debug mpls ldp checkpoint through debug mwi relay events

debug mpls ldp checkpoint

Note Effective with Cisco IOS Release 12.2(33)SRA, the **debugmplsldpcheckpoint**command is replaced by the **debugmplsvpnha**command. See the **debugmplsvpnha**command for more information.

To enable the display of Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP) checkpoint debugging information, use the **debugmplsldpcheckpoint**command in privileged EXEC mode. To disable the display of MPLS LDP checkpoint debugging information, use the **no** form of this command.

debug mpls ldp checkpoint

no debug mpls ldp checkpoint

Syntax Description

This command has no arguments or keywords.

Command Default

Debugging of MPLS LDP checkpointing is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(25)S	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB and implemented on the Cisco 10000 series routers.
12.2(33)SRA	This command was replaced by the **debugmplsvpnha**command.

Usage Guidelines

The following examples show sample output from the debug mpls ldp checkpoint command:

Sample Output on the Active Route Processor or PRE

```
Router# debug mpls ldp checkpoint
LDP Checkpointing events and errors debugging is on
LDP-CF: 0:10.3.3.3/32,20:: checkpointing local binding
LDP-CF: 0:10.3.3.3/32,20:: changing checkpoint state from none to add-send
LDP-CF: 0:10.3.3.3/32,20:: changing checkpoint state from add-send to add-wait
LDP-CF: received CF send-ack
LDP-CF: 0:10.3.3.3/32,20:: changing checkpoint state from add-wait to added
```

Sample Output on the Backup Route Processor or PRE

```
Router# debug mpls ldp checkpoint
LDP-CF: received 16-byte CF message: client 28 [0], ver 1, type 1
LDP-CF: 0:10.3.3.3/32,20:: adding checkpointed local binding
```

The following table describes the significant field in the sample display.

Table 1: debug mpls ldp checkpoint Command Field Descriptions

Field	Description
0:10.3.3.3/32,20::	The table ID, prefix, prefix length, and label of the checkpointed label binding.

Related Commands

Command	Description
show mpls ldp checkpoint	Displays information about the LDP checkpoint system on the active Route Processor.

debug mpls ldp graceful-restart

To display debugging information for Multiprotocol (MPLS) Label Distribution Protocol (LDP) stateful switchover (SSO) nonstop forwarding (NSF) support and Graceful Restart, use the **debugmplsldpgraceful-restart**command in privileged EXEC mode. To disable the display of this debugging information, use the **no** form of this command.

debug mpls ldp graceful-restart

no debug mpls ldp graceful-restart

Syntax Description

This command has no arguments or keywords.

Command Default

The display of debugging information is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(29)S	This command was introduced.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB and implemented on the Cisco 10000 series routers.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Usage Guidelines

This command shows events and errors related to LDP Graceful Restart.

Examples

The following example shows sample output from the debug mpls ldp graceful-restart command. The output shows that a session was lost. The status message show the events that happen during recovery of the bindings.

```
Router# debug mpls ldp graceful-restart
LDP GR: GR session 10.110.0.10:0:: lost
LDP GR: down nbr 10.110.0.10:0:: created [1 total]
LDP GR: GR session 10.110.0.10:0:: bindings retained
LDP GR: down nbr 10.110.0.10:0:: added all 7 addresses [7 total]
LDP GR: down nbr 10.110.0.10:0:: state change (None -> Reconnect-Wait)
LDP GR: down nbr 10.110.0.10:0:: reconnect timer started [120000 msecs]
LDP GR: down nbr 10.110.0.10:0:: added to bindings task queue [1 entries]
LDP GR: searching for down nbr record (10.110.0.10:0, 10.2.0.10)
```

```
LDP GR: search for down nbr record (10.110.0.10:0, 10.2.0.10) returned 10.110.0.10:0
LDP GR: Added FT Sess TLV (Rconn 120000, Rcov 120000) to INIT msg to 10.110.0.10:0
LDP GR: Tagcon querying for up to 12 bindings update tasks
LDP GR: down nbr 10.110.0.10:0:: requesting bindings MARK for {10.110.0.10:0, 1}
LDP GR: down nbr 10.110.0.10:0:: removed from bindings task queue [0 entries]
LDP GR: Requesting 1 bindings update tasks [0 left in queue]
LDP GR: 10.1.0.0/8:: updating binding from 10.110.0.10:0, inst 1:: marking stale;
LDP GR: 10.2.0.0/16:: updating binding from 10.110.0.10:0, inst 1:: marking stale;
LDP GR: 10.0.0.14/32:: updating binding from 10.110.0.10:0, inst 1:: marking stale;
LDP GR: searching for down nbr record (10.110.0.10:0, 10.2.0.10)
LDP GR: search for down nbr record (10.110.0.10:0, 10.2.0.10) returned 10.110.0.10:0
LDP GR: Added FT Sess TLV (Rconn 120000, Rcov 120000) to INIT msg to 10.110.0.10:0
LDP GR: searching for down nbr record (10.110.0.10:0, 10.2.0.10)
LDP GR: search for down nbr record (10.110.0.10:0, 10.2.0.10) returned 10.110.0.10:0
LDP GR: Added FT Sess TLV (Rconn 120000, Rcov 120000) to INIT msg to 10.110.0.10:0
LDP GR: searching for down nbr record (10.110.0.10:0, 10.2.0.10)
LDP GR: search for down nbr record (10.110.0.10:0, 10.2.0.10) returned 10.110.0.10:0
LDP GR: Added FT Sess TLV (Rconn 120000, Rcov 120000) to INIT msg to 10.110.0.10:0
LDP GR: searching for down nbr record (10.110.0.10:0, 10.2.0.10)
LDP GR: search for down nbr record (10.110.0.10:0, 10.2.0.10) returned 10.110.0.10:0
LDP GR: Added FT Sess TLV (Rconn 120000, Rcov 120000) to INIT msg to 10.110.0.10:0
LDP GR: searching for down nbr record (10.110.0.10:0, 10.2.0.10)
LDP GR: search for down nbr record (10.110.0.10:0, 10.2.0.10) returned 10.110.0.10:0
LDP GR: Added FT Sess TLV (Rconn 120000, Rcov 120000) to INIT msg to 10.110.0.10:0
LDP GR: Received FT Sess TLV from 10.110.0.10:0  (fl 0x1, rs 0x0, rconn 120000, rcov 120000)
LDP GR: GR session 10.110.0.10:0:: allocated instance, 2
LDP GR: GR session 10.110.0.10:0:: established
LDP GR: GR session 10.110.0.10:0:: found down nbr 10.110.0.10:0
LDP GR: down nbr 10.110.0.10:0:: reconnect timer stopped
LDP GR: down nbr 10.110.0.10:0:: state change (Reconnect-Wait -> Recovering)
LDP GR: down nbr 10.110.0.10:0:: recovery timer started [120000 msecs]
%LDP-5-GR: GR session 10.110.0.10:0 (inst. 2): starting graceful recovery
%LDP-5-NBRCHG: LDP Neighbor 10.110.0.10:0 is UP
LDP GR: 10.1.0.0//8:: refreshing stale binding from 10.110.0.10:0, inst 1 -> inst 2
LDP GR: 10.43.0.0//16:: refreshing stale binding from 10.110.0.10:0, inst 1 -> inst 2
LDP GR: down nbr 10.110.0.10:0:: recovery timer expired
%LDP-5-GR: GR session 10.110.0.10:0 (inst. 2): completed graceful recovery
LDP GR: down nbr 10.110.0.10:0:: destroying record [0 left]
LDP GR: down nbr 10.110.0.10:0:: state change (Recovering -> Delete-Wait)
LDP GR: down nbr 10.110.0.10:0:: added to bindings task queue [1 entries]
LDP GR: Tagcon querying for up to 12 bindings update tasks
LDP GR: down nbr 10.110.0.10:0:: requesting bindings DEL for {10.110.0.10:0, 1}
LDP GR: down nbr 10.110.0.10:0:: removed from bindings task queue [0 entries]
LDP GR: Requesting 1 bindings update tasks [0 left in queue]
LDP GR: GR session 10.110.0.10:0:: released instance, 1
```

The debug output is formatted in three general ways.

- LDP GR: GR session 10.110.0.10:0:: found down nbr 10.110.0.10:0
- down nbr 10.110.0.10:0:: removed from bindings task queue [0 entries]
- LDP GR: 2.0.0.0/8:: updating binding from 10.110.0.10:0, inst 1:: marking stale;

The following table describes the fields for the debug command output.

Table 2: debug mpls ldp graceful-restart Command Field Descriptions

Field	Description
LDP GR	Identifies LDP Graceful Restart application
GR session 10.110.0.10:0	ID of the LDP session that is enabled for Graceful Restart.
found down nbr 10.110.0.10:0	Describes the event that is happening to that LDP session.

Field	Description
down nbr 10.110.0.10:0::	Identifies the Down Neighbor record, which logs the state of a recently lost Graceful Restart session.
removed from bindings task queue [0 entries]	Describes the event that is happening to the recently lost Graceful Restart session.
2.0.0.0/8::	Identifies the Forwarding Equivalence Class (FEC) associated with the remote label binding being modified. The FEC identifies the Label Information Base (LIB) entry.
updating binding	Lists the operation being performed on the remote label binding.
10.110.0.10:0, inst 1:: marking stale;	Identifies the LDP session during which the remote label binding was learned.

Related Commands

Command	Description
show mpls ldp graceful-restart	Displays a summary of the LDP Graceful Restart status.

debug mpls ldp igp sync

To enable the display of events related to the Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP)-Interior Gateway Protocol (IGP) Synchronization feature, use the **debugmplsldpigpsync**command in **privilegedEXEC**mode. To disable this feature, use the **no** form of this command.

debug mpls ldp igp sync [**interface** *interface*] [**peer** *acl*]

no debug mpls ldp igp sync [**interface** *interface*] [**peer** *acl*]

Syntax Description

interface *interface*	(Optional) Enables the display of MPLS LDP-IGP synchronization events for the specified interface.
peer *acl*	(Optional) Enables the display of MPLS LDP-IGP synchronization events for the specified peer access control list (ACL).

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(30)S	This command was introduced.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.
12.0(32)S	The command output was modified to display events related to the delay timer on interfaces running Open Shortest Path First (OSPF) processes, if the delay timer is configured.
12.0(32)SY	The command output was modified to display events related to synchronization on interfaces running Intermediate System-to-Intermediate System (IS-IS) processes.
12.4(12)	This command was integrated into Cisco IOS Release 12.4(12).
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Examples

The following example shows events related to MPLS LDP-IGP synchronization on interfaces running OSPF:

```
Router# debug mpls ldp igp sync
```

```
LDP-SYNC: Et0/0, OSPF 1: notify status (required, not achieved, no delay, holddown infinite)
 internal status (achieved, timer running)
LDP-SYNC: E1/0, 10.0.0.1: Adj being deleted, sync_achieved goes down
LDP-SYNC: E1/0, OSPF 1: notify status (required, not achieved, delay, holddown infinite)
LDP-SYNC: Et0/0: Session already up and interface address advertised, sync_achieved comes
up.
LDP-SYNC: Et0/0, OSPF 1: notify status (required, achieved, no delay, holddown infinite)
```

The following example shows events associated when an IS-IS instance, ISIS-1, is configured for synchronization:

```
Router# debug mpls ldp igp sync
07:59:27: LDP-SYNC: Et0/0, OSPF 1: notify status (required, not achieved, no delay, holddown
 infinite) internal status (achieved, timer running)
07:59:27: LDP-SYNC: Enqueue request req_type 0 IGP ISIS ISIS-1 interface none.
07:59:27: LDP-SYNC: ISIS ISIS-1: SYNC enabled, added to global tree, informed IGP.
07:59:27: LDP-SYNC: Enqueue request req_type 3 IGP ISIS ISIS-1 interface Et0/0.
07:59:27: LDP-SYNC: Enqueue request req_type 3 IGP ISIS ISIS-1 interface Et0/0.
07:59:27: LDP-SYNC: Et0/0, ISIS ISIS-1: Added to per-interface IGP list.
07:59:27: LDP-SYNC: Et0/0: Enabled for SYNC by IGP
07:59:27: LDP-SYNC: Et0/0, ISIS ISIS-1: notify status (required, not achieved, delay,
holddown infinite)
07:59:27: LDP-SYNC: Et0/0, ISIS ISIS-1: Ignore IGP enable-interface request: already enabled.
```

The following table describes the significant fields shown in the displays.

Table 3: debug mpls ldp igp sync Field Descriptions

Field	Description
sync_achieved	The first line of the output for an interface shows the status of the MPLS LDP-IGP Synchronization feature in relation to the status of the interface.
notify status	Notify status shows the following MPLS LDP-IGP synchronization information for each interface: • If MPLS LDP-IGP synchronization is required. • If MPLS LDP-IGP synchronization has been achieved. • If the IGP should wait for MPLS LDP-IGP synchronization to be achieved. • The length of time the IGP should wait for MPLS LDP-IGP synchronization to be achieved.
internal status	Internal status displays the LDP internal synchronization status and the state of the timer. The internal status can be achieved or not achieved. The timer state can be running or not running.

The following example shows events associated with MPLS LDP-IGP synchronization on interfaces running OSPF when you configured a delay timer:

```
Router# debug mpls ldp igp sync
```

```
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: notify status (required, not achieved, no
delay, holddown infinite) internal status (achieved, timer running)
!
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: Sync disabled by IGP. Stop delay timer
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: TAGSW subblock destroyed. Stop delay timer
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: Sync down. Stop delay timer
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: Delay notifying IGP of sync achieved for 60
seconds
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: Delay timer expired, notify IGP of sync
achieved
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: Delay timer expired but sync is no longer
required won't notify IGP of sync achieved
*Jan 3 04:38:49.571: LDP-SYNC: Et0/0, OSPF 1: Delay timer expired but sync is down won't
notify IGP of sync achieved
```

Related Commands

Command	Description
mpls ldp sync	Enables MPLS LDP-IGP synchronization on all interfaces that belong to an OSPF process or IS-IS process.
show mpls ldp igp sync	Displays the status of the MPLS LDP-IGP synchronization process.

debug mpls ldp messages

To display specific information (such as message type, source, and destination) about Label Distribution Protocol (LDP) messages sent to and received from LDP peers, use the **debugmplsldpmessages**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls ldp messages {**sent**| **received**} **[all]** [**peer-acl** *acl*]

no debug mpls ldp messages {**sent**| **received**} **[all]** [**peer-acl** *acl*]

Syntax Description

sent	Displays LDP messages sent to LDP peers permitted by the access control list (*ACL)*.
received	Displays LDP messages received from LDP peers permitted by the *ACL*.
all	(Optional) Displays all LDP messages sent to and received from LDP peers (including periodic keepalive messages) permitted by the ACL.
peer-acl *acl*	(Optional) Limits the messages displayed for LDP peers in accordance with the ACL.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(50)SY	This command was integrated into Cisco IOS Release 12.2(50)SY.

Usage Guidelines

LDP requires periodic transmission of keepalive messages. If you do not specify the **all** keyword, periodic keepalive messages are not displayed.

Examples

The following is sample output from the **debugmplsldpmessages** command:

```
Router# debug mpls ldp messages received
LDP received messages, excluding periodic Keep Alives debugging is on
Router# debug mpls ldp messages sent
LDP sent PDUs, excluding periodic Keep Alives debugging is on
ldp: Rcvd init msg from 192.168.10.1 (pp 0x0)
ldp: Sent init msg to 192.168.10.1:0 (pp 0x0)
ldp: Sent keepalive msg to 192.168.10.1:0 (pp 0x0)
ldp: Rcvd keepalive msg from 192.168.10.1:0 (pp 0x0)
ldp: Sent address msg to 192.168.10.1:0 (pp 0x610F00E0)
ldp: Sent label mapping msg to 192.168.10.1:0 (pp 0x610F00E0)
ldp: Sent label mapping msg to 192.168.10.1:0 (pp 0x610F00E0)
ldp: Sent label mapping msg to 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd address msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610F00E0)
```

The following table describes the significant fields shown in the display.

Table 4: debug mpls ldp messages Field Descriptions

Field	Description
ldp:	Identifies the source of the displayed information as LDP.
Rcvd xxx msg Sent xxx msg	Type of message received or sent.
from a.b.c.d	Host that sent the message. Used in the early stages of the opening of an LDP session, when the LDP identifier is not yet known.
from a.b.c.d:e to a.b.c.d:e	LDP identifier of the peer that sent the message or to which the message was sent.
(pp 0xnnnnnnnn)	Identifies the data structure used to represent the peer at the label distribution level. Useful for correlating debug output.

