVE->STARTED
Debug: Jan  1 02:22:09 OSPFv3: Key Rollover, for v9, state change
NOT_ACTIVE->STARTED
```

Web events debug commands

The following command displays information related to web debugging.

debug web events

Syntax: [no] **debug web events**

This command enables web events debugging.

```
Brocade# debug web events
web:  Events debugging is on
```

Web authentication debug commands

The following commands display information related to the web authentication debugging.

debug webauth events

Syntax: [no] **debug webauth events**

This command enables debugging of web authentication events.

```
Brocade# debug webauth events
webauth:  Events debugging is on
```

debug webauth timers

Syntax: [no] **debug webauth timers**

This command enables debugging of web authentication timers.

```
Brocade# debug webauth timers
webauth:  Timers debugging is on
```

RA guard debug commands

The Router Advertisement (RA) guard feature controls the propagation of router advertisements through the local IPv6 networks and thereby improves the security of IPv6 networks. The following debug command enables RA guard debugging.

debug ipv6 raguard

Enables RA guard debug messages.

Syntax **debug ipv6 raguard**

no debug ipv6 raguard

Parameters None

Modes Privileged EXEC mode

Examples The command output resembles the following example.

```
Brocade# debug ipv6 raguard
00 days 00h40m49s:I:RAGuard: prefix in router advertisement rejected by
prefix-list
00 days 00h40m49s:I:RAGuard: prefix in router advertisement passed prefix-list

00 days 00h40m49s:I:RAGuard: Packet received on Host port 1/1/5. Dropped.
00 days 00h40m49s:I:RAGuard: Packet received on a Trusted port 1/1/5. Forward.

00 days 00h40m49s:I:RAGuard: Packet preference is greater than policy configured
maximum on port 1/1/5, vlan 10

00 days 00h40m49s:I:RAGuard: IPv6 address of packet matches Whitelist
fe80:100::100:1 on port 1/1/5, vlan 10
00 days 00h40m49s:I:RAGuard: IPv6 address of the packet does not match the
whitelist fe80:100::100:1 on port 1/1/5, vlan 10

00 days 00h40m49s:I:RAGuard: packet passes prefix-list examplelist1 on port 1/1/5,
vlan 10
00 days 00h40m49s:I:RAGuard: packet does not pass prefix-list examplelist1 on port
1/1/5, vlan 10
00 days 00h40m49s:I:RAGuard: prefix-list examplelist1 on policy examplepolicy1
not found

00 days 00h40m49s:I:RAGuard: Not enabled for vlan 10, port 1/1/5
00 days 00h40m49s:I:RAGuard: Feature ON for vlan 10, port 1/1/5
```

History

Release	Command History
Release 08.0.00a	The command was introduced.

Chapter

supportsave Diagnostics

8

This chapter contains the following sections:

supportsave overview

The supportsave is an important utility for collecting logs from the driver, internal libraries, and firmware. The collected logs are shared with the technical support personnel for investigating issues seen on the device.

Run the commands listed in the "supportsave commands" section to collect the logs and upload them to the remote server for troubleshooting.

supportsave commands

The **supportsave** command has the following advantages over the **show tech-support** command:

- Allows you to add additional commands to collect more data.
- Allows to transfer the collected data to an external server such as Trivial File Transfer Protocol (TFTP) server.

TFTP is disabled by default, if FIPS is enabled. Enable TFTP manually for uploading supportsave data. It is a prerequisite to have the TFTP server with a write permission and the server must be accessible from the device.

Use the **fips policy allow tftp-access** command in configuration mode to enable TFTP access while the FIPS is enabled.

Limitations and pre-requisites

The following are the limitations and pre-requisites of the supportsave utility:

- Parallel execution of the **supportsave** command from two different sessions is not allowed.
- Parallel execution of the **supportsave** command and the **copy tftp/scp** commands is not allowed.
- Only IPv4 address is supported for the TFTP destination.
- Only up to 32 commands can be added to the custom command list.
- Commands are not expanded while adding a command to the custom commands list.
- Commands added to the custom command list must be privileged EXEC mode MP commands.

- The **supportsave** command cannot be added to the custom commands list. Also, the commands which change the CLI mode (**exit**, **quit**, **end**) and commands which restart the router (**switchover**, **reload**, **reload-yes**) must not be added.
- Modifying the custom commands list using the **supportsave add-cmd** or **supportsave del-cmd** commands is not allowed while the supportsave data collection is in progress.
- Time taken by the **supportsave** command depends on the commands present in the list.
- The **supportsave** command does not support collecting crash logs from standby units in a mixed stack.
- The **supportsave core** command can collect core logs from one of the Brocade ICX 6450 units in the mixed stack.
- User tags cannot be added when collecting core logs from Brocade ICX 6450 units in a mixed stack.
- The **supportsave all** command will not collect core logs from Brocade ICX 6450 units in a mixed stack.

Table 2 displays the individual Brocade FastIron devices and the **supportsave** commands they support.

TABLE 2 Supported supportsave commands

supportsave commands	Mixed stacking		Brocade ICX 6430 Brocade ICX 6450 Brocade ICX 6650 Brocade ICX 7750 Brocade FastIron SX	Brocade ICX 6610 Brocade FCX
	Brocade ICX 6610	Brocade ICX 6450		
supportsave all	Supported	Supported	Supported	Supported
supportsave os	Not supported	Not supported	Supported	Not supported
supportsave core	[1]Supported	Supported	Supported	Not supported
supportsave l2	Supported	Supported	Supported	Supported
supportsave l3	Supported	Supported	Supported	Supported
supportsave custom	Supported	Supported	Supported	Supported
supportsave platform	Supported	Supported	Supported	Supported
supportsave show	Supported	Supported	Supported	Supported
supportsave cancel	Supported	Supported	Supported	Supported
supportsave add_cust_cmd index	Supported	Supported	Supported	Supported
supportsave del_cust_cmd index	Supported	Supported	Supported	Supported
supportsave del_cust_cmd all	Supported	Supported	Supported	Supported
supportsave list_cust_cmd	Supported	Supported	Supported	Supported
supportsave info enable	Supported	Supported	Supported	Supported
supportsave info disable	Supported	Supported	Supported	Supported

1. The **core** option is supported on the Brocade ICX 6610 only when the Brocade ICX 6610 is part of a mixed stack.

The following **supportsave** commands are used to collect all relevant support information to a remote TFTP server location.

supportsave

Syntax: **supportsave** [**all** | **os** | **platform** | **l2** | **l3** | **custom** | **core**] *tftp_server_ IP tftp_ server's_ relative_ path_name* [**user_tag***]*

- **all -** Collects all supportsave related log files.
- **os** - Collects operating system (OS) related information.
- **platform -** Collects platform related information.
- **l2** - Collects Layer 2 related information.
- **l3** - Collects Layer 3 related information.
- **custom -** Executes all commands present in the default and custom commands list and sends the output to the specified TFTP server address.
- **core -** Collects all core logs from one of the Brocade ICX 6450 units in the mixed stack.
- *tftp_server_ IP* - Specifies the IP address of the TFTP server.
- *tftp_server's_ relative_ path_ name* - Specifies the file path name of TFTP server relative to TFTP server's default directory to dump supportsave logs. To specify the default directory itself, use '/'.
- **user_tag -** Specifies the user tag to be added in the supportsave collected file name. Maximum of 10 characters are allowed.

NOTE
Brocade recommends to use **all** option to collect complete logs.

To collect all supportsave related logs, enter the **supportsave all** command as shown in the following example.