Related Commands

Command	Description
debug mpls ldp session io	Displays the contents of LDP messages sent to and received from LDP peers.

debug mpls ldp nsr

To enable the display of Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP) nonstop routing (NSR) debugging events for all NSR sessions or for a specified peer, use the **debug mpls ldp nsr** command in privileged EXEC mode. To disable the display of MPLS LDP NSR debugging information, use the **no** form of this command.

debug mpls ldp nsr [**peer-acl** *acl-name*]

no debug mpls ldp nsr [**peer-acl** *acl-name*]

Syntax Description

peer-acl *acl-name*	(Optional) Displays LDP NSR events for the specified peer access list.

Command Default

Debugging of MPLS LDP NSR events are not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.9S	This command was introduced.

Examples

The following is sample output from the **debug mpls ldp nsr** command:

```
Device# debug mpls ldp nsr
*Feb  5 22:14:55.666: LDP NSR is enabled
*Feb  5 22:14:55.666: LDP Non-Stop-Routing has been enabled
*Feb  5 22:14:55.871: LDP-CF: 0:0x2A9B99C9B8 for Serial4/0, adj_addr/xport_addr
10.2.4.4/10.4.0.1:: received standby session-up, 9, in state init-sent
*Feb  5 22:14:55.871: LDP NSR: Sess Sync Record created for peer 10.4.0.1:0, inst 2,type 1
*Feb  5 22:14:55.871: LDP NSR: Addr sync Rec added to tree for peer 10.4.0.1:0, inst 2,
msg-id 0, num-rec 3
*Feb  5 22:14:55.871: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from none
to send for msgid 0
*Feb  5 22:14:55.871: LDP NSR: Sess Sync Record created for peer 10.4.0.1:0, inst 2,type 3
*Feb  5 22:14:55.871: LDP NSR: Session Sync record deleted for peer 10.4.0.1:0, inst 2,
type 3
*Feb  5 22:14:55.871: LDP NSR: Sess Sync Record created for peer 10.4.0.1:0, inst 2,type 2
*Feb  5 22:14:55.872: LDP NSR: Rbind sync Rec added to tree for peer 10.4.0.1:0, inst 2,
msg-id 2, num-rec 9
*Feb  5 22:14:55.872: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from none
to send for msgid 2
*Feb  5 22:14:55.872: LDP NSR: Sess Sync Record created for peer 10.4.0.1:0, inst 2,type 4
*Feb  5 22:14:55.872: LDP NSR: Cap sync Rec added to tree for peer 10.4.0.1:0, inst 2,
msg-id 3, num-rec 10
*Feb  5 22:14:55.872: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from none
to send for msgid 3
*Feb  5 22:14:55.872: LDP NSR: Sess Sync Addr Msg for Peer 10.4.0.1:0, inst 2, msg_id 0,
num_records 3
*Feb  5 22:14:55.872: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from send
to wait for msgid 0
*Feb  5 22:14:55.872: LDP NSR: Peer 10.4.0.1:0 Addr Session sync sent, action 9, state wait
```

```
*Feb  5 22:14:55.872: LDP NSR: Sess Sync Rbind Msg for Peer 10.4.0.1:0, msg_id 2, num_records
 9
*Feb  5 22:14:55.872: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from send
to wait for msgid 2
*Feb  5 22:14:55.872: LDP NSR: Peer 10.4.0.1:0 Session sync sent, action 11, state wait
*Feb  5 22:14:55.872: LDP NSR: Sess Sync Cap Msg for Peer 10.4.0.1:0, msg_id 3, num_records
 10
*Feb  5 22:14:55.872: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from send
to wait for msgid 3
*Feb  5 22:14:55.872: LDP NSR: Peer 10.4.0.1:0 Session sync sent, action 12, state wait
*Feb  5 22:14:55.873: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from wait
to none for msgid 0
*Feb  5 22:14:55.873: LDP NSR: Session Sync record deleted for peer 10.4.0.1:0, inst 2,
type 1
*Feb  5 22:14:55.873: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from wait
to none for msgid 2
*Feb  5 22:14:55.873: LDP NSR: Session Sync record deleted for peer 10.4.0.1:0, inst 2,
type 2
*Feb  5 22:14:56.488: LDP NSR: Peer10.4.0.1:0, Inst 2, Changing sync_rec state from wait
to none for msgid 3
*Feb  5 22:14:56.488: LDP NSR: Session Sync record deleted for peer 10.4.0.1:0, inst 2,
type 4
*Feb  5 22:14:56.488: LDP-CF: 0:0x2A9B99C9B8 for Serial4/0, adj_addr/xport_addr
10.2.4.4/10.4.0.1:: received Session Sync Done, 13, in state session-sync
*Feb  5 22:14:56.488: LDP NSR: Active Chkpt sess_sync_done for Peer 10.4.0.1:0, inst 2,
type 11, seq 11
```

Related Commands

Command	Description
mpls ldp nsr	Enables or disables NSR for LDP sessions.

debug mpls ldp peer state-machine

To display information about state transitions for label distribution protocol (LDP) sessions, use the **debugmplsldppeerstate-machine**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls ldp peer state-machine

no debug mpls ldp peer state-machine

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.0(14)ST	This command was integrated into Cisco IOS Release 12.0(14)ST.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	This command was integrated into Cisco IOS Release 12.0(23)S.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

LDP manages peer sessions by means of two coupled state machines:

- A low-level state machine that deals with session establishment and shutdown
- A high-level state machine that deals with setting up and shutting down label advertisement

Use the **debugmplsldpsessionstate-machine** command to monitor the lower-level session state machine.

Use the **debugmplsldppeerstate-machine** command to monitor the higher-level session state machine.

Examples

The following shows sample output from the **debugmplsldppeerstate-machine** command:

```
Router# debug mpls ldp peer state-machine
tagcon: start session TCP timers for 144.0.0.44:0 (pp 0x610EEC84)
tagcon: Enqueue peer up work for 144.0.0.44:0 (pp 0x610EEC84)
tagcon: peer 144.0.0.44:0 (pp 0x610EEC84): Event unsol open
    unsol op pdg -> estab
tagcon: Send initial advertisements to peer 144.0.0.44:0
tagcon: Initial address advertisement to peer 144.0.0.44:0
tagcon: Initial label advertisement to peer 144.0.0.44:0
...
tagcon: peer 144.0.0.44:0 (pp 0x610EEC84): Event down
    estab -> destroyed
tagcon: peer 144.0.0.44:0 (pp 0x610EEC84): Event cleanup done
    destroyed -> non-ex
```

The following table describes the significant fields shown in the display.

Table 5: debug mpls ldp peer state-machine Field Descriptions

Field	Description
tagcon:	Identifies the source of the message as the label control subsystem.
a.b.c.d:e	LDP identifier of the peer for the session with the state change.
(pp 0xnnnnnnnn)	Address of the data structure used to represent the peer at the label distribution level. This address is useful for correlating debug output.
Event *E*	Event causing the state change.
s1 -> s2	State of the LDP session has changed from state s1 to state s2.

Related Commands

Command	Description
debug mpls ldp session io	Displays information about LDP messages sent to or received from LDP peers.

Command	Description
show mpls ldp neighbor	Displays the status of LDP sessions.

debug mpls ldp prev-label

To display debug information when a local label binding associated with a prefix is withdrawn and freed, use the **debugmplsldpprev-label**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls ldp prev-label [**prefix-acl** *acl* [**peer-acl** *acl*]]

no debug mpls ldp prev-label [**prefix-acl** *acl* [**peer-acl** *acl*]]

Syntax Description

prefix-acl *acl*	(Optional) Limits the displayed binding information to that allocated for prefixes permitted by a prefix access control list (ACL).
peer-acl *acl*	(Optional) Limits the displayed binding withdraw information to those Label Distribution Protocol (LDP) peers permitted by a peer ACL.

Command Default

Debugging of previous local label binding changes is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(21)ST	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.

Usage Guidelines

Use this command to monitor LDP information when a local label binding associated with a prefix is withdrawn and freed. LDP withdraws a previously advertised label before advertising a new label.

If you enter the **debugmplsldpprev-label**command without an optional keyword and argument, the command displays output for all previous label binding changes. Use the **prefix-acl***acl*or**peer-acl***acl*keywords and arguments to limit the output to prefixes defined by the respective ACLs.

Examples

The following is sample output from the **debugmplsldpprev-label** command:

```
Router# debug mpls ldp prev-label
tagcon: Changing state to WITHDRAWN for prefix=10.0.1.1, label31
tagcon: Creating prev_lbl_info for prefix=10.0.1.1, label31
```

```
tagcon: noroute hold timer expired for 10.0.1.1/255.255.255.255, tag withdrawn, seqno 47
tagcon: tibent(10.0.1.1/32): label 32 from 10.0.0.2:0 removed
tagcon: Deleting prev label info for prefix = 10.0.1.1, tag = 31
```

The following table describes the significant fields shown in the display.

Table 6: debug mpls ldp prev-label Field Descriptions

Field	Description
tagcon:	Identifies the source of the message as the label control subsystem.
Changing state to WITHDRAWN	Describes the label binding change; in this case, the label is to be withdrawn.
for prefix=10.0.1.1	The prefix (10.0.1.1) from which the local label binding is to be withdrawn and freed.
label31	The local label binding (31) that is to be withdrawn from the prefix.
tibent(10.0.1.1/32)	The hostname, network, and mask for the destination that has a label binding change.

Related Commands

Command	Description
debug mpls ldp bindings	Displays information about addresses and label bindings learned from LDP peers by means of LDP downstream unsolicited label distribution.

debug mpls ldp session io

To display the contents of Label Distribution Protocol (LDP) messages sent to and received from LDP peers, use the **debugmplsldpsessionio**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls ldp session io [all] [**peer-acl** *acl*]

no debug mpls ldp session io [all] [**peer-acl** *acl*]

Syntax Description

all	(Optional) Includes the contents of periodic keepalive messages in the displayed message output to LDP peers.
peer-acl *acl*	(Optional) Limits the displayed message output to the LDP peers permitted by the access control list (*ACL*).

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(50)SY	This command was integrated into Cisco IOS Release 12.2(50)SY.

Usage Guidelines

Displays the contents of all messages sent and received, except for periodic keepalive messages.

Examples

The following is sample output from the **debugmplsldpsessionio** command:

```
Router# debug mpls ldp session io all
LDP session I/O, including periodic Keep Alives debugging is on
Router# debug mpls ldp session io peer-acl acl1
LDP session I/O, excluding periodic Keep Alives debugging is on for peer ACL acl1
ldp: Rcvd init msg from 192.168.10.1 (pp 0x0)
ldp: LDP init msg: PDU hdr: LDP Id: 192.168.10.1:0; Msg Contents:
  0x00 0x01 0x00 0x20 0x90 0x00 0x00 0x2C 0x00 0x00 0x02 0x00 0x00 0x16 0x00 0x00
  0x10 0x21 0x05 0x00 0x00 0x0E 0x00 0x01 0x00 0xB4 0x00 0x00 0x00 0x00 0x85 0x00
  0x00 0x21 0x00 0x00
ldp: Sent init msg to 192.168.10.1:0 (pp 0x0)
ldp: LDP init msg: PDU hdr: LDP Id: 192.168.10.2:0; Msg Contents:
  0x00 0x01 0x00 0x20 0x85 0x00 0x00 0x21 0x00 0x00 0x02 0x00 0x00 0x16 0x00 0x00
  0x06 0x32 0x05 0x00 0x00 0x0E 0x00 0x01 0x00 0xB4 0x00 0x00 0x00 0x00 0x90 0x00
  0x00 0x2C 0x00 0x00
ldp: Sent keepalive msg to 192.168.10.1:0 (pp 0x0)
ldp: LDP keepalive msg: PDU hdr: LDP Id: 192.168.10.2:0; Msg Contents:
  0x00 0x01 0x00 0x0E 0x85 0x00 0x00 0x21 0x00 0x00 0x02 0x01 0x00 0x04 0x00 0x00
  0x06 0x33
ldp: Rcvd keepalive msg from 192.168.10.1:0 (pp 0x0)
ldp: LDP keepalive msg: PDU hdr: LDP Id: 192.168.10.1:0; Msg Contents:
  0x00 0x01 0x00 0x0E 0x90 0x00 0x00 0x2C 0x00 0x00 0x02 0x01 0x00 0x04 0x00 0x00
  0x10 0x22
ldp: Sent address msg to 192.168.10.1:0 (pp 0x610ECDD0)
ldp: LDP address msg: PDU hdr: LDP Id: 192.168.10.2:0; Msg Contents:
  0x00 0x01 0x00 0x34 0x85 0x00 0x00 0x21 0x00 0x00 0x03 0x00 0x00 0x2A 0x00 0x00
  0x06 0x34 0x01 0x01 0x00 0x22 0x00 0x01 0x02 0x00 0x00 0xA3 0x82 0x42 0x00 0x21
  0x82 0x4D 0x00 0x21 0x85 0x00 0x00 0x21 0x22 0x00 0x00 0x21 0x67 0x00 0x00 0x21
  0x23 0x00 0x00 0x21 0x26 0x00 0x00 0x21
ldp: Sent label mapping msg to 192.168.10.1:0 (pp 0x610ECDD0)
ldp: LDP label mapping msg: PDU hdr: LDP Id: 192.168.10.2:0; Msg Contents:
  0x00 0x01 0x00 0x22 0x85 0x00 0x00 0x21 0x00 0x00 0x04 0x00 0x00 0x18 0x00 0x00
  0x06 0x36 0x01 0x00 0x00 0x08 0x02 0x00 0x01 0x20 0xCB 0x00 0x07 0x07 0x02 0x00
  0x00 0x04 0x00 0x00 0x00 0x18
ldp: Rcvd address msg from 192.168.10.1:0 (pp 0x610ECDD0)
ldp: LDP address msg: PDU hdr: LDP Id: 192.168.10.1:0; Msg Contents:
  0x00 0x01 0x00 0x24 0x90 0x00 0x00 0x2C 0x00 0x00 0x03 0x00 0x00 0x1A 0x00 0x00
  0x10 0x23 0x01 0x01 0x00 0x12 0x00 0x01 0x90 0x00 0x00 0x2C 0x02 0x00 0x00 0xA4
  0x22 0x00 0x00 0x2C 0x2D 0x00 0x00 0x2C
ldp: Rcvd label mapping msg from 192.168.10.1:0 (pp 0x610ECDD0)
ldp: LDP label mapping msg: PDU hdr: LDP Id: 192.168.10.1:0; Msg Contents:
  0x00 0x01 0x00 0x22 0x90 0x00 0x00 0x2C 0x00 0x00 0x04 0x00 0x00 0x18 0x00 0x00
  0x10 0x24 0x01 0x00 0x00 0x08 0x02 0x00 0x01 0x20 0x90 0x00 0x00 0x2C 0x02 0x00
  0x00 0x04 0x00 0x00 0x00 0x03
```

The following table describes the significant fields shown in the display.

Table 7: debug mpls ldp session io Field Descriptions

Field	Description
ldp:	Identifies the source of the message as LDP.
Rcvd xxx msg	Indicates that a message of the specified type has been received.
from a.b.c.d	Host to which the message has been sent. Used in the early stages of the opening of an LDP session when the LDP identifier is not yet known.

Field	Description
Sent xxx msg	Indicates that a message of the specified type has been sent.
to a.b.c.d	Host to which the message has been sent. Used in the early stages of the opening of an LDP session when the LDP identifier is not yet known.
to a.b.c.d:e	LDP identifier of the peer to which the message has been sent.
(pp 0xnnnnnnnn)	Identifies the data structure used to represent the peer at the label distribution level. Useful for correlating debug output.
LDP xxx msg	Type of message that has been sent.
PDU hdr: LDP Id: a.b.c.d:e	LDP identifier of the sender included in the LDP protocol data unit (PDU) header.
Msg contents: 0xnn ... 0xnn	Contents of the message represented as a sequence of bytes.

Related Commands

Command	Description
debug mpls ldp messages	Displays specific information (such as message type, source, and destination) regarding LDP messages sent to and received from LDP peers.
debug mpls ldp session state-machine	Displays information about state transitions for LDP sessions.

debug mpls ldp session protection

To enable the display of events related to MPLS LDP Session Protection, use the **debug mpls ldp session protection**command in privileged EXEC mode. To disable this feature, use the **no** form of this command.

debug mpls ldp session protection [**peer-acl** *acl*]

no debug mpls ldp session protection [**peer-acl** *acl*]

Syntax Description

peer-acl acl	(Optional) Enables the display of events for the peers whose router IDs are listed in the access control list.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(30)S	This command was introduced.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Examples

In the following example, the display of events related to MPLS LDP Session Protection are enabled:

```
Router# debug mpls ldp session protection
```

Related Commands

Command	Description
clear mpls ldp neighbor	Forcibly resets an LDP session.
show mpls ldp neighbor	Displays the contents of the LDP.

debug mpls ldp session state-machine

To display information about state transitions for label distribution protocol (LDP) sessions, use the **debugmplsldpsessionstate-machine**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls ldp session state-machine [**peer-acl** *acl*]

no debug mpls ldp session state-machine [**peer-acl** *acl*]

Syntax Description

peer-acl *acl*	(Optional) Limits the displayed information to that for LDP peers permitted by the access control list (*acl*).

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.0(14)ST	This command was integrated into Cisco IOS Release 12.0(14)ST.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	This command was integrated into Cisco IOS Release 12.0(23)S.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.

Release	Modification
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

LDP manages peer sessions by means of two coupled-state machines:

- A low-level state machine that deals with session establishment and shutdown
- A high-level state machine that deals with setting up and shutting down label advertisement

Use the **debugmplsldpsessionstate-machine** command to monitor the lower-level session state machine.

Use the **debugmplsldppeerstate-machine** command to monitor the higher-level session state machine.

Examples

The following shows sample output from the **debugmplsldpsessionstate-machine** command:

```
Router# debug mpls ldp session state-machine
ldp: ptcl_adj:144.0.0.44(0x610EED30): Non-existent -> Role pasv
ldp: create ptcl_adj: tp = 0x610EED30, ipaddr = 144.0.0.44
ldp: ptcl_adj:144.0.0.44(0x610EED30): Event: Xport opened;
    Role pasv -> Role pasv
ldp: ptcl_adj:34.0.0.44(0x610EED30): Event: Rcv Init;
    Role pasv -> Init rcvd pasv
ldp: ptcl_adj:34.0.0.44(0x610EED30): Event: Rcv KA;
    Init rcvd pasv -> Oper
ldp: ptcl_adj:unknown(0x610EED30): Event: Xport closed;
    Oper -> Non-existent
```

The following table describes the significant fields shown in the display.

Table 8: debug mpls ldp session state-machine Field Descriptions

Field	Description
ldp:	Identifies the source of the message as LDP.
ptcl_adj:a.b.c.d	Identifies the network address of the LDP peer.
(0xnnnnnnnn)	Identifies the data structure used to represent the peer at the protocol level. Useful for correlating debug output.
Event: *E*	Event that caused the state transition.
s1 -> s2	State of the LDP session has changed from state s1 to state s2.

Related Commands

Command	Description
debug mpls ldp peer state-machine	Displays information about state transitions for LDP sessions.

debug mpls ldp targeted-neighbors

To display information about the target neighbor mechanism, use the **debugmplsldptargeted-neighbors**command in privileged EXEC mode. This mechanism establishes label distribution protocol (LDP) adjacencies to peers that are not directly adjacent, such as peers at either end of a tunnel. To disable this feature, use the no form of this command.

debug mpls ldp targeted-neighbors

no debug mpls ldp targeted-neighbors

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.0(14)ST	This command was integrated into Cisco IOS Release 12.0(14)ST.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	This command was integrated into Cisco IOS Release 12.0(23)S.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Platforms that are not directly connected may engage in LDP label distribution (for example, to support two-level labeling across an LSP tunnel).

An LDP session between nondirectly connected label switch routers (LSRs) is called a targeted session and is supported by LDP extended discovery which uses targeted Hello messages sent to specific IP addresses.

An LSR (Router 1) attempting to initiate an LDP targeted session with another LSR (Router 2) sends targeted Hello messages sent to a specific IP address of Router 2. If the configuration of Router 2 permits it to respond to targeted Hello messages from Router 1, it does so, and the LDP session can be established. In this situation, Router 1 is said to be an active LSR for the targeted session because it initiated the targeted Hello messages; Router 2 is said to be a passive LSR for the session because it responded to them.

As with LDP sessions between two directly connected LSRs, it is possible for a targeted session to be the result of multiple discovery activities which are targeted to different IP addresses for the same LSR. In addition, it is possible for both LSRs in a targeted session to be active and for both to be passive.

The debug messages enabled by debug mpls ldp targeted-neighbors report activity relating to targeted sessions.

Examples

The following shows sample output from the **debugmplsldptargeted-neighbors**command:

```
Router# debug mpls ldp targeted-neighbors
ldp-trgtnbr: 144.0.0.44 Req active
ldp-trgtnbr: 144.0.0.44 allocated
ldp-trgtnbr: 144.0.0.44 Set peer start; flags 0x0
ldp-trgtnbr: 144.0.0.44 Defer peer cleanup; cleancnt 1
ldp-trgtnbr: 144.0.0.44 Set peer finished; flags 0xF
ldp-trgtnbr: 144.0.0.44 ref count incremented to 1
ldp-trgtnbr: 144.0.0.44 Release active; ref count decremented to 0
ldp-trgtnbr: 144.0.0.44 Clear peer start; flags 0xF
ldp-trgtnbr: 144.0.0.44 Undefer cleanup start; clearcnt 0, flags 0xC
ldp-trgtnbr: 144.0.0.44 Undefer cleanup finish; clearcnt 0, flags 0x8
ldp-trgtnbr: 144.0.0.44 Clear peer finished; flags 0x8
ldp-trgtnbr: 144.0.0.44 freed
```

The following table describes the significant fields shown in the display.

Table 9: debug mpls ldp targeted-neighbors Field Descriptions

Field	Description
ldp-trgtnbr:	Identifies this as an LDP targeted neighbor debug statement.
144.0.0.44	IP address for the targeted neighbor.

Related Commands

Command	Description
show mpls ldp neighbor	Displays the status of LDP protocol sessions.

debug mpls ldp transport connections

To display information about the Transmission Control Protocol (TCP) connections used to support label distribution protocol (LDP) sessions, use the **debugmplsldptransportconnections**command in privilegedEXEC mode. To disable this feature, use the no form of this command.

debug mpls ldp transport connections [**peer-acl** *acl*] [**interface** *interface*]

no debug mpls ldp transport connections [**peer-acl** *acl*] [**interface** *interface*]

Syntax Description

peer-acl *acl*	(Optional) Limits the displayed information to that for LDP peers permitted by the access control list (*acl*).
interface *interface*	(Optional) Limits the displayed information to that for the specified interface.

Command Default

Display information about LDP TCP connection activity for all peers and all interfaces.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.0(14)ST	This command was integrated into Cisco IOS Release 12.0(14)ST.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	This command was integrated into Cisco IOS Release 12.0(23)S.

Release	Modification
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use this command to monitor LDP activity relating to the establishment of the transport (TCP) connection for LDP sessions.

When two devices establish a TCP connection for an LDP session, the device with the larger transport address plays an active role and the other plays a passive role. The active device attempts to establish a TCP connection to the well-known LDP port at the passive device. The passive device waits for the connection to the well-known port to be established.

Examples

The following shows sample output from the **debugmplsldptransportconnections** command:

```
Router#
debug mpls ldp transport connections
Debug output at active peer:
ldp: Opening listen port 646 for 144.0.0.44, 34.0.0.44
ldp: Open LDP listen TCB 0x60E105BC; lport = 646; fhost = 144.0.0.44
ldp: Add listen TCB to list; tcb 0x60E105BC; addr 144.0.0.44
ldp: Incoming ldp conn 133.0.0.33:646 <-> 144.0.0.44:11042
ldp: create ptcl_adj: tp = 0x610ECD64, ipaddr = 144.0.0.44
Debug output at passive peer:
ldp: Opening ldp conn; adj 0x60BAC33C, 144.0.0.44 <-> 133.0.0.33
ldp: ldp conn is up; adj 0x60BAC33C, 144.0.0.44:11042 <-> 133.0.0.33:646
```

The following table describes the significant fields shown in the display.

Table 10: debug mpls ldp transport connections Field Descriptions

Field	Description
ldp:	Identifies the source of the message as LDP.
adj 0xnnnnnnnn	Identifies the data structure used to represent the peer at the transport level. Useful for correlating debug output.
a.b.c.d -> p.q.r.s	Indicates a TCP connection between a.b.c.d and p.q.r.s.
a.b.c.d:x -> p.q.r.s:y	Indicates a TCP connection between a.b.c.d, port x and p.q.r.s, port y.

Related Commands

Command	Description
debug mpls ldp transport events	Prints information about the events related to the LDP peer discovery mechanism.

debug mpls ldp transport events

To display information about events related to the label distribution protocol (LDP) peer discovery mechanism, use the **debugmplsldptransportevents**command in privilegedEXEC mode. This mechanism is used to determine the devices with which you wish to establish LDP sessions. To disable this feature, use the no form of this command.

debug mpls ldp transport events [**peer-acl** *acl*] **[interface]**

no debug mpls ldp transport events [**peer-acl** *acl*] **[interface]**

Syntax Description

peer-acl *acl*	(Optional) Limits the displayed information to that for LDP peers permitted by the access control list (*acl*).
interface	(Optional) Limits the displayed information to that for the specified interface.

Command Default

Displays information about LDP discovery activity for all peers and all interfaces.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
12.0(14)ST	This command was integrated into Cisco IOS Release 12.0(14)ST.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.
12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.
12.2(2)T	This command was integrated into Cisco IOS Release 12.2(2)T.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.

Release	Modification
12.0(23)S	This command was integrated into Cisco IOS Release 12.0(23)S.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use this command to monitor LDP discovery activity.

This command might generate a great deal of output. Use the peer-acl option or interface option, or both, to limit the output to peers or interfaces of interest.

Note The command includes all of the output generated by the debug mpls ldp transport connection command.

Examples

The following shows sample output from the**debugmplsldptransportevents** command:

```
Router#
debug mpls ldp transport events
ldp: enabling ldp on Ethernet1/1/1
ldp: Set intf id: intf 0x611D684C, Ethernet1/1/1, not lc-atm, intf_id 0
ldp: Set intf id: intf 0x617C5638, ATM0/0.2, not lc-atm, intf_id 0
ldp: Send ldp hello; ATM3/0.1, src/dst 8.1.1.1/224.0.0.2, inst_id 1, tcatm
ldp: Rcvd ldp hello; ATM3/0.1, from 203.0.7.7 (203.0.7.7:2), intf_id 1, opt 0x8, tcatm
ldp: Send ldp hello; Ethernet1/1/1, src/dst 138.1.0.88/224.0.0.2, inst_id 0
ldp: Rcvd ldp hello; Ethernet1/1/1, from 10.105.0.9 (7.1.1.1:0), intf_id 0, opt 0xC
ldp: ldp Hello from 10.105.0.9 (7.1.1.1:0) to 224.0.0.2, opt 0xC
ldp: New adj 0x617C5EBC from 10.105.0.9 (7.1.1.1:0), Ethernet1/1/1
ldp: Opening ldp conn; adj 0x617C5EBC, 8.1.1.1 <-> 7.1.1.1
ldp: ldp conn is up; adj 0x617C5EBC, 8.1.1.1:11013 <-> 7.1.1.1:646
ldp: Send ldp hello; ATM3/0.1, src/dst 8.1.1.1/224.0.0.2, inst_id 1, tcatm
ldp: Rcvd ldp hello; ATM3/0.1, from 203.0.7.7 (203.0.7.7:2), intf_id 1, opt 0x8, tcatm
ldp: Send ldp hello; Ethernet1/1/1, src/dst 138.1.0.88/224.0.0.2, inst_id 0
ldp: Rcvd ldp hello; Ethernet1/1/1, from 10.105.0.9 (7.1.1.1:0), intf_id 0, opt 0xC
...
ldp: Send ldp hello; Ethernet1/1/1, src/dst 138.1.0.88/224.0.0.2, inst_id 0
ldp: Send ldp hello; ATM3/0.1, src/dst 8.1no tag ip
.0.2, inst_id 1, tcatm
ldp: disabling ldp on Ethernet1/1/1
ldp: Hold timer expired for adj 0x617C5EBC, will close conn
ldp: Closing ldp conn 8.1.1.1:11013 <-> 7.1.1.1:646, adj 0x617C5EBC
ldp: Adjacency 0x617C5EBC, 10.105.0.9 timed out
ldp: Adj 0x617C5EBC; state set to closed
ldp: Rcvd ldp hello; ATM3/0.1, from 203.0.7.7 (203.0.7.7:2), intf_id 1, opt 0x8, tcatm
ldp: Ignore Hello from 10.105.0.9, Ethernet1/1/1; no intf
```

The following table describes the significant fields shown in the display.

Table 11: debug mpls ldp transport events Field Descriptions

Field	Description
ldp:	Identifies the source of the message as LDP.

Field	Description
adj 0xnnnnnnnn	Identifies the data structure used to represent the peer at the transport level. Useful for correlating debug output.
a.b.c.d (p.q.r.s:n)	Network address and LDP identifier of the peer.
intf_id	Interface identifier (non-zero for LC-ATM interfaces; 0 otherwise).
opt 0xn	Bits that describe options in the LDP discovery Hello packet: • 0x1--Targeted Hello option • 0x2--Send targeted Hello option • 0x4--Transport address option • 0x8--LDP Hello message (as opposed to TDP Hello message)

Related Commands

Command	Description
debug mpls ldp transport connections	Displays information about the TCP connections used to support LDP sessions.
show mpls ldp discovery	Displays the status of the LDP discovery process.

debug mpls lfib cef

To print detailed information about label rewrites being created, resolved, and deactivated as Cisco Express Forwarding (CEF) routes are added, changed, or removed, use the **debugmplslfibcef** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls lfib cef

no debug mpls lfib cef

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.1(3)T	This command was modified to reflect new MPLS IETF terminology and CLI syntax.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Several lines of output are produced for each route placed into the label-forwarding information base (LFIB). If your router has thousands of labeled routes, be careful about issuing this command. When label switching is first enabled, each of these routes is placed into the LFIB, and several lines of output are displayed for each route.

Examples

The following is sample output from the **debugmplslfibcef**command**:**

```
Router# debug mpls lfib cef
Cisco Express Forwarding related TFIB services debugging is on
tagcon: tc_ip_rtlookup fail on 10.0.0.0/8:subnet_lookup failed
TFIB: route tag chg 10.7.0.7/32,idx=1,inc=Withdrn,outg=Withdrn,enabled=0x2
TFIB: fib complete delete: prefix=10.7.0.7/32,inc tag=26,delete_info=1
TFIB: deactivate tag rew for 10.7.0.7/32,index=0
TFIB: set fib rew: pfx 10.7.0.7/32,index=0,add=0,tag_rew->adj=Ethernet2/3
TFIB: resolve tag rew,prefix=10.7.0.7/32,no tag_info,no parent
TFIB: fib scanner start:needed:1,unres:0,mac:0,loadinfo:0
TFIB: resolve tag rew,prefix=10.7.0.7/32,no tag_info,no parent
TFIB: fib upd loadinf 10.100.100.100/32,tag=Tun_hd,fib no loadin,tfib no loadin
TFIB: fib check cleanup for 10.100.100.100/32,index=0,return_value=0
TFIB: fib_scanner_end
TFIB: create dynamic entry for 10.11.0.11/32
TFIB: call find_route_tags,dist_method=1,next_hop=10.93.0.11,Et2/3
```

```
TFIB: route tag chg 10.11.0.11/32,idx=0,inc=26,outg=Unkn,enabled=0x3
TFIB: create tag info 10.11.0.11/32,inc tag=26,has no info
TFIB: resolve tag rew,prefix=10.11.0.11/32,has tag_info,no parent
TFIB: finish fib res 10.11.0.11/32:index 0,parent outg tag no parent
TFIB: fib upd loadinf 10.11.0.11/32,tag=26,fib no loadin,tfib no loadin
TFIB: set fib rew: pfx 10.11.0.11/32,index=0,add=1,tag_rew->adj=Ethernet2/3
tagcon: route_tag_change for: 10.250.0.97/32
        intag 33, outtag 28, nexthop tsr 10.11.0.11:0
TFIB: route tag chg 10.250.0.97/32,idx=0,inc=33,outg=28,enabled=0x3
TFIB: deactivate tag rew for 10.250.0.97/32,index=0
TFIB: set fib rew: pfx 10.250.0.97/32,index=0,add=0,tag_rew->adj=Ethernet2/3
TFIB: create tag info 10.250.0.97/32,inc tag=33,has old info
On VIP:
TFIB: route tag chg 10.13.72.13/32,idx=0,inc=34,outg=Withdrn,enabled=0x3
TFIB: deactivate tag rew for 10.13.72.13/32,index=0
TFIB: set fib rew: pfx 10.13.72.13/32,index=0,add=0,tag_rew->adj=
TFIB: create tag info 10.13.72.13/32,inc tag=34,has old info
TFIB: resolve tag rew,prefix=10.13.72.13/32,has tag_info,no parent
TFIB: finish fib res 10.13.72.13/32:index 0,parent outg tag no parent
TFIB: set fib rew: pfx 10.100.100.100/32,index=0,add=0,tag_rew->adj=
TFIB: create tag info 10.100.100.100/32,inc tag=37,has old info
TFIB: resolve tag rew,prefix=10.100.100.100/32,has tag_info,no parent
TFIB: finish fib res 10.100.100.100/32:index 0,parent outg tag no parent
TFIB: fib upd loadinf 10.100.100.100/32,tag=37,fib no loadin,tfib no loadin
```

The following table lists the significant fields and a description of special labels that appear in the output of this **debug** command shown in the display.