```
Brocade# supportsave all 10.37.2.40 fi/prince/supportsave
Supportsave started. This operation may take several minutes.
Press "A" to abort supportsave operation.
ICX6430-48P
Brocade#.**********************.....................................................
.....................................................................................
Supportsave completed in 10 seconds
```

To collect core logs from a Brocade ICX 6450 unit in a mixed stack, enter the **supportsave core** command and specify the unit number from which core logs need to be collected as shown in the following example. The unit number can take values from 1 through 8.

```
Brocade# supportsave core 10.70.41.42 / unit 2
Supportsave started. This operation may take several minutes.
Press "A" to abort supportsave operation.
ICX6610-24
Brocade#.**********************.....................................................
.....................................................................................
Supportsave completed in 132 seconds
```

supportsave show

Syntax: supportsave [**show** | **cancel**]

- **show** - Displays the progress of the log collection that has been executed on a terminal other than the one on which the **supportsave** command is executed.
- **cancel** - Cancels the current log collection that has been executed on a terminal other than the one on which the **supportsave** command is executed.

The following is the sample output from the **supportsave show** command.

```
Brocade# supportsave show
Supportsave is executing commands: 19% completed
Brocade# supportsave show
Supportsave is executing commands: 34% completed
Brocade# supportsave show
Supportsave is executing commands: 60% completed
Brocade# supportsave show.
Supportsave is executing commands: 100% completed
Supportsave is transporting files...
```

supportsave add_cust_cmd index

Syntax: supportsave add_cust_cmd index *index_number "cli string"*

- *index_number* - Specifies the index in the custom commands list where an additional command need to be added. The valid range is from 1 through 32.
- *cli string* - Specifies the commands to be added to the custom commands list.

This command adds a run-time executable command into the custom module at a particular index of 32 characters long (maximum of 32 characters are allowed) during the run time.

```
Brocade# supportsave add_cust_cmd index 1 "show chassis"
```

supportsave del_cust_cmd index

Syntax: supportsave del_cust_cmd index *index_number*

The *index_number* variable specifies the index of a command in the custom commands list that needs to be removed. The valid range is from 1 through 32.

This command deletes a command from the custom module at a particular index during the run time.

supportsave del_cust_cmd all

Syntax: supportsave del_cust_cmd all

This command deletes all the custom module commands.

supportsave list_cust_cmd

Syntax: supportsave list_cust_cmd

This command displays all commands present in the custom commands list.

supportsave info enable

Syntax: supportsave info enable

This command enables the header to be displayed for all show commands being executed on CLI. The header format is modified to include the following information:

- BEGIN - Indicates the command that will begin execution next.
- CONTEXT - Indicates in which session the command is executed.
- TIME STAMP - A time stamp, with millisecond granularity, helps to determine time difference between separate runs of the same command.
- HW/SW INFO - Indicates the hardware and software version information.

The following sample output of **show cpu** command shows the modified header format.

```
Brocade# supportsave info enable
Brocade# show cpu

================================================================================
BEGIN : show cpu
CONTEXT : CONSOLE#0
TIME STAMP : 12:09:19.238 GMT+05:30 Wed Nov 06 2013
HW/SW INFO : ICX6610-48-HPOE/FCXR08010b1
================================================================================
1 percent busy, from 1689 sec ago
1   sec avg:  1 percent busy
5   sec avg:  1 percent busy
60  sec avg:  1 percent busy
300 sec avg:  1 percent busy
```

supportsave info disable

Syntax: supportsave info disable

This command disables the header to be displayed for all show commands being executed on CLI.

Recovery during continuous reload

Perform the following steps to collect the logs in case of continuous reload of the device on a given partition:

- Press **b** to stop reloading in boot prompt.
- Boot from the stable firmware from the other partition using *boot_primary* or *boot_secondary* options.
- Run **supportsave** command to collect the logs after the reload is finished with the stable firmware.
- Execute the **use default-configuration** command from the boot prompt for stopping the start up configuration during reload, and make sure the device loads successfully. Otherwise, contact the Brocade technical support for assistance.

Chapter

Technical Support Diagnostics

9

This chapter contains the following sections:

show tech-support

The **show tech-support** command is useful when collecting a large amount of information about the Brocade FastIron devices for troubleshooting purposes. The output of this command can be used by technical support representatives when a problem is reported.

show tech-support

Syntax: [no] **show tech-support** [**acl** | **cluster** | **cpu** | **license** | **l2** | **l3** | **memory** | **multicast** | **packet-loss** | **stack**]

- **acl** - Generates system and debugging information specific to ACL configurations and counters.
- **cluster** - Generates system and debugging information specific to cluster configurations.
- **cpu** - Generates CPU-related information.
- **license** - Generates license-related information.
- **l2** - Generates system and debugging information specific to Layer 2 configurations.
- **l3** - Generates system and debugging information specific to Layer 3 configurations.
- **memory** - Generates memory-related information of the device.
- **multicast** - Generates system and debugging information specific to Layer 2 and Layer 3 multicast configurations.
- **packet-loss** - Generates packet statistics-related debugging information.
- **stack** - Generates system and debugging information specific to stacking configurations.

The **show tech-support** command displays the output of several show commands at once. The output from this command varies depending on the router configuration.

The default output of the **show tech-support** command includes the following information:

- Running configuration
- Image version
- Port status
- Port counters
- Static and dynamic log buffers
- dm statistics
- Boot, monitor, and system

- Registers information
- Possible stack trace
- Active stack (if applicable)
- Last packet (Application Data)
- Possible data structure
- MCT cluster details
- License details

The following example shows the truncated output of the **show tech-support** command.

```
Brocade# show tech-support
Current configuration:
!
ver 07.4.00b1T7f3
!
stack unit 1
  module 1 fcx-48-poe-port-management-module
  module 2 fcx-cx4-2-port-16g-module
!