Table 12: debug mpls lfib cef Field Descriptions

Field	Description
tagcon	The name of the subsystem issuing the debug output (Label Control).
LFIB	The name of the subsystem issuing the debug output.
tc_ip_rtlookup fail on x.y.w.z/m: subnet_lookup failed	The destination with IP address and mask shown is not in the routing table.
route tag chg x.y.w.z/m	Request to create the LFIB entry for the specified prefix/mask.
idx=-1	The index within the FIB entry of the path whose LFIB entry is being created. The parameter -1 means all paths for this FIB entry.
inc=s	Incoming label of the entry being processed.
outg=s	Outgoing label of the entry being processed.
enabled=0xn	Bit mask indicating the types of label switching currently enabled: • 0x1 = dynamic • 0x2 = TSP tunnels • 0x3 = both

Field	Description
fib complete delete	Indicates that the FIB entry is being deleted.
prefix=x.y.w.z/m	A destination prefix.
delete_info=1	Indicates that label_info is also being deleted.
deactivate tag rew for x.y.w.z/m	Indicates that label rewrite for specified prefix is being deleted.
index=n	Index of path in the FIB entry being processed.
set fib rew: pfx x.y.w.z/m	Indicates that label rewrite is being installed or deleted from the FIB entry for the specified destination for label imposition purposes.
add=0	Indicates that label rewrite is being deleted from the FIB (no longer imposing labels).
tag_rew->adj=s	Adjacency of label rewrite for label imposition.
resolve tag rew,prefix=x.y.w.z/m	Indicates that the FIB route to the specified prefix is being resolved.
no tag_info	Indicates that there is no label_info for the destination (destination not labeled).
no parent	Indicates that the route is not recursive.
fib scanner start	Indicates that the periodic scan of the FIB has started.
needed:1	Indicates that the LFIB needs the FIB to be scanned.
unres:n	Indicates the number of unresolved TFIB entries.
mac:n	Indicates the number of TFIB entries missing MAC strings.
loadinfo:n	Indicates whether the nonrecursive accounting state has changed and whether the loadinfo information in the LFIB needs to be adjusted.
fib upd loadinf x.y.w.z/m	Indicates that a check for nonrecursive accounting is being made and that the LFIB loadinfo information for the specified prefix is being updated.
tag=s	Incoming label of entry.
fib no loadin	Indicates that the corresponding FIB entry has no loadinfo.

Field	Description
tfib no loadin	Indicates that the LFIB entry has no loadinfo.
fib check cleanup for x.y.w.z/m	Indicates that a check is being made on the LFIB entry for the specified destination to determine if rewrite needs to be removed from the LFIB.
return_value=x	If x is 0, indicates that no change has occurred in the LFIB entry. If x is 1, there was a change.
fib_scanner_end	Indicates that the FIB scan has come to an end.
create dynamic entry for x.y.w.z/m	Indicates that the LFIB has been enabled and that an LFIB entry is being created for the specified destination.
call find_route_tags	Indicates that the labels for that destination are being requested.
dist_method=n	Identifies the label distribution method--TDP, TC-ATM, and so on.
next_hop=x.y.z.w	Identifies the next hop for the destination.
interface name	Identifies the outgoing interface for the destination.
create tag info	Indicates that a label_info data structure is being created for the destination.
has no info	Indicates that the destination does not already have label_info.
finish fib re x.y.z.w/m	Indicates that the LFIB entry for the specified route is being completed.
parent outg tag s	If recursive, specifies the outgoing label of the route through which it is recursive (the parent). If not recursive, s = “no parent.”
tagcon: route_tag_change for: x.y.z.w/m	Indicates that label control is notifying LFIB that labels are available for the specified destination.
intag s	Identifies the incoming label for the destination.
outtag s	Identifies the outgoing label for the destination.
nexthop tsr x.y.z.w.i	Identifies the TDP ID of the next hop that sent the tag.

Related Commands

Command	Description
debug mpls lfib lsp	Prints detailed information about label rewrites being created and deleted as LSP tunnels are added or removed.
debug mpls lfib state	Traces what happens when label switching is enabled or disabled.
debug mpls lfib struct	Traces the allocation and freeing of LFIB-related data structures, including the LFIB itself, label rewrites, and label_info data.

debug mpls lfib enc

To print detailed information about label encapsulations while label rewrites are created or updated and placed in the label-forwarding information base (LFIB), use the **debugmplslfibenc** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls lfib enc

no debug mpls lfib enc

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.1(3)T	This command was modified to reflect new MPLS IETF terminology and CLI syntax.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Several lines of output are produced for each route placed into the LFIB. If your router has thousands of labeled routes, issue this command with care. When label switching is first enabled, each of these routes is placed into the LFIB and a label encapsulation is created. The command output shows you on which adjacency the label rewrite is being created and the labels assigned.

Examples

The following is sample output from the **debugmplslfibenc** command. This example shows the encapsulations for three routes that have been created and placed into the LFIB.

```
Router# debug mpls lfib enc
TFIB: finish res:inc tag=28,outg=Imp_null,next_hop=10.93.72.13,Ethernet4/0/3
TFIB: update_mac, mac_length = 14,addr=10.93.72.13,idb=Ethernet4/0/3
TFIB: get ip adj: addr=10.93.72.13,is_p2p=0,fibidb=Ethernet4/0/3,linktype=7
TFIB: get tag adj: addr=10.93.72.13,is_p2p=0,fibidb=Ethernet4/0/3,linktype=79
TFIB: encaps:inc=28,outg=Imp_null,idb:Ethernet4/0/3,sizes 14,14,1504,type 0
TFIB: finish res:inc tag=30,outg=27,next_hop=10.93.72.13,Ethernet4/0/3
TFIB: get ip adj: addr=10.93.72.13,is_p2p=0,fibidb=Ethernet4/0/3,linktype=7
TFIB: get tag adj: addr=10.93.72.13,is_p2p=0,fibidb=Ethernet4/0/3,linktype=79
TFIB: encaps:inc=30,outg=27,idb:Ethernet4/0/3,sizes 14,18,1500,type 0
TFIB: finish res:inc tag=30,outg=10,next_hop=0.0.0.0,ATM0/0.1
TFIB: get ip adj: addr=0.0.0.0,is_p2p=1,fibidb=ATM0/0.1,linktype=7
TFIB: get tag adj: addr=0.0.0.0,is_p2p=1,fibidb=ATM0/0.1,linktype=79
TFIB: encaps:inc=30,outg=10,idb:ATM0/0,sizes 4,8,4470,type 1
```

The following table describes the significant fields shown in the display.

Table 13: debug mpls lfib enc Field Descriptions

Field	Description
TFIB	Identifies the source of the message as the LFIB subsystem.
finish res	Identifies that the LFIB resolution is being finished.
inc tag=x or inc=x	An incoming (local) label for the LFIB entry is being created. Labels can be numbers or special values.
outg=y	An outgoing (remote) label for the LFIB entry is being created.
next_hop=a.b.c.d	IP address of the next hop for the destination.
interface	The outgoing interface through which a packet will be sent.
get ip adj	Identifies that the IP adjacency to use in the LFIB entry is being determined.
get tag adj	Identifies that the label switching adjacency to use for the LFIB entry is being determined.
addr = a.b.c.d	The IP address of the adjacency.
is_p2p=x	If x is 1, this is a point-to-point adjacency. If x is 0, it is not.
fibidb = s	Indicates the interface of the adjacency.
linktype = x	The link type of the adjacency, as follows: • 7 = LINK_IP • 79 = LINK_TAG
sizes x,y,z	Indicates the following values: • x = length of macstring • y = length of tag encapsulation • z = tag MTU

Field	Description
type = x	Tag encapsulation type, as follows: • 0 = normal • 1 = TCATM • 2 = TSP tunnel
idb:s	Indicates the outgoing interface.
update_mac	Indicates that the macstring of the adjacency is being updated.

The following table describes the special labels, which sometimes appear in the debug output, and their meanings.

Table 14: Special Labels Appearing in debug Command Output

Special Label	Meaning
Unassn--Inital value	No label assigned yet.
Unused	This destination does not have a label (for example, a BGP route).
Withdrn	The label for this destination has been withdrawn.
Unkn	This destination should have a label, but it is not yet known.
Get_res	A recursive route that will get a label when resolved.
Exp_null	Explicit null label--used over TC-ATM.
Imp_null	Implicit null label--for directly connected routes.
Tun_hd	Identifies head of TSP tunnel.

Related Commands

Command	Description
debug mpls lfib cef	Prints detailed information about label rewrites being created, resolved, and deactivated as CEF routes are added, changed, or removed.

Command	Description
debug mpls lfib lsp	Prints detailed information about label rewrites being created and deleted as LSP tunnels are added or removed.
debug mpls lfib state	Traces what happens when label switching is enabled or disabled.
debug mpls lfib struct	Traces the allocation and freeing of LFIB-related data structures, including the LFIB itself, label rewrites, and label_info data.

debug mpls lfib fast-reroute database

To enable debugging information about changes to the fast reroute database, use the debug mpls lfib fast-reroute database command in privileged EXEC command. To disable debugging output, use to no form of this command.

debug mpls lfib fast-reroute database

no debug mpls lfib fast-reroute database

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(10)ST	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Because using debug commands can use a great deal of bandwidth on your system, use caution when enabling the debug mpls lfib fast-reroute database command.

Examples

The following example displays debugging output for in Fast Reroute processing:

```
Router# debug mpls lfib fast-reroute database
LFIB-FRR:Clear headend FRR info for Tunnel1
LFIB-FRR:FRR info for Tunnel1 changed
LFIB-FRR:update headend FRR info for 10.8.0.1/32
LFIB-FRR:item B13D94 [Tu1] (group PO0/0->Tu4000):destroying entry for 10.8.0.1/32... [514
left]
LFIB-FRR:item B13D94 [Tu1]:removed from name tree
LFIB-FRR:item B13D94 [Tu1]:removed from group PO0/0->Tu4000 tree
%LINK-5-CHANGED:Interface Tunnel1, changed state to administratively down
LFIB-FRR:Clear headend FRR info for Tunnel1
LFIB-FRR:FRR info for Tunnel1 changed
LFIB-FRR:Clear headend FRR info for Tunnel1
LFIB-FRR:FRR info for Tunnel1 changed
LFIB-FRR:Set headend FRR info for Tunnel1 {main=PO0/0,backup=Tu4000,label=18}
LFIB-FRR:FRR info for Tunnel1 changed
%SYS-5-CONFIG_I:Configured from console by console
LFIB-FRR:update headend FRR info for 10.8.0.1/32
LFIB-FRR:item B13D94 [Tu1]:inserted in name tree
LFIB-FRR:item B13D94 [Tu1]:inserted in group PO0/0->Tu4000 tree
LFIB-FRR:item B13D94 [Tu1] (group PO0/0->Tu4000):full entry created for 10.8.0.1/32 [total
 515]
LFIB-FRR:update headend FRR info for 10.8.0.1/32
```

```
LFIB-FRR:item B13D94 [Tu1] (group PO0/0->Tu4000):updating entry for 10.8.0.1/32...
LFIB-FRR:item B13D94 [Tu1] (group PO0/0->Tu4000):... updated
%LINK-3-UPDOWN:Interface Tunnel1, changed state to up
LFIB-FRR:update headend FRR info for 10.43.0.0/16
LFIB-FRR:item B04C2C [Tu486]:inserted in name tree
LFIB-FRR:item B04C2C [Tu486]:inserted in group PO0/0->Tu4000 tree
LFIB-FRR:item B04C2C [Tu486] (group PO0/0->Tu4000):full entry created for 10.43.0.0/16
[total 516]
LFIB-FRR:update headend FRR info for 10.43.0.0/16
LFIB-FRR:item B04BB4 [Tu481]:inserted in name tree
LFIB-FRR:item B04BB4 [Tu481]:inserted in group PO0/0->Tu4000 tree
LFIB-FRR:item B04BB4 [Tu481] (group PO0/0->Tu4000):full entry created for 10.43.0.0/16
[total 517]
LFIB-FRR:update headend FRR info for 10.2.0.0/16
LFIB-FRR:item B04B3C [Tu486]:inserted in name tree
LFIB-FRR:item B04B3C [Tu486]:inserted in group PO0/0->Tu4000 tree
LFIB-FRR:item B04B3C [Tu486] (group PO0/0->Tu4000):full entry created for 10.2.0.0/16 [total
 518]
LFIB-FRR:update headend FRR info for 10.2.0.0/16
LFIB-FRR:item B04AC4 [Tu481]:inserted in name tree
LFIB-FRR:item B04AC4 [Tu481]:inserted in group PO0/0->Tu4000 tree
```

Related Commands

Command	Description
debug mpls traffic-eng tunnels fast-reroute events	Displays debugging information about fast reroute events.
debug mpls traffic-eng tunnels fast-reroute reroutes	Displays debugging information about the rerouting of traffic from link-protected interfaces to backup tunnels.

debug mpls lfib fast-reroute events

To display debugging information about fast reroute events, use the debug mpls lfib fast-reroute events command in privileged EXEC command. To disable debugging output, use to no form of this command.

debug mpls lfib fast-reroute events

no debug mpls lfib fast-reroute events

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(10)ST	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Because using debug commands can use a great deal of bandwidth on your system, use caution when enabling the debug mpls lfib fast-reroute events command.

Examples

The following example reports on the rerouting of traffic to a backup tunnel because of a change of state at a link-protected physical interface.

```
Router# debug mpls lfib fast-reroute events
LFIB-FRR:enqueued interface DOWN event for PO0/0 (Up)
LFIB-FRR:discarded interface DOWN event for PO0/0 (Up)
LFIB-FRR:processing interface DOWN event for PO0/0 (Up)
LFIB-FRR:group PO0/0->Tu4000:output if fixup:Backup(Tu4000) -> Backup(Tu4000)
```

Note The state given in parentheses reflects what the FRR database currently understands to be the state of the physical interface. This may or may not be the same as the event state reported earlier on that same display line.

Related Commands

Command	Description
debug mpls traffic-eng tunnels fast-reroute database	Displays debugging information about changes to the fast reroute database.

Command	Description
debug mpls traffic-eng tunnels fast-reroute reroutes	Displays debugging information about the rerouting of traffic from link-protected interfaces to backup tunnels.

debug mpls lfib fast-reroute reroutes

To enable debugging information about the rerouting of protected Label Forwarding Information Base (LFIB) entries between the primary and backup outgoing interfaces, use the debug mpls lfib fast-reroute reroutes command in privileged EXEC command. To disable debugging output, use to no form of this command.

debug mpls lfib fast-reroute reroutes

no debug mpls lfib fast-reroute reroutes

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(10)ST	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Because using debug commands can use a great deal of bandwidth on your system, use caution when enabling the debug mpls lfib fast-reroute reroutes command. The output of this command increases in proportion to the number of tunnels that utilize fast reroute.

Examples

The following example reports the results of reroute attempts:

```
Router# debug mpls lfib fast-reroute reroutes
LFIB-FRR:item B0E844 [Tu139]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0E8BC [Tu138]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0E934 [Tu387]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0E9AC [Tu137]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0EA24 [Tu136]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0EA9C [Tu135]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0EB14 [Tu384]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0EB8C [Tu134]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0EC04 [Tu133]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
LFIB-FRR:item B0EC7C [Tu132]:output if fixup, Main(PO0/0) -> Backup(Tu4000), succeeded
```

Related Commands

Command	Description
debug mpls traffic-eng tunnels fast-reroute database	Displays debugging information about changes to the fast reroute database.

Command	Description
debug mpls traffic-eng tunnels fast-reroute events	Displays debugging information about fast reroute events.

debug mpls lfib lsp

To print detailed information about label rewrites being created and deleted as label-switched path (LSP) tunnels are added or removed, use the **debugmplslfiblsp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls lfib lsp

no debug mpls lfib lsp

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.1(3)T	This command was modified to reflect new MPLS IETF terminology and CLI syntax.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following is sample output from the **debugmplslfiblsp** command:

```
Router# debug mpls lfib lsp
TSP-tunnel related TFIB services debugging is on
TFIB: tagtun,next hop=10.93.72.13,inc=35,outg=1,idb=Et4/0/3
TFIB: tsptunnel:next hop=10.93.72.13,inc=35,outg=Imp_null,if_number=7
TFIB: tsptun update loadinfo:tag=35,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun tag chg linec,fiblc=0,in tg=35,o tg=1,if=7,nh=10.93.72.13
TFIB: tagtun,next hop=10.92.0.7,inc=36,outg=1,idb=Et4/0/2
TFIB: tsptunnel:next hop=10.92.0.7,inc=36,outg=Imp_null,if_number=6
TFIB: tsptun update loadinfo:tag=36,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun tag chg linec,fiblc=0,in tg=36,o tg=1,if=6,nh=10.92.0.7
TFIB: tagtun_delete, inc = 36
tagtun tag del linec,itag=12
TFIB: tagtun_delete, inc = 35
tagtun tag del linec,itag=12
TFIB: tagtun,next hop=10.92.0.7,inc=35,outg=1,idb=Et4/0/2
TFIB: tsptunnel:next hop=10.92.0.7,inc=35,outg=Imp_null,if_number=6
TFIB: tsptun update loadinfo:tag=35,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun tag chg linec,fiblc=0,in tg=35,o tg=1,if=6,nh=10.92.0.7
On VIP:
TFIB: tagtun chg msg,in tg=35,o tg=1,nh=10.93.72.13,if=7
TFIB: tsptunnel:next hop=10.93.72.13,inc=35,outg=Imp_null,if_number=7
TFIB: tsptun update loadinfo:tag=35,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun chg msg,in tg=36,o tg=1,nh=10.92.0.7,if=6
TFIB: tsptunnel:next hop=10.92.0.7,inc=36,outg=Imp_null,if_number=6
```

```
TFIB: tsptun update loadinfo:tag=36,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun chg msg,in tg=35,o tg=1,nh=10.93.72.13,if=7
TFIB: tsptunnel:next hop=10.93.72.13,inc=35,outg=Imp_null,if_number=7
TFIB: tsptun update loadinfo:tag=35,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun chg msg,in tg=36,o tg=1,nh=10.92.0.7,if=6
TFIB: tsptunnel:next hop=10.92.0.7,inc=36,outg=Imp_null,if_number=6
TFIB: tsptun update loadinfo:tag=36,loadinfo_reqd=0,no new loadinfo,no old loadinfo
TFIB: tagtun chg msg,in tg=35,o tg=1,nh=10.92.0.7,if=6
TFIB: tsptunnel:next hop=10.92.0.7,inc=35,outg=Imp_null,if_number=6
TFIB: tsptun update loadinfo:tag=35,loadinfo_reqd=0,no new loadinfo,no old loadinfo
```

The following table describes the significant fields shown in the sample display.

Table 15: debug mpls lfib lsp Field Descriptions

Field	Description
tagtun	Name of routine entered.
next hop=x.y.z.w	Next hop for the tunnel being created.
inc=x	Incoming label for this hop of the tunnel being created.
outg=x	Outgoing label (1 means Implicit Null label).
idb=s	Outgoing interface for the tunnel being created.
if_number=7	Interface number of the outgoing interface.
tsptunnel	Name of the routine entered.
tsptun update loadinfo	The procedure being performed.
tag=x	Incoming label of the LFIB slot whose loadinfo is being updated.
loadinfo_reqd=x	Indicates whether a loadinfo is expected for this entry (non-recursive accounting is on).
no new loadinfo	No change required in loadinfo.
no old loadinfo	No previous loadinfo available.
tagtun tag chg linec	Line card is being informed of the TSP tunnel.
fiblc=x	Indicates which line card is being informed (0 means all).
in tg=x	Indicates the incoming label of new TSP tunnel.
o tg=x	Indicates the outgoing label of new TSP tunnel.
if=x	Indicates the outgoing interface number.

Field	Description
nh=x.y.w.z	Indicates the next hop IP address.
tagtun_delete	Indicates that a procedure is being performed: delete a TSP tunnel.
tagtun tag del linec	Informs the line card of the TSP tunnel deletion.
tagtun chg msg	Indicates that the line card has received a message to create a TSP tunnel.

Related Commands

Command	Description
debug mpls lfib cef	Prints detailed information about label rewrites being created, resolved, and deactivated as CEF routes are added, changed, or removed.
debug mpls lfib state	Traces what happens when label switching is enabled or disabled.
debug mpls lfib struct	Traces the allocation and freeing of LFIB-related data structures, including the LFIB itself, label rewrites, and label_info data.

debug mpls lfib state

To trace what happens when label switching is enabled or disabled, use the **debugmplslfibstate**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls lfib state

no debug mpls lfib state

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.1(3)T	This command was modified to reflect new MPLS IETF terminology and CLI syntax.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use this command when you wish to trace what happens to the label-forwarding information base (LFIB) when you issue the **mplsip** or the **mplstsp-tunnel** command.

Examples

The following is sample output from the **debugmplslfibstate** command:

```
Router# debug mpls lfib state
TFIB enable/disable state debugging is on
TFIB: Upd tag sb 6(status:0xC1,tmtu:1500,VPI:1-1 VC=0/32,et:0/0/0),lc 0x0
TFIB: intf status chg: idb=Et4/0/2,status=0xC1,oldstatus=0xC3
TFIB: interface dyntag change,change in state to Ethernet4/0/2
TFIB: enable entered, table exists,enabler type=0x2
TFIB: enable, TFIB already enabled, types now 0x3,returning
TFIB: enable entered, table exists,enabler type=0x1
TFIB: disable entered, table exists,type=0x1
TFIB: cleanup: tfib[32] still non-0
On linecard only:
TFIB: disable lc msg recvd, type=0x1
TFIB: Ethernet4/0/1 fibidb subblock message received
TFIB: enable lc msg recvd, type=0x1
TFIB: Tunnel301 set encapfix to 0x6016A97C
```

The following table describes the significant fields shown in the display.

Table 16: debug mpls lfib state Field Descriptions

Field	Description
LFIB	Identifies the source of the message as the LFIB subsystem.
Upd tag sb x	Indicates that the status of the "xth" label switching sub-block is being updated, where x is the interface number. There is a label switching sub-block for each interface on which label switching has been enabled.
(status:0xC1,tmtu:1500,VPI:1-1VC=0/32, et:0/0/0),lc 0x0)	Identifies the values of the fields in the label switching sub-block, as follows: • status byte • maximum transmission unit (*tmtu*) • range of ATM VPs • control VP • control VC (if this is a TC-ATM interface) • encapsulation type (*et*) • encapsulation information • tunnel interface number (*lc*) • line card number to which the update message is being sent (0 means all line cards)
intf status chg	Indicates that there was an interface status change.
idb=Et4/0/2	Identifies the interface whose status changed.
status=0xC1	Indicates the new status bits in the label switching sub-block of the idb.
oldstatus=0xC3	Indicates the old status bits before the change.
interface dyntag change, change in state to Ethernet4/0/2	Indicates that there was a change in the dynamic label status for the particular interface.
enable entered	Indicates that the code that enables the LFIB was invoked.
TFIB already enabled	Indicates that the LFIB was already enabled when this call was made.
table exists	Indicates that an LFIB table had already been allocated in a previous call.

Field	Description
cleanup: tfib[x] still non-0	Indicates that the LFIB is being deleted, but that slot x is still active.
disable lc mesg recvd, type=0x1	Indicates that a message to disable label switching type 1 (dynamic) was received by the line card.
disable entered, table exists,type=0x1	Indicates that a call to disable dynamic label switching was issued.
Ethernet4/0/1 fibidb subblock message received	Indicates that a message giving fibidb status change was received on the line card.
enable lc msg recvd,type=0x1	Indicates that the line card received a message to enable label switching type 1 (dynamic).
Tunnel301 set encapfix to 0x6016A97C	Shows that fibidb Tunnel301 on the line card received an encapsulation fixup.
types now 0x3, returning	Shows the value of the bitmask indicating the type of label switching enabled on the interface, as follows: • 0x1--means dynamic label switching • 0x2--means tsp-tunnels • 0x3--means both

Related Commands

Command	Description
debug mpls lfib cef	Prints detailed information about label rewrites being created, resolved, and deactivated as CEF routes are added, changed, or removed.
debug mpls lfib lsp	Prints detailed information about label rewrites being created and deleted as LSP tunnels are added or removed.
debug mpls lfib struct	Traces the allocation and freeing of LFIB-related data structures, including the LFIB itself, label rewrites, and label_info data.

debug mpls lfib struct

To trace the allocation and freeing of label-forwarding information base (LFIB)-related data structures, such as the LFIB itself, label rewrites, and label_info data, use the **debugmplslfibstruct** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls lfib struct

no debug mpls lfib struct

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.1(3)T	This command was modified to reflect new MPLS IETF terminology and CLI syntax.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following is sample output from the **debugmplslfibstruct** command:

```
Router# debug mpls lfib struct
TFIB data structure changes debugging is on
TFIB: delete tag rew, incoming tag 32
TFIB: remove from tfib,inc tag=32
TFIB: set loadinfo,tag=32,no old loadinfo,no new loadinfo
TFIB: TFIB not in use.  Checking for entries.
TFIB: cleanup: tfib[0] still non-0
TFIB: remove from tfib,inc tag=Tun_hd
TFIB: set loadinfo,tag=Exp_null,no old loadinfo,no new loadinfo
TFIB: TFIB freed.
TFIB: enable, TFIB allocated, size 4024 bytes, maxtag = 500
TFIB: create tag rewrite: inc Tun_hd,outg Unkn
TFIB: add to tfib at Tun_hd, first in circular list, mac=0,enc=0
TFIB: delete tag rew, incoming tag Tun_hd
TFIB: remove from tfib,inc tag=Tun_hd
TFIB: set loadinfo,tag=Exp_null,no old loadinfo,no new loadinfo
TFIB: create tag rewrite: inc Tun_hd,outg Unkn
TFIB: add to tfib at Tun_hd, first in circular list, mac=0,enc=0
TFIB: create tag rewrite: inc 26,outg Unkn
TFIB: add to tfib at 26, first in circular list, mac=0,enc=0
TFIB: add to tfib at 27, added to circular list, mac=0,enc=0
TFIB: delete tag rew, incoming tag Tun_hd
TFIB: remove from tfib,inc tag=Tun_hd
TFIB: set loadinfo,tag=Exp_null,no old loadinfo,no new loadinfo
```

```
TFIB: add to tfib at 29, added to circular list, mac=4,enc=8
TFIB: delete tag rew, incoming tag 29
TFIB: remove from tfib,inc tag=29
```

The following table describes the significant fields shown in the display.

Table 17: debug mpls lfib struct Field Descriptions

Field	Description
TFIB	The subsystem issuing the message.
delete tag rew	A label rewrite is being freed.
remove from tfib	A label rewrite is being removed from the LFIB.
inc tag=s	The incoming label of the entry being processed.
set loadinfo	The loadinfo field in the LFIB entry is being set (used for nonrecursive accounting).
tag=s	The incoming label of the entry being processed.
no old loadinfo	The LFIB entry did not have a loadinfo before.
no new loadinfo	The LFIB entry should not have a loadinfo now.
TFIB not in use. Checking for entries.	Label switching has been disabled and the LFIB is being freed up.
cleanup: tfib[x] still non-0	The LFIB is being checked for any entries in use, and entry x is the lowest numbered slot still in use.
TFIB freed	The LFIB table has been freed.
enable, TFIB allocated, size x bytes, maxtag = y	Label switching has been enabled and an LFIB of x bytes has been allocated. The largest legal label is y.
create tag rewrite	A label rewrite is being created.
inc s	The incoming label.
outg s	The outgoing label.
add to tfib at s	A label rewrite has been placed in the LFIB at slots.
first in circular list	This LFIB slot had been empty and this is the first rewrite in the list.
mac=0,enc=0	Length of the MAC string and total encapsulation length, including labels.

Field	Description
added to circular list	A label rewrite is being added to an LFIB slot that already had an entry. This rewrite is being inserted in the circular list.

Related Commands

Command	Description
debug mpls lfib cef	Prints detailed information about label rewrites being created, resolved, and deactivated as CEF routes are added, changed, or removed.
debug mpls lfib lsp	Prints detailed information about label rewrites being created and deleted as LSP tunnels are added or removed.
debug mpls lfib state	Traces what happens when label switching is enabled or disabled.

debug mpls lspv

To display information related to the Multiprotocol Label Processing (MPLS) label switched path (LSP) Ping/Traceroute feature, use the **debugmplslspv**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls lspv [tlv] [error] [event] [ipc] [**packet** [**data**| **error**]] **[path-discovery] [multipath] [all]**

no debug mpls lspv

Syntax Description

tlv	(Optional) Displays MPLS) echo packet type, length, values (TLVs) information as it is being coded and decoded.
error	(Optional) Displays error conditions encountered during MPLS echo request and echo reply encoding and decoding. See the table below.
event	(Optional) Displays MPLS echo request and reply send and receive event information.
ipc	(Optional) Interprocess communication. Displays debug information regarding communication between the Route Processor and line cards.
packet data	(Optional) Displays detailed debugging information for the MPLS echo packets sent and received. This output is seen only on the originating router and the router generating the reply.
packet error	(Optional) Displays packet errors for MPLS echo request and reply. No output is expected for this command.
path-discovery	(Optional) Provides information regarding LSP traceroute path discovery operations.
multipath	(Optional) Displays multipath information.
all	(Optional) Enables all the command keywords.

Command Default

MPLS LSP debugging is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(27)S	This command was introduced.
12.4(6)T	The following keywords were added: **ipc**, **path-discovery**, **multipath**, and **all**.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.28(SB) and implemented on the Cisco 10000 series router.
12.0(32)SY	This command was integrated into Cisco IOS Release 12.0(32)SY.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(11)T	This command was integrated into Cisco IOS Release 12.4(11)T.
12.2(31)SB2	This command was integrated into Cisco IOS Release 12.2(31)SB2.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.0(33)S	The following keywords were added for Cisco IOS Release 12.0(33)S: **ipc**, **path-discovery**, **multipath**, and **all**.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.

Usage Guidelines

Use this command to monitor activity associated with the **pingmpls** and the **tracempls** commands.

The following table lists the messages displayed by the **debugmplslspverror**command and the reason for each error message.

Table 18: Messages Displayed by the debug mpls lspv error Command

Message	Reason Why Message Is Displayed
Echo reply discarded because not routable	An echo reply message is sent because the IP header indicates that the packet has the Router Alert set and the packet is not routable.
UDP checksum error, packet discarded	A packet is received on the port being used by Label Switched Path Verification (LSPV) and there is a checksum error on the packet.
Invalid echo message type	An MPLS echo packet with an invalid echo message type (neither a request nor a reply) is received.

Message	Reason Why Message Is Displayed
Illegal Action	The state machine that drives the LSPV software detects an invalid condition.