!
enable snmp config-tacacs
hostname FxRouter
ip route 0.0.0.0 0.0.0.0 172.26.64.1
!
logging console
username brocade password .....
tacacs-server host 172.26.50.95 auth-port 49 default key 1 $d=
tacacs-server host 172.26.50.94 auth-port 49 default key 1 $bn^^s
tacacs-server host 172.26.50.93 auth-port 49 default key 1 $r=?!\!
tacacs-server host ipv6 :: auth-port 49 default key 1 $!66i
tacacs-server key 1 $!66i
snmp-server community ..... ro
snmp-server community ..... rw
snmp-server community ..... rw

interface management 1
 ip address 172.26.67.52 255.255.240.0
!
!
!
end

  Copyright (c) 1996-2011 Brocade Communications Systems, Inc.
    UNIT 1: compiled on Dec 20 2011 at 08:58:01 labeled as FCXR07400b1
       (7155950 bytes) from Secondary FCXR07400b1.bin
        SW: Version 07.4.00b1T7f3
  Boot-Monitor Image size = 369369, Version:07.3.00T7f5 (grz07300b1)
  HW: Stackable FCX648S-HPOE-PREM (PROM-TYPE FCX-ADV-U)
==========================================================================
UNIT 1: SL 1: FCX-48GS POE 48-port Management Module
     Serial  #: BCYXXXXXXXX
     License: FCX_ADV_ROUTER_SOFT_PACKAGE   (LID: deaHHJIhFJv)
     P-ENGINE  0: type DB90, rev 01
     P-ENGINE  1: type DB90, rev 01
     PROM-TYPE: FCX-ADV-U
==========================================================================
UNIT 1: SL 2: FCX-2XGC 2-port 16G Module (2-CX4)
```

```
==========================================================================
  800 MHz Power PC processor 8544E (version 0021/0022) 400 MHz bus
65536 KB flash memory
  256 MB DRAM
Monitor Option is on
STACKID 1  system uptime is 31 minutes 41 seconds
The system : started=warm start reloaded=by "reload"
*** NOT FOR PRODUCTION ***

Port    Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/1   Down    None    None None  None  No  1    0   0000.00c9.5680
1/1/2   Down    None    None None  None  No  1    0   0000.00c9.5681
1/1/3   Down    None    None None  None  No  1    0   0000.00c9.5682
1/1/4   Down    None    None None  None  No  1    0   0000.00c9.5683
1/1/5   Up      Forward Full 1G    None  No  1    0   0000.00c9.5684
1/1/6   Down    None    None None  None  No  1    0   0000.00c9.5685
1/1/7   Down    None    None None  None  No  1    0   0000.00c9.5686
1/1/8   Down    None    None None  None  No  1    0   0000.00c9.5687
1/1/9   Up      Forward Full 1G    None  No  1    0   0000.00c9.5688
1/1/10  Down    None    None None  None  No  1    0   0000.00c9.5689
1/1/11  Down    None    None None  None  No  1    0   0000.00c9.568a
1/1/12  Down    None    None None  None  No  1    0   0000.00c9.568b
1/1/13  Down    None    None None  None  No  1    0   0000.00c9.568c
1/1/14  Down    None    None None  None  No  1    0   0000.00c9.568d
1/1/15  Down    None    None None  None  No  1    0   0000.00c9.568e
1/1/16  Down    None    None None  None  No  1    0   0000.00c9.568f
1/1/17  Down    None    None None  None  No  1    0   0000.00c9.5690
1/1/18  Down    None    None None  None  No  1    0   0000.00c9.5691
1/2/1   Down    None    None None  None  No  1    0   0000.00c9.56b1
1/2/2   Down    None    None None  None  No  1    0   0000.00c9.56b2
mgmt1   Up      None    Full 1G    None  No  None 0   0000.00c9.5680

Port    Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/1   Down    None    None None  None  No  1    0   0000.00c9.5680

 Port 1/1/1 Counters:
         InOctets                    0          OutOctets                    0
           InPkts                    0            OutPkts                    0
  InBroadcastPkts                    0    OutBroadcastPkts                   0
  InMulticastPkts                    0    OutMulticastPkts                   0
    InUnicastPkts                    0      OutUnicastPkts                   0
        InBadPkts                    0
      InFragments                    0
       InDiscards                    0          OutErrors                    0
              CRC                    0         Collisions                    0
         InErrors                    0      LateCollisions                   0
      InGiantPkts                    0
      InShortPkts                    0
         InJabber                    0
   InFlowCtrlPkts                    0     OutFlowCtrlPkts                   0
     InBitsPerSec                    0       OutBitsPerSec                   0
     InPktsPerSec                    0       OutPktsPerSec                   0
    InUtilization                0.00%      OutUtilization                0.00%

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 15 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
```

```
0d00h00m11s:I:System: Stack unit 1 POE  Power supply 1  with 410000 mwatts
capacity is up

Dynamic Log Buffer (50 lines):
0d00h31m23s:W:System:Stack unit 1 Fan speed changed automatically to 2
0d00h28m33s:W:System:Stack unit 1 Fan speed changed automatically to 1
0d00h22m33s:W:System:Stack unit 1 Fan speed changed automatically to 2
.
.
(output truncated)
```

The format of the **show tech-support** command output is modified to include a header and a footer for each of the commands which gets called from the CLI to automatically parse the output for easier lookup.

The header contains the following information:

- BEGIN - Indicates a sub-command that will begin execution next.
- CONTEXT - Indicates in which context and session the sub-commands are executed.
- TIME STAMP - A time stamp, with millisecond granularity, helps to determine the time difference between separate runs of the same command.
- HW/SW INFO - Indicates the hardware and software version information.

The footer contains the following information:

- TIME STAMP - A time stamp, with millisecond granularity, helps to determine the time difference between separate runs of the same command.
- END - Indicates the sub-command which has completed execution.
- TIME TAKEN - Indicates the total time taken in nanoseconds for the command execution.

By default, the header and footer will not be displayed for all **show tech-support** commands being executed on the CLI. You can enable the display of the header and footer for **show tech-support** commands by using the **supportsave info enable** command.

You can disable the display of the header and footer for **show tech-support** commands by using the **supportsave info disable** command.

The following sample output of **show tech-support license** command shows the header and footer displayed.

```
Brocade# supportsave info enable

Brocade# show tech-support license
================================================================================
BEGIN : show tech license
CONTEXT : CONSOLE#0
TIME STAMP : 12:06:12.439 GMT+05:30 Wed Nov 06 2013
HW/SW INFO : ICX6610-48-HPOE/FCXR08010b1
================================================================================
================================================================================
BEGIN : show license
CONTEXT : CONSOLE#0 : LICENSE
TIME STAMP : 12:06:12.439 GMT+05:30 Wed Nov 06 2013
HW/SW INFO : ICX6610-48-HPOE/FCXR08010b1
================================================================================

License Information
===================
Index      License Name         Lid          License   Status   License   License
```

```
                                                Type               Period   Capacity
Stack unit 1:
1          ICX6610-10G-LIC-POD  rtIHFJFFHNO   Normal   Active   Unlimited  8
================================================================================
TIME STAMP : 12:06:12.447 GMT+05:30 Wed Nov 06 2013
END : show license
TIME TAKEN : 419222 ticks (8384440 nsec)
================================================================================
```

show tech-support acl

Syntax: show tech-support acl

This command displays debugging information related to ACL configuration and counters. Command output resembles the following example.