Examples

The following is sample output from the**pingmpls** command when LSPV event debugging is enabled:

```
Router# debug mpls lspv event
LSPV event debugging is on
Router# ping mpls ipv4 10.131.159.252/32 repeat 1
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 48/48/48 ms
Router#
*Dec 31 19:31:15.366: LSPV:
waiting for 2 seconds
*Dec 31 19:31:15.366: LSPV: sender_handle: 2000002D, Event Echo Requests Start,
[Idle->Waiting for Echo Reply]
*Dec 31 19:31:15.414: LSPV: sender_handle: 2000002D, Event Echo Reply Received,
[Waiting for Echo Reply->Waiting for Interval]
*Dec 31 19:31:15.466: LSPV: sender_handle: 2000002D, Event Echo Requests Cancel,
[Waiting for Interval->Idle]
Router# undebug all
All possible debugging has been turned off
```

The following is sample output from the**pingmpls** command when LSPV TLV debugging is enabled:

```
Router# debug mpls lspv tlv
LSPV tlv debugging is on
Router# ping mpls ipv4 10.131.159.252/32 repeat 1
Sending 1, 100-byte MPLS Echos to 10.131.159.252/32,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target
Type escape sequence to abort.
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 40/40/40 ms
Router#
*Dec 31 19:32:32.566: LSPV: Echo Hdr encode: version 1, msg type 1, reply mode 2
, return_code 0, return_subcode 0, sender handle 9400002E, sequence number 1,
timestamp sent 14:32:32 EST Wed Dec 31 2003, timestamp rcvd 19:00:00 EST Thu Dec 31 1899
*Dec 31 19:32:32.566: LSPV: IPV4 FEC encode: destaddr 10.131.159.252/32
*Dec 31 19:32:32.566: LSPV: Pad TLV encode: type 1, size 18, pattern 0xABCD
*Dec 31 19:32:32.606: LSPV: Echo Hdr decode: version 1, msg type 2, reply mode 2,
return_code 3, return_subcode 0, sender handle 9400002E, sequence number 1,
timestamp sent 14:32:32 EST Wed Dec 31 2003, timestamp rcvd 14:32:32 EST Wed Dec 31 2003
Router# undebug all
All possible debugging has been turned off
```

The following is sample output from the **tracemplsmultipath** command when LSPV multipath debugging is on:

```
Router# debug mpls lspv multipath
multipath information debugging is on
Router# trace mpls multipath ipv4 10.5.5.5/32

Starting LSP Multipath Traceroute for 10.5.5.5/32
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
```

```
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
LL
*Aug 30 20:39:03.719: LSPV: configuring bitmask multipath, base 0x7F000000, bitmapsize 32,
 start 0x7F000000, numbits 32
*Aug 30 20:39:03.719: LSPV: multipath info: info_length 4, bitmapsize 32, multipath_length
 8, start 127.0.0.0, base 127.0.0.0, numbits 32
*Aug 30 20:39:03.719: LSPV: multipath info: info_length 4, bitmapsize 32, multipath_length
 8, start 127.0.0.0, base 127.0.0.0, numbits 32
*Aug 30 20:39:03.719: LSPV: getnext bit_cursor 0, index 0, mask 0x80000000
*Aug 30 20:39:03.719: LSPV: next addr 127.0.0.1
*Aug 30 20:39:03.719: LSPV: multipath info: datagramsize 8
*Aug 30 20:39:03.719: 7F 00 00 00 FF FF FF FF
*Aug 30 20:39:04.007: LSPV: multipath info: !
Path 0 found,
 output interface Et1/0 source 10.2.3.2 destination 127.0.0.1
Paths (found/broken/unexplored) (1/0/0)
 Echo Request (sent/fail) (3/0)
 Echo Reply (received/timeout) (3/0)
 Total Time Elapsed 924 ms
Router#
*Aug 30 20:39:04.007: 7F 00 00 00 FF FF FF FF
*Aug 30 20:39:04.007: LSPV: ds map convert: rtr_id A030404, mtu 1500 intf_addr 10.3.4.4
hashkey 8, multipath length 8, info 2130706432
*Aug 30 20:39:04.007: LSPV: multipath info: hashkey type 8, base 0x7F000000, bitmapsize 32,
 info0 0xFFFFFFFF
*Aug 30 20:39:04.007: LSPV: multipath info: info_length 4, bitmapsize 32, multipath_length
 8, start 127.0.0.1, base 127.0.0.1, numbits 32
*Aug 30 20:39:04.007: LSPV: getnext bit_cursor 0, index 0, mask 0x80000000
*Aug 30 20:39:04.007: LSPV: next addr 127.0.0.1
*Aug 30 20:39:04.007: LSPV: multipath info: datagramsize 8
*Aug 30 20:39:04.007: 7F 00 00 00 FF FF FF FF
*Aug 30 20:39:04.299: LSPV: multipath info: datagramsize 8
*Aug 30 20:39:04.299: 7F 00 00 00 FF FF FF FF
*Aug 30 20:39:04.299: LSPV: ds map convert: rtr_id A040505, mtu 1504 intf_addr 10.4.5.5
hashkey 8, multipath length 8, info 2130706432
*Aug 30 20:39:04.299: LSPV: multipath info: hashkey type 8, base 0x7F000000, bitmapsize 32,
 info0 0xFFFFFFFF
*Aug 30 20:39:04.299: LSPV: multipath info: info_length 4, bitmapsize 32, multipath_length
 8, start 127.0.0.1, base 127.0.0.1, numbits 32
*Aug 30 20:39:04.299: LSPV: getnext bit_cursor 0, index 0, mask 0x80000000
*Aug 30 20:39:04.299: LSPV: next addr 127.0.0.1
*Aug 30 20:39:04.299: LSPV: multipath info: datagramsize 8
*Aug 30 20:39:04.299: 7F 00 00 00 FF FF FF FF
Router# undebug all

multipath information debugging is off
```

Related Commands

Command	Description
ping mpls	Checks MPLS LSP connectivity.
trace mpls	Discovers MPLS LSP routes that packets will actually take when traveling to their destinations.

debug mpls mldp all

To enable debugging output for all Multicast Label Distribution Protocol (MLDP) events, use the **debug mpls mldp all**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp all

no debug mpls mldp all

Syntax Description

This command has no arguments or keywords.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for all MLDP events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network MVPN core network. Issuing this command is equivalent to issuing the following commands:

- debug mpls mldp filter opaque_type
- debug mpls mldp generic
- debug mpls mldp gr
- debug mpls mldp mfi
- debug mpls mldp mrib
- debug mpls mldp neighbor
- debug mpls mldp packet

Examples

The following example shows how to enable debugging output for all MLDP events:

```
Router# debug mpls mldp all
```

Related Commands

Command	Description
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.
debug mpls mldp generic	Enables debugging output for generic MLDP events.
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp filter opaque_type

To enable filtering of Multicast Label Distribution Protocol (MLDP) debugging output using the opaque type, use the **debug mpls mldp filter opaque_type**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp filter opaque_type *type*

no debug mpls mldp opaque_type *type*

Syntax Description

type	The opaque type to be used for filtering. The following types are supported: • **ipv4** *source-group* --this represents the "IPv4 Protocol Independent Source-Specific Transit" multicast application type. The IPv4 source address and group address are also specified. • **ipv6** *source-group* --this represents the "IPv6 Protocol Independent Source-Specific Transit" multicast application type.The IPv6 source address and group address are also specified. • **mdt** *vpn-id mdt-number* --this represents the "Multicast Virtual Private Network (MVPN)" multicast application type. The VPN identifier and the Multicast Distribution Tree (MDT) number are also specified. • **vpnv4** *source-group route-distinguisher* --this represents the "Direct MDT (VPNv4) " multicast application type.The IPv4 source address, group address, and the VPN route distinguisher are also specified. • *type-number* --the type-number. Valid values are from 0-65535.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables filtering of MLDP debugging output using the opaque type. This output occurs when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN) core network. The opaque type represents the type of multicast application (for example, MVPN) occurring across an MPLS core network.

Examples

The following example shows how to enable filtering of MLDP debugging output using the opaque type:

```
Router# debug mpls mldp filter opaque_type mdt 100:2 0
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp generic	Enables debugging output for generic MLDP events.
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp generic

To enable debugging output for generic Multicast Label Distribution Protocol (MLDP) events, use the **debug mpls mldp generic**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp generic [**i** *d*]

no debug mpls mldp generic [**i** *d*]

Syntax Description

id	(Optional) The hexadecimal Label Switched Multicast (LSM) system ID.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for generic MLDP events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN) core network.

Examples

The following example shows how to enable debugging output for generic MLDP events:

```
Router# debug mpls mldp generic
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.

Command	Description
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp gr

To enable debugging output for Multicast Label Distribution Protocol (MLDP) graceful restart (GR) events, use the **debug mpls mldp gr**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp gr [*id*]

no debug mpls mldp [*id*]

Syntax Description

id	(Optional) The hexadecimal Label Switched Multicast (LSM) system ID.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for MLDP graceful restart events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN) core network.

Examples

The following example shows how to enable debugging output for MLDP GR events:

```
Router# debug mpls mldp gr
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.
debug mpls mldp generic	Enables debugging output for generic MLDP events.

Command	Description
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp mfi

To enable debugging output for Multicast Label Distribution Protocol/Multicast Forwarding Information (MLDP/MFI) interaction events, use the **debug mpls mldp mfi**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp mfi [*id*]

no debug mpls mldp mfi [*id*]

Syntax Description

id	(Optional) The hexadecimal Label Switched Multicast (LSM) system ID.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for MLDP/MFI interaction events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN) core network.

Examples

The following example shows how to enable debugging output for MLDP/MFI interaction events:

```
Router# debug mpls mldp mfi
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.
debug mpls mldp generic	Enables debugging output for generic MLDP events.

Command	Description
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp mrib

To enable debugging output for Multicast Label Distribution Protocol/Multicast Routing Information Base (MLDP/MRIB) interaction events, use the **debug mpls mldp mrib**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp mrib

no debug mpls mldp mrib

Syntax Description

This command has no arguments or keywords.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for MLDP/MRIB interaction events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN) core network.

Examples

The following example shows how to enable debugging output for MLDP/MRIB interaction events:

```
Router# debug mpls mldp mrib
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.
debug mpls mldp generic	Enables debugging output for generic MLDP events.
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.

Command	Description
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp neighbor

To enable debugging output for Multicast Label Distribution Protocol (MLDP) neighbor events, use the **debug mpls mldp neighbor**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp neighbor

no debug mpls mldp neighbor

Syntax Description

This command has no arguments or keywords.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for MLDP neighbor events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN)core network.

Examples

The following example shows how to enable debugging output for MLDP neighbor events:

```
Router# debug mpls mldp neighbor
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.
debug mpls mldp generic	Enables debugging output for generic MLDP events.
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.

Command	Description
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp packet	Enables debugging output for MLDP-generated MPLS control plane events.
show mpls mldp database	Displays MLDP information.

debug mpls mldp packet

To enable debugging output for Multicast Label Distribution Protocol (MLDP)-generated Multiprotocol Label Switching (MPLS) control plane events, use the **debug mpls mldp packet**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls mldp packet

no debug mpls mldp packet

Syntax Description

This command has no arguments or keywords.

Command Default

The command is disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)S	This command was introduced.

Usage Guidelines

Use this command when the MLDP-based MVPN feature appears not to be functioning. The command enables debugging output for MLDP-generated MPLS control plane events that occur when the MLDP-based MVPN feature is enabled. This feature provides extensions to Label Distribution Protocol (LDP) for the setup of point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) label switched paths (LSPs) for transport in the Multicast Virtual Private Network (MVPN) core network.

Examples

The following example shows how to enable debugging output for MLDP-generated MPLS control plane events:

```
Router# debug mpls mldp packet
```

Related Commands

Command	Description
debug mpls mldp all	Enables debugging output for all MLDP events.
debug mpls mldp filter opaque_type	Enables filtering of MLDP debugging output using the opaque type.
debug mpls mldp generic	Enables debugging output for generic MLDP events.
debug mpls mldp gr	Enables debugging output for MLDP graceful restart events.

Command	Description
debug mpls mldp mfi	Enables debugging output for MLDP/MFI interaction events.
debug mpls mldp mrib	Enables debugging output for MLDP/MRIB interaction events.
debug mpls mldp neighbor	Enables debugging output for MLDP neighbor events.
show mpls mldp database	Displays MLDP information.

debug mpls netflow

To display debug messages for MPLS egress NetFlow accounting, use the **debug mpls netflow**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls netflow

no debug mpls netflow

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(10)ST	This command was introduced.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXI4	This command was integrated into Cisco IOS Release 12.2(33)SXI4.

Examples

Here is sample output from the **debug mpls netflow** command:

```
Router# debug mpls netflow
MPLS Egress NetFlow debugging is on
Router#
Router#
Router#
4d00h:Egress flow:entry created, dest 3.3.3.3/32, src 34.0.0.1/8
Router#
Router#
4d00h:Egress flow:entry created, dest 3.3.3.3/32, src 42.42.42.42/32
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# int eth1/4
Router(config-if)# no mpls netflow egress
Router(config-if)#
4d00h:MPLS output feature change, trigger TFIB scan
4d00h:tfib_scanner_walk, prefix 5.5.5.5/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 2.0.0.0/8, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 3.3.3.3/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 40.40.40.40/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 50.50.50.50/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 100.100.100.100/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 180.1.1.0/24, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 190.1.1.0/24, rewrite flow flag 1
```

```
4d00h:tfib_scanner_walk, prefix 2.0.0.0/8, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 4.4.4.4/32, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 40.40.40.40/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 50.50.50.50/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 177.1.1.0/24, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 180.1.1.0/24, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 190.1.1.0/24, rewrite flow flag 1
Router(config-if)#
Router(config-if)# mpls netflow egress
Router(config-if)#
4d00h:Interface refcount with output feature enabled = 2
4d00h:MPLS output feature change, trigger TFIB scan
4d00h:tfib_scanner_walk, prefix 5.5.5.5/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 2.0.0.0/8, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 3.3.3.3/32, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 40.40.40.40/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 50.50.50.50/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 100.100.100.100/32, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 180.1.1.0/24, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 190.1.1.0/24, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 2.0.0.0/8, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 4.4.4.4/32, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 40.40.40.40/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 50.50.50.50/32, rewrite flow flag 0
4d00h:tfib_scanner_walk, prefix 177.1.1.0/24, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 180.1.1.0/24, rewrite flow flag 1
4d00h:tfib_scanner_walk, prefix 190.1.1.0/24, rewrite flow flag 1
4d00h:Egress flow:entry created, dest 3.3.3.3/32, src 42.42.42.42/32
Router(config-if)#
Router(config-if)# end
Router# show run int eth1/4
Building configuration...
Current configuration:
!
interface Ethernet1/4
 ip vrf forwarding vpn1
 ip address 180.1.1.1 255.255.255.0
 no ip directed-broadcast
 mpls netflow egress
end
Router#
Router#
4d00h:%SYS-5-CONFIG_I:Configured from console by console
Router#
```

Note Flow flag 1 prefixes are reachable through this interface; therefore, MPLS egress NetFlow accounting is applied to all packets going out the destination prefix. Flow flag 0 prefixes are not reachable through this interface; therefore, MPLS egress NetFlow accounting is not applied to any packets going out the destination prefix.

Related Commands

Command	Description
show debug	Displays active debug output.

debug mpls packets

To display Multiprotocol Label Switching (MPLS) labeled packets switched by the host router, use the **debugmplspackets**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls packets [*interface*]

no debug mpls packets [*interface*]

Syntax Description

interface	(Optional) The interface or subinterface name.

Command Default

The debug output displays all labeled packets, regardless of the interface.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1CT	This command was introduced.
12.1(3)T	This command was modified for new MPLS terminology and syntax.
12.2(25)S	The command ouput was enhanced to display MPLS high availability information.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB and implemented on the Cisco 10000 series routers.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Usage Guidelines

The optional> interface parameter restricts the display to only those packets received or sent on the indicated interface or subinterface.

Note Use this command with care because it generates output for every packet processed. Furthermore, enabling this command causes fast and distributed label switching to be disabled for the selected interfaces. To avoid adversely affecting other system activity, use this command only when traffic on the network is at a minimum.

Examples

The following is sample output from the **debugmplspackets** command:

```
Router# debug mpls packets
TAG: Hs3/0: recvd: CoS=0, TTL=254, Tag(s)=27
TAG: Hs0/0: xmit: (no tag)
TAG: Hs0/0: recvd: CoS=0, TTL=254, Tag(s)=30
TAG: Hs3/0: xmit: CoS=0, TTL=253, Tag(s)=27
```

The following table describes the significant fields shown in the display.

Table 19: debug mpls packets Field Descriptions

Field	Description
Hs0/0	The identifier for the interface on which the packet was received or sent.
recvd	Packet received.
xmit	Packet transmitted.
CoS	Class of Service field from the packet label header.
TTL	Time to live field from the packet label header.
(no tag)	Last label was popped off the packet and sent unlabeled.
Tag(s)	A list of labels on the packet, ordered from the top of the stack to the bottom.

Examples

The following is sample output from the **debugmplspackets** command:

```
Router# debug mpls packets
Gi6/0/0: rx: Len 118 Stack {30 6 255} - ipv4 data
Gi6/1/1: tx: Len 118 Stack {22 6 254} - ipv4 data
```

Related Commands

Command	Description
show mpls forwarding-table	Displays the contents of the MPLS forwarding table.

debug mpls static binding

To display information related to static binding events, use the **debug mpls static binding** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls static binding [**vrf** *vpn-name*]

no debug mpls static binding [**vrf** *vpn-name*]

Syntax Description

vrf *vpn-name*	(Optional) Displays information only for the specified VPN routing and forwarding instance.

Command Default

Static binding event information is not displayed.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.0(26)S	This command was integrated into Cisco IOS Release 12.0(26)S.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines

Use the **debug mpls static binding** command to troubleshoot problems related to Multiprotocol Lbel Switching (MPLS) static labels or VRF-aware MPLS static labels. If you specify the **vrf** keyword, debugging is enabled only for a specified VRF. You can enable debugging only for a VRF.

Examples

The following is sample output from the **debug mpls static binding** command:

```
Router# debug mpls static binding vrf vpn100
MPLS Static label bindings debugging is on
00:15:13: mpls: Add remote static binding: 10.0.0.0/8; label 0; nexthop 172.16.0.8:0
00:15:13: mpls: Add static label binding for 10.0.0.0/8
```

```
00:15:13: mpls: Add static label binding for 10.0.0.1/8
00:15:13: mpls: Add remote static binding: 10.0.0.1/8; label 2607; nexthop 172.17.0.66:0
00:15:13: mpls: Add static label binding for 172.18.0.0/16
00:15:18: mpls: Periodic static label adjust
00:15:18: mpls: Static label update: 10.0.0.0/8
00:15:18:       Add remote label: nexthop: 172.16.0.8:0; label: 0
00:15:18: mpls: Periodic static label adjust
00:15:18: mpls: Static label update: 10.0.0.1/8
00:15:18:       Waiting for withdrawal of dynamic local label 55
00:15:18:       Add remote label: nexthop: 172.17.0.66:0; label: 2607
00:15:18: mpls: Periodic static label adjust
00:15:18: mpls: Static label update: 172.18.0.0/16
00:15:18:       Waiting for withdrawal of dynamic local label 17
00:15:28: mpls: Periodic static label adjust
00:15:28: mpls: Periodic static label adjust
00:15:28: mpls: Static label update: 10.0.0.1/8
00:15:28:       Local label 55 added to tib
00:15:28:       Signal route tag change, in label 55;out label 8388611; nh 192.168.44.77
00:15:28: mpls: Periodic static label adjust
00:15:28: mpls: Static label update: 172.18.0.0/16
00:15:28:       Local label 17 added to tib
00:15:28:       Signal route tag change, in label 17;out label 8388611; nh 192.168.44.66
00:15:38: mpls: Periodic static label adjust
00:15:38: mpls: Periodic static label adjust
00:15:38: mpls: Periodic static label adjust
```

Related Commands

Command	Description
mpls static binding ipv4 vrf	Binds a prefix to a local label.
show debug	Displays active debug output.

debug mpls tp

To enable debugging for Multiprotocol Label Switching (MPLS)-Transport Profile (TP), use the **debug mpls tp** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug mpls tp [**all**| **cli**| **error**| **event**| **fault-oam**| **ha**| **init**| **link-num**| **lsp-db**| **lsp-ep**| **lsp-mp**| **mem**| **psc** | {**packet** | **event**}| **tun-db**| **tunnel**]

no debug mpls tp

Syntax Description

all	(Optional) Displays all debug messages.
cli	(Optional) Displays MPLS-TP CLI debug messages.
error	(Optional) Displays MPLS-TP error debug messages.
event	(Optional) Displays MPLS-TP event debug messages.
fault-oam	(Optional) Displays MPLS-TP fault Operation, Administration, and Maintenance (OAM) messages.
ha	(Optional) Displays MPLS-TP high-availability debug messages.
init	(Optional) Displays MPLS-TP initialization debug messages.
link-num	(Optional) Displays MPLS-TP link-management debug messages
lsp-db	(Optional) Displays MPLS-TP midpoint link-state packet (LSP) database debug messages.
lsp-ep	(Optional) Displays MPLS-TP endpoint LSP debug messages.
lsp-mp	(Optional) Displays MPLS-TP midpoint LSP debug messages.
mem	(Optional) Displays MPLS-TP memory allocation and usage debug messages.
psc packet	(Optional) Displays MPLS packets received or transmitted by the Protection State Coordination (PSC) Protocol.

psc event	(Optional) Displays how the Protection State Coordination (PSC) Protocol behaves for any event it receives.
tun-db	(Optional) Displays MPLS-TP tunnel database debug messages.
tunnel	(Optional) Displays MPLS-TP tunnel debug messages.

Command Default

Debug messages are disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.1(1)SA	This command was introduced.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.
Cisco IOS Release 3.9S	This command was modified. The **psc packet** and **event** keywords were added.

Examples

The following example enables the display of MPLS-TP endpoint LSP debug messages:

```
Router# debug mpls tp lsp-ep
debug mpls-tp endpoint lsp setup or use debugging is on
```

Related Commands

Command	Description
show mpls tp	Displays information summary or detailed information about MPLS-TP) settings.

debug mpls traffic-eng areas

To print information about traffic engineering area configuration change events, use the **debug mpls traffic-eng areas** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng areas

no debug mpls traffic-eng areas

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)ST	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about traffic engineering area configuration change events:

```
Router# debug mpls traffic-eng areas
TE-AREAS:isis level-1:up event
TE-PCALC_LSA:isis level-1
```

debug mpls traffic-eng autoroute

To print information about automatic routing over traffic engineering tunnels, use the **debug mpls traffic-eng autoroute** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng autoroute

no debug mpls traffic-eng autoroute

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)ST	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about automatic routing over traffic engineering tunnels:

```
Router# debug mpls traffic-eng autoroute
TE-Auto:announcement that destination 0001.0000.0003.00 has 1 tunnels
    Tunnel1 (traffic share 333, nexthop 10.112.0.12)
```

debug mpls traffic-eng auto-tunnel backup

To print system information about traffic engineering backup autotunnels, use the **debug mpls traffic-eng auto-tunnel backup** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng auto-tunnel backup [**all**| **events**| **state**]

no debug mpls traffic-eng auto-tunnel backup [**all**| **events**| **state**]

Syntax Description

all	(Optional) Enables all backup autotunnel debugging output.
events	(Optional) Prints backup autotunnel system events.
state	(Optional) Prints the system state of backup autotunnels.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(32)S	This command was introduced.
12.4(11)T	This command was integrated into Cisco IOS Release 12.4(11)T.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
15.2(2)S	This command was modified. The output was enhanced to show debugging information for autotunnel and automesh stateful switchover (SSO) tunnels.
Cisco IOS XE Release 3.6S	This command was modified. The output was enhanced to show debugging information for autotunnel and automesh stateful switchover (SSO) tunnels.

Examples

This command shows how to display debugging information about the system state of backup autotunnels:

```
Router# debug mpls traffic-eng auto-tunnel backup state
```

Related Commands

Command	Description
debug mpls traffic-eng auto-tunnel primary	Prints system information about traffic engineering primary tunnels.
debug mpls traffic-eng tunnels events	Prints information about traffic engineering tunnel management system events.
mpls traffic-eng auto-tunnel backup	Automatically builds NHOP and NNHOP backup tunnels.
show ip explicit-paths	Displays the configured IP explicit paths.
show mpls traffic-eng tunnels	Displays information about tunnels.

debug mpls traffic-eng auto-tunnel primary

To print system information about traffic engineering primary autotunnels, use the **debug mpls traffic-eng auto-tunnel primary** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng auto-tunnel primary [**all**| **events**| **state**]

no debug mpls traffic-eng auto-tunnel primary [**all**| **events**| **state**]

Syntax Description

all	(Optional) Enables all primary autotunnel debugging output.
events	(Optional) Prints primary autotunnel system events.
state	(Optional) Prints the system state of primary autotunnels.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(32)S	This command was introduced.
12.4(11)T	This command was integrated into Cisco IOS Release 12.4(11)T.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
15.2(2)S	This command was modified. The output was enhanced to show debugging information for autotunnel and automesh stateful switchover (SSO) tunnels.
Cisco IOS XE Release 3.6S	This command was modified. The output was enhanced to show debugging information for autotunnel and automesh stateful switchover (SSO) tunnels.

Examples

In the following example, debugging information is printed about system events of primary autotunnels:

```
Router# debug mpls traffic-eng auto-tunnel primary events

*Feb 6 18:12:57.871: TE_AUTO_TUN: primary CLI command:
interface tunnel3000
```

```
no logging event link-status
ip unnumbered Loopback0
tunnel destination 192.168.1.1
tunnel mode mpls traffic-eng
end
```

In the following example, debugging information is printed about the system state of primary autotunnels:

```
Router# debug mpls traffic-eng auto-tunnel primary state

Sample for debug mpls traffic-eng auto-tunnel primary state:  *Feb 6 18:11:44.363:
TE_AUTO_TUN: Didn't find protected Up Tunnel3000 to router id 192.168.1.1 out POS2/0
Sample for debug mpls traffic-eng auto-tunnel backup events (this is one log that prints
on multiple lines): *Feb 6 18:19:04.303: TE_AUTO_TUN: CLI command:
ip explicit-path name __dynamic_tunnel4000
index 1 next-address 192.168.1.2
```

Related Commands

Command	Description
debug mpls traffic-eng auto-tunnel backup	Prints system information about traffic engineering backup autotunnels.
debug mpls traffic-eng tunnels events	Prints information about traffic engineering tunnel management system events.
mpls mpls traffic-eng auto-tunnel primary config	Enables IP processing without an explicit address.
show ip explicit-paths	Displays the configured IP explicit paths.
show mpls traffic-eng tunnels	Displays information about tunnels.

debug mpls traffic-eng filter

To filter the display of Multiprotocol Label Switching (MPLS) traffic engineering messages by access control list (ACL), point-to-point (P2P) messages, or point-to-multipoint (P2MP) messages, use the **debug mpls traffic-eng filter** command in Privileged EXEC configuration mode. To disable the display of these messages, use the **no** form of this command.

debug mpls traffic-eng filter {**acl** *acl-num*| **dest-mode** {**p2p**| **p2mp**}}

no debug mpls traffic-eng filter {**acl** *acl-num*| **dest-mode** {**p2p**| **p2mp**}}

Syntax Description

acl *acl-num*	Displays debug information for the specified ACL. Valid values are 1-199 and 1300-2699.
dest-mode	Displays debug information sorted by P2P or P2MP messages.
p2p	Displays debug information about P2P tunnels.
p2mp	Displays debug information about P2MP tunnels.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(33)SRE	This command was introduced.

Examples

The following example enables the display of debug message for P2MP tunnels:

```
Router# debug mpls traffic-eng filter dest-mode p2mp
Setting filter for TE P2MP Tunnels/LSPs
```

Related Commands

Command	Description
show mpls traffic-eng tunnels	Displays information about P2P and P2MP tunnels.

debug mpls traffic-eng forwarding-adjacency

To display debug messages for traffic engineering (TE) forwarding adjacency events, use the **debug mpls traffic-eng forwarding-adjacency** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng forwarding-adjacency [detail] [*access-list-number*]

no debug mpls traffic-eng forwarding-adjacency

Syntax Description

detail	(Optional) Prints detailed debug information.
access-list-number	(Optional) Displays number of the access list. • A standard IP access list is in the range 1 to 199. • An extended IP access list is in the range 1300 to 2699.

Command Default

This command is disabled by default.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(15)S	This command was introduced.
12.0(16)ST	This command was integrated into Cisco IOS Release 12.0(16)ST.
12.2(18)S	This command was integrated into Cisco IOS Release 12.2(18)S.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(27)SBC	This command was integrated into Cisco IOS Release 12.2(27)SBC.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Usage Guidelines

Use the **debug mpls traffic-eng forwarding-adjacency** command to troubleshoot any problems that occur after you configure the **tunnel mpls traffic-eng forwarding-adjacency** command.

If you enter the **detail** keyword before the *access-list-number* argument, you can specify an access list. However, if you enter an access list before you enter the **detail** keyword, you cannot specify the **detail** keyword.

Examples

The following is sample output from the **debug mpls traffic-eng forwarding-adjacency**command:

```
Router# debug mpls traffic-eng forwarding-adjacency
MPLS traffic-eng debugging is on
```

With a tunnel configured, the following output appears:

```
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 192.168.1.7
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng forwarding-adjacency
 tunnel mpls traffic-eng priority 7 7
 tunnel mpls traffic-eng path-option 10 dynamic
end
```

When the tunnel comes up, the command generates the following debug output:

```
*Oct  2 12:27:07.846:TE-Auto:announcement that destination 0168.0001.0007.00 has 1 tunnels
*Oct  2 12:27:07.846:    Tunnel0     (traffic share 142857, nexthop 192.168.1.7)
*Oct  2 12:27:07.846:                (flags: Forward-Adjacency, holdtime 0)
```

Related Commands

Command	Description
show debug	Displays active debug output.
show mpls traffic-eng forwarding-adjacency	Displays TE tunnels being advertised as links in an IGP network.
tunnel mpls traffic-eng forwarding-adjacency	Advertises a TE tunnel as a link in an IGP network.

debug mpls traffic-eng ha sso

To display debugging output for Multiprotocol Label Switching (MPLS) traffic engineering high availability (HA) activities during the graceful switchover from an active Route Processor (RP) to a redundant standby RP, use the **debug mpls traffic-eng ha sso** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng ha sso {**auto-tunnel**| **errors**| **link-management** {**events**| **standby**| **recovery**| **checkpoint**}| **tunnel** {**events**| **standby**| **recovery**}}

no debug mpls traffic-eng ha sso {**auto-tunnel**| **errors**| **link-management** {**events**| **standby**| **recovery**| **checkpoint**}| **tunnel** {**events**| **standby**| **recovery**}}

Syntax Description

auto-tunnel	Displays information about autotunnel activity during the MPLS traffic engineering stateful switchover (SSO) process.
errors	Displays errors encountered during the MPLS traffic engineering SSO process.
link-management	Displays information about link management activity during the MPLS traffic engineering SSO process.
events	Displays significant events that occur during the MPLS traffic engineering SSO process.
standby	Displays information about the standby behavior during the MPLS traffic engineering SSO process.
recovery	Displays information about recovery activity during the MPLS traffic engineering SSO process.
checkpoint	Display information about checkpointing activities during the MPLS traffic engineering SSO process. Checkpointing occurs when a message is sent and acknowledged.
tunnel	Displays information about tunnel activity during the MPLS traffic engineering SSO process.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(33)SRA	This command was introduced.

Release	Modification
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
15.2(2)S	This command was modified. The output for the **debug mpls traffic-eng ha sso tunnel events**, the **debug mpls traffic-eng ha sso standby**, and the **debug mpls traffic-eng ha sso tunnel recovery** commands was enhanced to show debugging information for autotunnel and automesh stateful switchover (SSO) tunnels.
Cisco IOS XE Release 3.6S	This command was modified. The output for the **debug mpls traffic-eng ha sso tunnel events**, the **debug mpls traffic-eng ha sso standby**, and the **debug mpls traffic-eng ha sso tunnel recovery** commands was enhanced to show debugging information for autotunnel and automesh stateful switchover (SSO) tunnels.

Usage Guidelines

This command displays debugging output about the SSO process for MPLS traffic engineering tunnels, autotunnels, and link management systems. The SSO process occurs when the active router becomes unavailable and system control and routing protocol execution are transferred from the now inactive RP to the redundant standby RP, thus providing uninterrupted network services.