```
Brocade# show tech-support acl
ACL Config Information.
ipv6 access-list bvs: 52 entries
Standard IP access list 1  (hw usage : 2)
permit any  (hw usage : 1)

Standard IP access list 2  (hw usage : 2)
permit any  (hw usage : 1)

Standard IP access list 3  (hw usage : 2)
permit any  (hw usage : 1)

Standard IP access list 4  (hw usage : 2)
permit any  (hw usage : 1)

Standard IP access list 5  (hw usage : 2)
permit any  (hw usage : 1)

Extended IP access list 101  (hw usage : 3)
permit tcp any any  (hw usage : 1)
permit ip host 192.168.2.2 any  (hw usage : 1)

ACL-ON-ARP list information:
Port        ACL ID  Filter Count
ve2         103     0
ve3         101     0
ve4         101     0
ACL Accounting Information :
Traffic Policy - g_voip:
General Counters:
Port Region#Byte Count Packet Count
------------------ -------------------- ----------------------
7 (4/1 - 4/12) 85367040 776064
All port regions 84367040 776064

Rate Limiting Counters:
Port Region# Green Conformance Yellow Conformance Red Conformance
-------------------------------- ------------------ ----------------

7 (4/1 - 4/12) 329114195612139520 37533986897781760 0
All port regions    329114195612139520  37533986897781760 0
```

show tech-support cpu

Syntax: show tech-support cpu

This command displays debugging information related to CPU usage and utilization. Command output resembles the following example.

```
Brocade# show tech-support cpu
CPU Usage Information.
4 percent busy, from 1931 sec ago
1   sec avg:  1 percent busy
5   sec avg:  1 percent busy
60  sec avg:  1 percent busy
300 sec avg:  1 percent busy

CPU Utilization information.(5 -Times)
idle pid 0 id 1 cpu 102738 usage                    0
main pid 1 id 1 cpu 102738 usage                   21
flash pid 0 id 1 cpu 102738 usage                   0
loop cnt: 168752871 168752871 -> 0
read cnt: 67897 67897 -> 0
loop rate = 0/1 = 0
rcode = 100 - (100*0/2300) = 100
99/100 busy, from 0 sec ago

CPU(per process) usage  Information.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.01      0.02      0.02           1639
BGP              0.00      0.00      0.00      0.00              0
DOT1X            0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.00      0.00      0.00      0.00            519
IP               0.00      0.00      0.00      0.00            489
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              2
STP              0.01      0.01      0.01      0.01           1078
VRRP             0.00      0.00      0.00      0.00            392
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
IPv6             0.01      0.01      0.01      0.01           1053
ICMP6            0.01      0.02      0.02      0.02           1723
ND6              0.00      0.00      0.00      0.00              7
RIPng            0.00      0.00      0.00      0.00              4
OSPFv3           0.00      0.00      0.00      0.00              0
IPV6_RX          0.00      0.00      0.00      0.00              0
```

show tech-support l2

Syntax: show tech-support l2

This command displays the following debugging information related to Layer 2 configurations:

- Interface status
- Virtual Local Area Network (VLAN) configuration
- Private VLAN
- Spanning tree and 802.1W spanning tree
- Per VLAN Spanning Tree Plus (PVST+)
- Bridge Protocol Data Unit (BPDU) Guard and STP root protection
- VLAN topology group

- Link aggregation configuration
- Media Access Control (MAC) table
- Dot1x port security
- MAC security
- Metro Ring Protocol (MRP)
- Virtual Switch Redundancy Protocol (VSRP)
- Interface statistics

The following example shows the truncated output of the **show tech-support l2** command.

```
Brocade# show tech-support l2
VLAN Brief Summary Information :

System-max vlan Params: Max(4095) Default(64) Current(64)
Default vlan Id :1
Total Number of Vlan Configured :34
VLANs Configured :1 to 2 10 to 19 21 to 40 50 900

Private VLAN Information :
------------------------
Primary VLAN   10
 Port 1/1/7 1/1/8 1/1/9 1/1/10

Primary VLAN   30
 Port 1/1/1 1/1/2 1/1/3 1/1/4 1/1/5 1/1/6 1/1/7 1/1/8 1/1/9 1/1/20 1/1/31 1/1/32
1/1/33 1/1/34 1/1/35 1/1/36 1/1/37 1/1/38 1/1/39 1/1/40 1/1/41 1/1/42 1/1/43
1/1/44 1/1/45

Primary VLAN   40
 Port 1/1/1 1/1/2 1/1/3 1/1/4 1/1/5 1/1/6 1/1/7 1/1/8 1/1/9 1/1/20 1/1/31 1/1/32
1/1/33 1/1/34 1/1/35 1/1/36 1/1/37 1/1/38 1/1/39 1/1/40 1/1/41 1/1/42 1/1/43
1/1/44 1/1/45

Primary VLAN   50
 Port 1/1/31 1/1/32 1/1/33 1/1/34 1/1/35

Primary VLAN  900
 Port 1/1/31 1/1/32 1/1/33 1/1/34 1/1/35 1/1/36 1/1/37 1/1/38

Spanning Tree Information :
--------------------------
STP instance owned by topology-group 1, Master VLAN 1

Global RSTP (IEEE 802.1W/D3) Parameters:

VLAN Root             Root Root   Prio Max He- Ho- Fwd Last    Chg Bridge
 ID   ID              Cost Port   rity Age llo ld  dly Chang   cnt Address
                               Hex  sec sec sec sec sec
   1 8000002438c94fc0 0    Root   8000 20  2   1   15  6947    1   002438c94fc0

Port STP Parameters:
-------------------
Port   Prio Path  State       Fwd    Design  Designated       Designated
Num    rity Cost              Trans  Cost    Root             Bridge
       Hex
1/1/28 80   0     DISABLED    0      0       0000000000000000 0000000000000000
1/1/29 80   0     DISABLED    0      0       0000000000000000 0000000000000000
1/1/46 80   0     DISABLED    0      0       0000000000000000 0000000000000000

Spanning-tree is not configured on port-vlan 2
```

```
STP instance owned by VLAN 10

Global STP (IEEE 802.1D) Parameters:
-----------------------------------
VLAN Root             Root Root   Prio Max He- Ho- Fwd Last    Chg Bridge
 ID   ID             Cost Port   rity Age llo ld  dly Chang   cnt Address
                                  Hex  sec sec sec sec sec
  10 8000002438c94fc0 0    Root   8000 20  2   1   15  7077    0   002438c94fc0

Port STP Parameters:
-------------------
Port   Prio Path  State       Fwd    Design  Designated       Designated
Num    rity Cost              Trans  Cost    Root             Bridge
       Hex
1/1/7  80   0     DISABLED    0      0       0000000000000000 0000000000000000
1/1/8  80   0     DISABLED    0      0       0000000000000000 0000000000000000
1/1/9  80   0     DISABLED    0      0       0000000000000000 0000000000000000
1/1/10 80   0     DISABLED    0      0       0000000000000000 0000000000000000

802.1W Spanning Tree Information :
---------------------------------
IEEE 802-1w is not configured on port-vlan 1

--- VLAN 2 [ STP Instance owned by VLAN 2 ] ---------------------------

Bridge IEEE 802.1W Parameters:
-----------------------------
Bridge           Bridge Bridge Bridge Force    tx
Identifier       MaxAge Hello  FwdDly Version  Hold
hex              sec    sec    sec             cnt
8000002438c94fc0 20     2      15     Default  3

RootBridge       RootPath  DesignatedBri-   Root   Max Fwd Hel
Identifier       Cost      dge Identifier   Port   Age Dly lo
hex                        hex                     sec sec sec
8000002438c94fc0 0         8000002438c94fc0 Root   20  15  2

Port IEEE 802.1W Parameters:
---------------------------
       <--- Config Params --><-------------- Current state ----------------->
Port   Pri PortPath P2P Edge Role       State       Designa-  Designated
Num        Cost     Mac Port                        ted cost  bridge
1/1/6  128 0        F   F    DISABLED   DISABLED    0         0000000000000000

IEEE 802-1w is not configured on port-vlan 10

--- VLAN 11 [ STP Instance owned by VLAN 11 ] --------------------------

Bridge IEEE 802.1W Parameters:
-----------------------------
Bridge           Bridge Bridge Bridge Force    tx
Identifier       MaxAge Hello  FwdDly Version  Hold
hex              sec    sec    sec             cnt
8000002438c94fc0 20     2      15     Default  3

RootBridge       RootPath  DesignatedBri-   Root   Max Fwd Hel
Identifier       Cost      dge Identifier   Port   Age Dly lo
hex                        hex                     sec sec sec
8000002438c94fc0 0         8000002438c94fc0 Root   20  15  2
.
(output truncated)
```

show tech-support l3 ipv4-uc

Syntax: show tech-support l3 ipv4-uc

This command displays the following debugging information related to the Layer 3 IPv4 unicast configurations:

- IPv4 routing table
- IPv4 interface
- Routing Information Protocol (RIP) routes and neighbors
- Border Gateway Protocol (BGP) routes and neighbors
- Open Shortest Path First Version 2 (OSPFv2) routes and neighbors

The following example shows the truncated output of the **show tech-support l3 ipv4-uc** command.

```
Brocade# show tech-support l3 ipv4-uc
IP Cache:
--------
Total number of cache entries: 5
D:Dynamic  P:Permanent  F:Forward  U:Us  C:Complex Filter
W:Wait ARP  I:ICMP Deny  K:Drop  R:Fragment  S:Snap Encap
  IP Address   Next Hop      MAC            Type Port       Vlan Pri
1 10.1.1.5     DIRECT        0000.0000.0000 PU   n/a             0
2 10.18.243.32 DIRECT        0000.0000.0000 PU   n/a             0
3 10.20.20.4   DIRECT        0000.0000.0000 PU   n/a             0
4 10.21.21.40  DIRECT        0000.0000.0000 PU   n/a             0
5 10.255.255.255 DIRECT      0000.0000.0000 PU   n/a             0
IP Interfaces:
-------------
Interface   IP-Address      OK?  Method    Status             Protocol
Eth 7/2     10.1.1.6       YES  NVRAM     admin. down        down
Eth 7/9     10.1.1.5        YES  NVRAM     up                 up
Eth 7/10    10.20.20.4      YES  NVRAM     up                 up
Eth 7/24    10.21.21.40     YES  NVRAM     up                 up
Eth mgmt1   10.18.243.32    YES  NVRAM     up                 up
Ve 3000     10.1.1.2         YES  NVRAM     down               down
IP Routing Table:
---------------
Total number of IP routes: 5, avail: 262139 (out of max 262144)
B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
 Destination    NetMask         Gateway         Port       Cost   Type
  0.0.0.0        0.0.0.0         10.18.243.1     mgmt1      1      S
1 10.18.243.0    255.255.255.128 0.0.0.0         mgmt1      1      D
2 10.20.20.0    255.255.255.0   0.0.0.0         7/10       1      D
3 10.21.21.0     255.255.255.0   0.0.0.0         7/24       1      D
4 10.1.1.0       255.255.255.0   0.0.0.0         7/9        1      D
IP Routing Summary:
-----------------
IP Routing Table - 5 entries:
  4 connected, 1 static, 0 RIP, 0 OSPF, 0 BGP
  Number of prefixes:
  /0: 1 /24: 3 /25: 1

IP Ospf Trap :
-------------
Interface State Change Trap:                    Enabled
Virtual Interface State Change Trap:            Enabled
Neighbor State Change Trap:                     Enabled
Virtual Neighbor State Change Trap:             Enabled
```

```
Interface Configuration Error Trap:               Enabled
Virtual Interface Configuration Error Trap:       Enabled
Interface Authentication Failure Trap:            Enabled
Virtual Interface Authentication Failure Trap:    Enabled
Interface Receive Bad Packet Trap:                Enabled
Virtual Interface Receive Bad Packet Trap:        Enabled
Interface Retransmit Packet Trap:                 Enabled
Virtual Interface Retransmit Packet Trap:         Enabled
Originate LSA Trap:                               Enabled
Originate MaxAge LSA Trap:                        Enabled
Link State Database Overflow Trap:                Enabled
Link State Database Approaching Overflow Trap:    Enabled

IP Ospf Error :
--------------
--- The following are for debug. They may not mean error or warning. ---
IP Ospf Resource :
-----------------
                     alloc in-use  avail allo-fail up-limit   get-mem
IP route info        264192      6 264186         0   264192         6
OSPF/IP route union  264192      5 264187         0   264192         6
OSPF route           264192      1 264191         0   264192         1
pool1 size 40B         2000      1   1999         0                 37
pool2 size 64B         4000      1   3999         0                 34
pool3 size 160B          32      0     32         0                  0
pool4 size 260B          16      0     16         0                  0
pool5 size 516B          32      1     31         0                  1
pool6 size 1504B         32      0     32         0                  0
pool7 size 45303B        16      1     15         0                  1

             alloc   in-use  avail get-fail  limit  get-mem  size init
distribute-list intf     64      0     64        0    14848        0    45   64
simple link list         32      0     32        0     7424        0    16   32
IP Ospf Neighbor Detail :
------------------------

Port        Address         Pri State      Neigh Address   Neigh ID
            Ev Op Cnt
1/1/4         10.1.1.2        1   FULL/DR    10.1.1.1        10.1.1.1
            6  2  0    Life:39, area=0.0.0.0
IP Ospf Virtual-link :
--------------------
No ospf virtual-link entries available
IP Ospf Virtual-neighbor :
------------------------
No ospf virtual-neighbor entries available
IP RIP Routes :
-------------
 Destination Gateway Port Metric Aging-Timer Status
10.1.1.1255.255.255.01/10
IP RIP Interfaces :
-----------------
Interface Eth 7/9 :
        RIP Mode : V1
        Route summarization : Enabled
        Poison reverse : On
        Authentication : Disabled
        Number of packets received : 2068
        Number of packets transmitted : 1
        Aging timeout value : 180
        Garbage collection timeout : 120
```

```
        Update time interval : 30
        Last broadcast time : 29
.
(output truncated)
```

show tech-support l3 ipv6-uc

Syntax: show tech-support l3 ipv6-uc

This command displays the following debugging information related to the Layer 3 IPv6 unicast configurations:

- IPv6 routing table
- IPv6 interface
- IPv6 neighbor
- OSPF Version 3 (OSPFv3) neighbors

The **show tech-support l3 ipv6-uc** command output resembles the following example.

```
Brocade# show tech-support l3 ipv6-uc
IPV6 Routing Table:
-----------------
IPv6 Routing Table - 1 entries:
Type Codes:  C - Connected, S - Static, R - RIP, O - OSPF, B - BGP
OSPF Sub Type Codes:  O - Intra, Oi - Inter, O1 - Type1 external, O2 - Type2
external
Type IPv6 Prefix   Next Hop Router      Interface  Dis/Metric
C  2001:DB8::/64        ::                   e 1/1/8     0/0

IPV6 Interfaces:
-------------
Routing Protocols : R - RIP  O - OSPF
Interface      Status     Routing  Global Unicast Address
Eth 1/1/8         up/up      O        2001:DB8::1/64
IPV6 Routing Summary:
-------------------
IPv6 Routing Table - 1 entries:
  1 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP
  Number of prefixes:
  /64: 1
IPV6 Cache:
----------
Total number of cache entries: 2
    IPv6 Address               Next Hop            Port
1   2001:DB8::1                LOCAL               e 1/1/8
2   fe80::768e:f8ff:fe2d:7a00  LOCAL               e 1/1/8
IPV6 Neighbor:
-----------
Total number of Neighbor entries: 1
 IPv6 Address       LinkLayer-Addr State  Age  Port    vlan  IsR
fe80::768e:f8ff:fe2d:5c80  748e.f82d.5c80 STALE   15  e 1/1/8   -    1

IPV6 OSPF Interfaces :
--------------------
Interface  OSPF      Status State     Area
e 1/1/8    enabled   up     DR        0
IPV6 OSPF Neighbor:
------------------

Total number of neighbors in all states: 1
```

```
Number of neighbors in state Full      : 1

RouterID        Pri State    DR              BDR             Interface[State]
172.26.67.67      1 Full     172.26.67.65    172.26.67.67    e 1/1/8    [DR]
IPV6 OSPF Memory :
----------------
   Total Static Memory Allocated : 19508 bytes
   Total Dynamic Memory Allocated : 2750268 bytes
   Memory Type               Size   Allocated  Max-alloc  Alloc-Fails
   MTYPE_OSPF6_TOP           81956      1          1          0
   MTYPE_OSPF6_LSA_HDR       56         6          9          0
   MTYPE_OSPF6_RMAP_COMPILED 0          0          0          0
   MTYPE_OSPF6_OTHER         20         0          1          0
   MTYPE_THREAD_MASTER       84         1          1          0
   MTYPE_OSPF6_AREA          107631     1          4          0
   MTYPE_OSPF6_AREA_RANGE    22         0          16         0
   MTYPE_OSPF6_SUMMARY_ADDRE 25         0          16         0
   MTYPE_OSPF6_IF            274        1          64         0
   MTYPE_OSPF6_NEIGHBOR      12471      1          32         0
   MTYPE_OSPF6_ROUTE_NODE    21         3          4096       0
   MTYPE_OSPF6_ROUTE_INFO    35         3          4096       0
   MTYPE_OSPF6_PREFIX        20         0          16         0
   MTYPE_OSPF6_LSA           73         6          4096       0
   MTYPE_OSPF6_VERTEX        166        3          64         0
   MTYPE_OSPF6_SPFTREE       44         1          2          0
   MTYPE_OSPF6_NEXTHOP       28         4          256        0
   MTYPE_OSPF6_EXTERNAL_INFO 40         0          4096       0
   MTYPE_THREAD              212        8          1024       0
   MTYPE_OSPF6_LINK_LIST     20         3121       20480      0
   MTYPE_OSPF6_LINK_NODE     12         29         20480      0
IPV6 OSPF Virtual-Link:
----------------------
Index Transit Area ID      Router ID Interface Address State
1        1    10.1.1.1    2001:DB8::2 P2P

IPV6 OPSF Virtual-neighbor:
---------------------------
Index Router ID Address      State Interface
1      10.1.1.1 2001:DB8::1  Full  ethe 2/3

IPV6 RIP Route :
----------------
IPv6 RIP Routing Table is empty!IPV6 RIP Route:
----------------
IPv6 RIP Routing Table is empty!
```

show tech-support memory

Syntax: show tech-support memory

This command displays debugging information related to memory usage and tracking. Command output resembles the following example.

```
Brocade# show tech-support memory
MEMORY Related Information :
Stack unit 1:
  Total DRAM: 268435456 bytes
  Dynamic memory: 196669440 bytes total, 130510848 bytes free, 33% used

FLASH Related Information :
Stack unit 1:
```

```
  Compressed Pri Code size = 7091393,
  Compressed Sec Code size = 7063608,
  Compressed Boot-Monitor Image size = 369292, Version:07.1.00T7f5
  Code Flash Free Space = 50462720

DM-Memory Related Information  :
memory tracking has run 00:00:00
malloc 0 times, 0 B, free: #0 , 0 B, not in: 0
                     alloc in-use  avail get-fail    limit  get-mem  size init
mem-ptr pool           200      0    200        0    10000        0    12  200
stack pool             100      0    100        0     1000        0    64  100
mem-ptr hash: size=0, anch=503, pool: s=503, use=0, fail=0
stack hash: size=0, anch=209, pool: s=209, use=0, fail=0
```

show tech-support multicast

Syntax: show tech-support multicast

This command displays the following debugging information specific to Layer 2 and Layer 3 multicast configurations:

- Multicast VLAN
- Multicast table
- Multicast counters
- Internet Group Management Protocol (IGMP) group
- Protocol Independent Multicast (PIM) neighbor
- IP Multicast routing table

The **show tech-support multicast** command output resembles the following example.

```
Brocade# show tech-support multicast
IGMP Group :
----------
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
e3 : 1 groups, 1 group-port
   group    p-port   ST     QR     life mode    source     local
1 230.1.1.1 e3       no     yes    260  EX      0          YES

PIM Group :
---------
Total number of groups: 1
Index 1         Group 230.1.1.1
     Group member at  e3:  EX 0,

PIM Neighbor :
------------
Total number of neighbors: 1 on 1 ports
Port   Phy_p  Neighbor        Holdtime Age   UpTime  GenID
e5     e5     10.10.10.20  180      180   480     0x154A3D0E

PIM Cache :
---------
Total 1 entries

Example: (S G) in v40 (e2/3) cnt= : e2/3 is phy. of input v40, cnt: SW hit incl.
drop
HW: CAM switched, SW: cpu switched, OAR: SW one-arm-routing, VL: vlan
trunking: TR(e3/3,e3/4): e3/3 is primary trunk port, e3/4 is real out p.
Gre tnnl(OIF): tn5:e1/1: e1/1 is tunnel out-port
```

```
               tn5:TR(e1/1): e1/1 is primary-port of tunnel out-trunk
FLAGS: fast/slow: could be HW switched or not
       pru: pruned from upstream
       frag: packet fragmented, SW forwarded unless multicast-perf configured
       tag/tnnl:  OIF has tagged/tunnel ports
       swL2/hwL2: SW or HW L2 forwarding
       age:   send 1 pkt to cpu from HW switch to reset age
       0L2C:   no L2 CAM hash
       ClSr: has client on input port, never send prune even OIF empty
       drop: use cam to drop pkts if no OIFs.

1    (10.21.21.21 230.1.1.1) in e5 (e5), cnt=28
     upstream neighbor=10.10.10.20 on e5 using ip route
     L3 (HW) 1: e3(VL1)
     fast=1 slow=0 pru=0 graft
     age=60s up-time=1m HW=1 L2-vidx=8188 has mll

Multicast VLAN information :
-------------------------
Summary of all vlans. Please use "sh ip mu vlan vlan-id" for details
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=260

Multicast Group Table information :
---------------------------------
e3 : 1 groups, 1 group-port
  group     p-port   ST     QR     life mode    source     local
1 230.1.1.1  e3       no     yes    260  EX      0          YES
Multicast Cache information :
---------------------------
vlan 10, 0 cache
```

show tech-support packet-loss

Syntax: show tech-support packet-loss

This command displays packet statistics information that helps in debugging packet loss scenarios. Command output resembles the following example.

```
Brocade# show tech-support packet-loss
Total Tx statistics :
--------------------
PP total Tx stats since last read: First read contains counters from previous
setting.
Total Ucast = 8, Total Bcast = 14, Total Mcast = 8, Cntl pkt=10634
Total Bridge Egress Filtered = 0, Total Congestion Drops = 0, Fwd restriction=0

Port Statistics.
----------------
Port            In Packets          Out Packets       In Errors      Out Errors
1/1/1                    0                    0               0               0
1/1/2                    0                    0               0               0
1/1/3                15040                15230               0               0
1/1/4                22325                12399               0               0
1/1/5                    0                    0               0               0
1/1/6                    0                    0               0               0
1/1/7                    0                    0               0               0
1/1/8                    0                    0               0               0
1/1/9                    0                    0               0               0
1/1/10                   0                    0               0               0
1/1/11                   0                    0               0               0
1/1/12                   0                    0               0               0
```

```
1/1/13          0               0               0               0
1/1/14          0               0               0               0
1/1/15          0               0               0               0
1/1/16          0               0               0               0
1/1/17          0               0               0               0
1/1/18          0               0               0               0
1/1/19          0               0               0               0
1/1/20          0               0               0               0
1/1/21          0               0               0               0
1/1/22          0               0               0               0
1/1/23          0               0               0               0
1/1/24          0               0               0               0
1/1/25          0               0               0               0
1/1/26          0               0               0               0
1/1/27          0               0               0               0
1/1/28          0               0               0               0
1/1/29          0               0               0               0
1/1/30          0               0               0               0
```

Commands

The following commands support the features described in this chapter:

- show tech-support cluster
- show tech-support license
- show tech-support stack

show tech-support cluster

Collects and displays complete debugging information related to Multi-Chassis Trunking (MCT) cluster configuration.

Syntax **show tech-support cluster**

Parameters None

Modes Privileged EXEC mode

Command Output The **show tech-support cluster** command output includes the output of the **show cluster** and the **show cluster ccp peer detail** commands. The **show tech-support cluster** command output displays the following information:

- Cluster configuration
- Spanning Tree Protocol (STP), if configured
- Rapid Spanning Tree Protocol (RSTP), if configured
- Peer details
- Cluster Communication Protocol (CCP)
- Last reason for CCP down
- Inter-Chassis Link (ICL)
- MCT client
- Transmission Control Protocol (TCP) connection

Examples The following example displays sample output from the **show tech-support cluster** command.

```
Brocade# show tech-support cluster
Cluster Configuration information
cluster SX 3000   rbridge-id 2   session-vlan 3000   keep-alive-vlan 3001   icl
SX-MCT ethernet 1/7  peer 1.1.1.2 rbridge-id 3 icl SX-MCT  deploy   client
client-1    rbridge-id 100    client-interface ethernet 1/5
  deploy    !
Cluster SX 3000
============
Rbridge Id: 3, Session Vlan: 3000, Keep-Alive Vlan: 3001
Cluster State: Deploy
Client Isolation Mode: Loose
Member Vlan Range: 10 20 30
MCT Peer's Reachability using Keep-Alive Vlan: Peer Reachable

 ICL Info:
 ---------
 Name            Port  Trunk
 SX-MCT          7/5   4

 Peer Info:
 ----------
 Peer IP: 10.20.152.21, Peer Rbridge Id: 2, ICL: SX-MCT
 KeepAlive Interval: 10 , Hold Time: 90, Fast Failover
 Active Vlan Range:  10 20 30
 Last Reason for CCP Down: ICL VE interface down  State: CCP Up (Up Time:   0 days:
5 hr:42 min:13 sec)
```

```
 Client Info:
 ------------
 Number of Clients configured: 2
 Name                    Rbridge-id Config      LACP Port  Trunk FSM-State
 client-1                100        Deployed    no   7/7     2    Up
 client-2                200        Deployed    no   7/9     3    Up
**************Peer Session Details*********************
IP address of the peer                          10.20.152.21
Rbridge ID of the peer                          2
Session state of the peer                       OPERATIONAL
Next message ID to be send                      2426
Keep Alive interval in seconds                  10
Hold Time Out in seconds                        90
Fast Failover is enable for the session
UP Time                                          0 days: 6 hr:43 min:53 sec
Number of  tcp packet allocations failed          0
Message    Init          Keepalive     Notify        Application   Badmessages
Send       1             2425          0             52            0
Receive    1             2425          0             41            0
TCP connection is up
TCP connection is initiated by                  10.20.152.50
TCP connection tcbHandle not pending
TCP connection packets received
**************TCP Connection Details********************
    TCP Connection state: ESTABLISHED      Maximum segment size: 1436
       Local host:  10.20.152.50, Local  Port: 12019
       Remote host: 10.20.152.21, Remote Port: 4175
       ISentSeq:    3510235  SendNext:    3568780  TotUnAck:          0
       TotSent:       58545  ReTrans:           0  UnAckSeq:    3568780
       IRcvSeq:     2020934  RcvNext:     2075601  SendWnd:       16384
       TotalRcv:      54667  DupliRcv:          0  RcvWnd:        16384
       SendQue:           0  RcvQue:            0  CngstWnd:       3051
```

History

Release	Command History
Release 08.0.00a	The **show tech-support** command was enhanced to display MCT cluster configuration information.

Related Commands

show cluster, show cluster ccp peer detail

show tech-support license

Displays information about the license installed on the device.

Syntax **show tech-support license**

Parameters None

Modes Privileged EXEC mode

Command Output The **show tech-support license** command output includes the following information:

- License name
- License ID
- License type
- Status of license
- License period
- License capacity

Examples The following example displays sample output from the **show tech-support license** command.

```
Brocade# show tech-support license

License Information
===================
Index      License Name      Lid          License   Status     License  License
                                          Type                 Period  Capacity
Stack unit 1:
1          FCX-ADV-LIC-SW    dexHHHIhFJm  Normal    Active     Unlimited    1
2          FCX-ADV-LIC-SW    dexHHHGhFGr  Normal    Invalid    Unlimited    1
Stack unit 2:
1          FCX-ADV-LIC-SW    dexHHHGhFGr  Normal    Active     Unlimited    1
2          FCX-ADV-LIC-SW    dexHHHIhFJm  Normal    Invalid    Unlimited    1
```

History

Release	Command History
Release 08.0.00a	The **show tech-support** command was enhanced to display license information.

show tech-support stack

Collects and displays debugging information related to stacking configurations.

Syntax **show tech-support stack**

Parameters None

Modes Privileged EXEC mode

Command Output The **show tech-support stack** command output displays the following information related to stacking configurations:

- Stacking status
- Stack resource details
- Stack IPC statistics
- Port statistics for all stack ports
- Stack IPC
- Stack mail traffic statistics
- Stack mailbox neighbors

The command output includes the output of the following commands:

- **show stacking**
- **show stack resource**
- **show stack rel-ipc**
- **show statistics stack-ports**
- **show stack ipc**
- **dm stk show debug mail traffic**
- **dm stk show debug mail neighbors**

NOTE
For information about the show commands, refer to the *FastIron Ethernet Switch Stacking Configuration Guide*.

Examples The following example displays truncated output from the **show tech-support stack** command.

```
Brocade# show tech-support stack
Stacking Status.
alone: standalone, D: dynamic config, S: static config
ID   Type         Role    Mac Address    Pri State   Comment
1  S ICX6610-24P  active  0000.0001.0100 128 local   Ready
5  S ICX6610-24F  standby 0000.0002.0100   0 remote  Ready
6  S ICX6450-24   member  0000.00b0.4fa0   0 remote  Ready

    standby      active
     +---+        +---+
  2/6| 5 |2/1==2/1| 1 |2/6
     +---+        +---+

    active
      ---         +---+
```

```
     ( 1 )1/1--1/1| 6 |2/3
      ---         +---+

peri-ports w/o link: 1/3/1

Standby u5 - No hitless failover. Reason: hitless-failover not configured
Current stack management MAC is 0000.0001.0100
Note: no "stack mac" config. My MAC will change after failover.

Image-Auto-Copy is Disabled.

      Stack Port Status                    Neighbors
Unit# Stack-port1       Stack-port2        Stack-port1       Stack-port2
1     up (1/2/1-1/2/2)  dn (1/2/6-1/2/7)   U5 (5/2/1-5/2/2)  none
5     up (5/2/1-5/2/2)  dn (5/2/6-5/2/7)   U1 (1/2/1-1/2/2)  none
6     up (6/1/1)        up (6/2/3)         U1 (1/1/1)        none

Unit# System uptime
1     23 hours 13 minutes 33 seconds
5     23 hours 13 minutes 32 seconds
6     23 hours 14 minutes 11 seconds

Stack Resource information.
                     alloc in-use  avail get-fail    limit  get-mem  size init
register attribute   19200  13635   5565        0   556800    18707   336 2400
general 12B data        32      3     29        0     7424        3    12   32
RB-tree node         16384  13639   2745        0   237568    14112    18 1024
variable length link  3113      4   3109        0   722216        4     8 3113
AU msg dev0           4092      0   4092        0    16368        0    16 4092

Stack IPC Statistics.
Reliable IPC statistics:

Global statistics:
Pkts rcvd w/no session: 2
Msgs rcvd w/no handler: 0

Unit statistics:

Unit 5 statistics:
Msgs sent: 31650  Msgs received: 18546, Pkt sends failed: 0

Message types sent:
    [9]=31645,        [10]=3,           [11]=2,

Message types received:
    [9]=18542,        [10]=1,           [11]=3,

Session statistics: base-channel, unit 5, channel 0:
Session state: established (last established 23 hours 11 minutes 21 seconds ago)
Connections established: 1
Remote resets: 0, Reset packets sent: 0
Connection statistics (for current connection, if established):
Msgs sent: 22286, Msgs received: 18541
Atomic batches sent: 0, Atomic batches received: 0
Pkts sent: 48417, Pkts received: 46702
Msg bytes sent: 4560292, Msg bytes received: 1628850
Pkt bytes sent: 5950708, Pkt bytes received: 2596884
Flushes requested: 47, Suspends: 0, Resumes: 0
Packets sent with data (DAT), ACKs, and window updates (WND):
```

```
Other: 1897, ACK: 18332, WND: 568, ACK+WND: 8
DAT: 27452, DAT+ACK: 158, DAT+WND: 2, DAT+ACK+WND: 0
Data retransmits done: 5681, Zero-window probes sent: 0
Dup ACK pkts rcvd: 3010, Pkts rcvd w/dup data: 1
```

History

Release	Command History
Release 08.0.00a	The **show tech-support stack** command was enhanced to display additional stacking information.

Related Commands

show stacking, show stack resource, show stack rel-ipc, show statistics stack-ports, show stack ipc

Diagnostic Command Index

D

S