Examples

The following is sample output from the **debug mpls traffic-eng ha sso** command when you have enabled debugging keywords to monitor the SSO process for tunnels and link management systems as the standby router becomes active:

```
Router# debug mpls traffic-eng ha sso link-management events
MPLS traffic-eng SSO link management events debugging is on
Router# debug mpls traffic-eng ha sso link-management recovery
MPLS traffic-eng SSO link management recovery debugging is on
Router# debug mpls traffic-eng ha sso link-management standby
MPLS traffic-eng SSO link management standby behavior debugging is on
Router# debug mpls traffic-eng ha sso link-management
checkpoint
MPLS traffic-eng SSO link management checkpointed info debugging is on
Router# debug mpls traffic-eng ha sso tunnel standby
MPLS traffic-eng SSO tunnel standby behavior debugging is on
Router# debug mpls traffic-eng ha sso tunnel recovery
MPLS traffic-eng SSO tunnel head recovery debugging is on
Router# debug mpls traffic-eng ha sso tunnel events
MPLS traffic-eng SSO events for tunnel heads debugging is on
Router# debug mpls traffic-eng ha sso errors
MPLS traffic-eng SSO errors debugging is on
Router# show debug
<-----
This command displays the debugging that is enabled.
MPLS TE:
  MPLS traffic-eng SSO link management events debugging is on
  MPLS traffic-eng SSO link management recovery debugging is on
  MPLS traffic-eng SSO link management standby behavior debugging is on
  MPLS traffic-eng SSO link management checkpointed info debugging is on
  MPLS traffic-eng SSO tunnel standby behavior debugging is on
  MPLS traffic-eng SSO tunnel head recovery debugging is on
  MPLS traffic-eng SSO events for tunnel heads debugging is on
  MPLS traffic-eng SSO errors debugging is on
```

```
Router#
Standby-Router#
```

Following is the sample debugging output displayed during a successful SSO recovery on the standby router as it becomes active:

```
*May 12 20:03:15.303: RRR_HA_STATE: Told to wait for IGP convergence
*May 12 20:03:14.807: %FABRIC-SP-STDBY-5-FABRIC_MODULE_ACTIVE: The Switch Fabric Module in
 slot 5 became active.
*May 12 20:03:15.763: RRR_HA_REC: Attempting to recover last flooded info; protocol: OSPF,
 area: 0
*May 12 20:03:15.763: RRR_HA_REC: recovered ospf area 0 instance 0x48FFF240
*May 12 20:03:15.763: RRR_HA_REC: recovered system info
*May 12 20:03:15.763: RRR_HA_REC: recovered link[0] info
*May 12 20:03:15.763: RRR_HA: Recovered last flooded info for igp: OSPF, area: 0
*May 12 20:03:15.763: Pre announce tunnel 10
*May 12 20:03:15.763: TSPVIF_HA_EVENT: added Router_t10 to dest list
*May 12 20:03:15.763: TSPVIF_HA_EVENT: Completed announcement of 1 tunnel heads to IGP
*May 12 20:03:15.763: TSPVIF_HA_REC: Attempting to recover Tunnel10 after SSO
*May 12 20:03:15.763: LSP-TUNNEL-REOPT: Tunnel10 [61] set to recover
*May 12 20:03:15.763: TSPVIF_HA_REC: Recovered number hops = 5
*May 12 20:03:15.763: TSPVIF_HA_REC: recovered ospf area 0 instance 0x48FFF240
*May 12 20:03:15.763: TSPVIF_HA_REC: Recovered Hop 0: 10.0.3.1, Id: 10.0.0.3 Router Node
(ospf) flag:0x0
*May 12 20:03:15.763: TSPVIF_HA_REC: Recovered Hop 1: 10.0.3.2, Id: 10.0.0.7 Router Node
(ospf) flag:0x0
*May 12 20:03:15.763: TSPVIF_HA_REC: Recovered Hop 2: 10.0.6.1, Id: 10.0.0.7 Router Node
(ospf) flag:0x0
*May 12 20:03:15.763: TSPVIF_HA_REC: Recovered Hop 3: 10.0.6.2, Id: 10.0.0.9 Router Node
(ospf) flag:0x0
*May 12 20:03:15.763: TSPVIF_HA_REC: Recovered Hop 4: 10.0.0.9, Id: 10.0.0.9 Router Node
(ospf) flag:0x0
*May 12 20:03:15.763: TSPVIF_HA_REC: signalling recovered setup for Tunnel10: popt 1
[61], weight 2
*May 12 20:03:15.891: TSPVIF_HA_REC: recovered Tu10 forwarding info needed by query
*May 12 20:03:15.891: TSPVIF_HA_REC:     output_idb: GigabitEthernet3/2, output_nhop:
180.0.3.2
Standby-Router#
Router#
*May 12 20:03:25.891: TSPVIF_HA_REC: recovered Tu10 forwarding info needed by query
*May 12 20:03:25.891: TSPVIF_HA_REC:     output_idb: GigabitEthernet3/2, output_nhop:
10.0.3.2
*May 12 20:03:35.891: TSPVIF_HA_REC: recovered Tu10 forwarding info needed by query
*May 12 20:03:35.891: TSPVIF_HA_REC:     output_idb: GigabitEthernet3/2, output_nhop:
10.0.3.2
*May 12 20:03:35.895: RRR_HA_STATE: IGP flood prevented during IGP recovery
*May 12 20:03:38.079: LSP-TUNNEL-REOPT: Tunnel10 [61] received RESV for recovered setup
*May 12 20:03:38.079: LSP-TUNNEL-REOPT: Tunnel10 [61] removed as recovery
*May 12 20:03:38.079: TSPVIF_HA_EVENT: notifying RSVP HA to add lsp_info using key
10.0.0.3->10.0.0.9 Tu10 [61] 10.0.0.3
*May 12 20:03:38.079: TSPVIF_HA_EVENT: updated 7600-1_t10 state; action = add; result =
success
*May 12 20:03:38.079: TSPVIF_HA_EVENT: 7600-1_t10 fully recovered; rewrite refreshed
*May 12 20:03:38.079: TSPVIF_HA_EVENT: notifying CBTS bundle about Router_t10
*May 12 20:03:38.079: TSPVIF_HA_EVENT: notifying RSVP HA to remove lsp_info using key
10.0.0.3->10.0.0.9 Tu10 [61] 10.0.0.3
*May 12 20:03:38.079: RRR_HA: Received notification recovery has ended.  Notify IGP to
flood.
*May 12 20:03:38.079: TSPVIF_HA_EVENT: Received notification recovery has ended
*May 12 20:03:38.079: TSPVIF_HA_STANDBY: prevent verifying setups; IGP has not converged
*May 12 20:03:38.083: TSPVIF_HA_STANDBY: preventing new setups; reason: IGP recovering
*May 12 20:03:38.083: TSPVIF_HA_STANDBY: prevent verifying setups; IGP has not converged
*May 12 20:03:38.083: TSPVIF_HA_STANDBY: preventing new setups; reason: IGP recovering
*May 12 20:03:38.083: RRR_HA_STATE: IGP flood prevented during IGP recovery
7600-1#
*May 12 20:03:47.723: RRR_HA: Received notification that RIB table 0 has converged.
*May 12 20:03:47.723: RRR_HA: Received notification all RIBs have converged.  Notify IGP
to flood.
*May 12 20:03:47.723: RRR_HA_STATE: Told not to wait for IGP convergence
*May 12 20:03:47.723: RRR_HA_INFO: update flooded system info; action = add; result = success
*May 12 20:03:47.723: LM System key::
*May 12 20:03:47.723:   Flooding Protocol: ospf
```

```
*May 12 20:03:47.723:   IGP Area ID: 0
*May 12 20:03:47.723: LM Flood Data::
*May 12 20:03:47.723:   LSA Valid flags: 0x0  Node LSA flag: 0x0
*May 12 20:03:47.723:   IGP System ID: 10.0.0.3  MPLS TE Router ID: 10.0.0.3
*May 12 20:03:47.723:   Flooded links: 1  TLV length: 0 (bytes)
*May 12 20:03:47.723:   Fragment id: 0
*May 12 20:03:47.723: rrr_ha_lm_get_link_info_size: link size: 212 bytes; num TLVs: 0
*May 12 20:03:47.723: rrr_ha_sizeof_lm_link_info: link size: 212 bytes; num TLVs: 0
*May 12 20:03:47.723: RRR_HA_INFO: update flooded link[0] info; action = add;
result = success
*May 12 20:03:47.723: RRR HA Checkpoint Info Buffer::
*May 12 20:03:47.723:   Info Handle:          0x490BB1C8
*May 12 20:03:47.723:   Max Size:             212
*May 12 20:03:47.723:   Info Size:            212
*May 12 20:03:47.723:   Info Write Pointer:   0x490BB29C
*May 12 20:03:47.723: LM Link key::
*May 12 20:03:47.723:   Flooding Protocol: ospf  IGP Area ID: 0  Link ID: 0
(GigabitEthernet3/2)
*May 12 20:03:47.723:     Ifnumber: 5  Link Valid Flags: 0x193B
*May 12 20:03:47.723      Link Subnet Type: Broadcast
*May 12 20:03:47.723:     Local Intfc ID: 0  Neighbor Intf ID: 0
*May 12 20:03:47.723:     Link IP Address: 10.0.3.1
*May 12 20:03:47.723:     Neighbor IGP System ID: 10.0.3.2  Neighbor IP Address: 10.0.0.0
*May 12 20:03:47.723:     IGP Metric: 1  TE Metric: 1
*May 12 20:03:47.723:     Physical Bandwidth: 1000000 kbits/sec
*May 12 20:03:47.723:     Res. Global BW: 3000 kbits/sec
*May 12 20:03:47.723:     Res. Sub BW: 0 kbits/sec
*May 12 20:03:47.723:     Upstream::
Router#
*May 12 20:03:47.723:                               Global Pool   Sub Pool
*May 12 20:03:47.723:                               -----------   ----------
*May 12 20:03:47.723:     Reservable Bandwidth[0]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[1]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[2]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[3]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[4]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[5]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[6]:          0            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[7]:          0            0 kbits/sec
*May 12 20:03:47.723:     Downstream::
*May 12 20:03:47.723:                               Global Pool   Sub Pool
*May 12 20:03:47.723:                               -----------   ----------
*May 12 20:03:47.723:     Reservable Bandwidth[0]:       3000            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[1]:       3000            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[2]:       3000            0 kbits/sec
*May 12 20:03:47.723:     Reservable Bandwidth[3]:       3000            0 kbits/sec
*May 12 20:03:47.727:     Reservable Bandwidth[4]:       3000            0 kbits/sec
*May 12 20:03:47.727:     Reservable Bandwidth[5]:       3000            0 kbits/sec
*May 12 20:03:47.727:     Reservable Bandwidth[6]:       3000            0 kbits/sec
*May 12 20:03:47.727:     Reservable Bandwidth[7]:       2900            0 kbits/sec
*May 12 20:03:47.727:     Affinity Bits: 0x0
*May 12 20:03:47.727:     Protection Type: Capability 0,  Working Priority 0
*May 12 20:03:47.727:     Number of TLVs: 0
*May 12 20:03:47.727: RRR_HA: Updated flood state for ospf area 0 with 1 links); result =
success
Router#
```

The following example shows how to turn off debugging:

```
Router# no debug mpls traffic-eng ha sso link-management events
MPLS traffic-eng SSO link management events debugging is off
Router# no debug mpls traffic-eng ha sso link-management recovery
MPLS traffic-eng SSO link management recovery debugging is off
Router# no debug mpls traffic-eng ha sso link-management standby
MPLS traffic-eng SSO link management standby behavior debugging is off
Router# no debug mpls traffic-eng ha sso link-management checkpoint
MPLS traffic-eng SSO link management checkpointed info debugging is off
Router# no debug mpls traffic-eng ha sso tunnel standby
MPLS traffic-eng SSO tunnel standby behavior debugging is off
Router# no debug mpls traffic-eng ha sso tunnel recovery
MPLS traffic-eng SSO tunnel head recovery debugging is off
Router# no debug mpls traffic-eng ha sso tunnel events
MPLS traffic-eng SSO events for tunnel heads debugging is off
```

```
Router# no debug mpls traffic-eng ha errors
MPLS traffic-eng SSO errors debugging is off
```

Related Commands

Command	Description
debug ip rsvp high-availability	Displays debugging output for RSVP HA activities that improve the accessibility of network resources.
debug ip rsvp sso	Displays debugging output for RSVP activities during the graceful switchover from an active RP to a redundant RP.

debug mpls traffic-eng link-management admission-control

To print information about traffic engineering label-switched path (LSP) admission control on traffic engineering interfaces, use the **debug mpls traffic-eng link-management admission-control**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management admission-control [detail] [*acl-number*]

no debug mpls traffic-eng link-management admission-control [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information only for those LSPs that match the access list.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T, and the **detail** keyword and the *acl-number* argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about traffic engineering LSP admission control on traffic engineering interfaces:

```
Router# debug mpls traffic-eng link-management admission-control
TE-LM-ADMIT:tunnel 10.106.0.6 1_10002:created [total 4]
TE-LM-ADMIT:tunnel 10.106.0.6 1_10002: “None” -> “New”
TE-LM-ADMIT:tunnel 10.106.0.6 1_10002: “New” -> “Admitting 2nd Path Leg”
TE-LM-ADMIT:tunnel 10.106.0.6 1_10002: “Admitting 2nd Path Leg” -> “Path Admitted”
TE-LM-ADMIT:Admission control has granted Path query for 10.106.0.6 1_10002 (10.112.0.12)
on link Ethernet4/0/1 [reason 0]
TE-LM-ADMIT:tunnel 10.106.0.6 1_10002: “Path Admitted” -> “Admitting 1st Resv Leg”
```

```
TE-LM-ADMIT:tunnel 10.106.0.6 1_10002: “Admitting 1st Resv Leg” -> “Resv Admitted”
TE-LM-ADMIT:Admission control has granted Resv query for 10.106.0.6 1_10002 (10.112.0.12)
on link Ethernet4/0/1 [reason 0]
```

debug mpls traffic-eng link-management advertisements

To print information about resource advertisements for traffic engineering interfaces, use the **debugmplstraffic-englink-managementadvertisements** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management advertisements [detail] [*acl-number*]

no debug mpls traffic-eng link-management advertisements [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T. The **detail** keyword and the *acl-number*argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about resource advertisements for traffic engineering interfaces:

```
Router# debug mpls traffic-eng link-management advertisements detail
TE-LM-ADV:area isis level-1:IGP announcement:link Et4/0/1:info changed
TE-LM-ADV:area isis level-1:IGP msg:link Et4/0/1:includes subnet type (2), described nbrs
(1)
TE-LM-ADV:area isis level-1:IGP announcement:link Et4/0/1:info changed
TE-LM-ADV:area isis level-1:IGP msg:link Et4/0/1:includes subnet type (2), described nbrs
(1)
TE-LM-ADV:LSA:Flooding manager received message:link information change (Et4/0/1)
TE-LM-ADV:area isis level-1:*** Flooding node information ***
  System Information::
```

```
    Flooding Protocol:   ISIS
  Header Information::
    IGP System ID:       0001.0000.0001.00
    MPLS TE Router ID:   10.106.0.6
    Flooded Links:       1
  Link ID:: 0
    Link IP Address:     10.1.0.6
    IGP Neighbor:        ID 0001.0000.0001.02
    Admin. Weight:       10
    Physical Bandwidth:  10000 kbits/sec
    Max Reservable BW:   5000 kbits/sec
    Downstream::
      Reservable Bandwidth[0]:       5000 kbits/sec
      Reservable Bandwidth[1]:       2000 kbits/sec
      Reservable Bandwidth[2]:       2000 kbits/sec
      Reservable Bandwidth[3]:       2000 kbits/sec
      Reservable Bandwidth[4]:       2000 kbits/sec
      Reservable Bandwidth[5]:       2000 kbits/sec
      Reservable Bandwidth[6]:       2000 kbits/sec
Attribute Flags:  0x00000000
```

The following table describes the significant fields shown in the display.

Table 20: debug mpls traffic-eng link-management advertisements Field Descriptions

Field	Description
Flooding Protocol	Interior Gateway Protocol (IGB) that is flooding information for this area.
IGP System ID	Identification that IGP flooding uses in this area to identify this node.
MPLS TE Router ID	MPLS traffic engineering router ID.
Flooded Links	Number of links that are flooded in this area.
Link ID	Index of the link that is being described.
Link IP Address	Local IP address of this link.
IGP Neighbor	IGP neighbor on this link.
Admin. Weight	Administrative weight associated with this link.
Physical Bandwidth	Link's bandwidth capacity (in kbps).
Max Reservable BW	Maximum amount of bandwidth that is currently available for reservation at this priority.
Reservable Bandwidth	Amount of bandwidth that is available for reservation.
Attribute Flags	Attribute flags of the link being flooded.

debug mpls traffic-eng link-management bandwidth-allocation

To print detailed information about bandwidth allocation for traffic engineering label-switched paths (LSPs), use the **debug mpls traffic-eng link-management bandwidth-allocation** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management bandwidth-allocation [detail] [*acl-number*]

no debug mpls traffic-eng link-management bandwidth-allocation [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information only for those LSPs that match the access list.

Command Default

No default behavior or values.

Command Modes

PrivilegedEXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T. The **detail** keyword and the *acl-number*argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about bandwidth allocation for traffic engineering LSPs:

```
Router# debug mpls traffic-eng link-management bandwidth-allocation
TE-LM-BW:tunnel 10.106.0.6 1_10002:requesting Downstream bw hold (3000000 bps [S]) on link
 Et4/0/1
TE-LM-BW:tunnel 10.106.0.6 1_10002:Downstream bw hold request succeeded
TE-LM-BW:tunnel 10.106.0.6 1_10002:requesting Downstream bw lock (3000000 bps [S]) on link
 Et4/0/1
TE-LM-BW:tunnel 10.106.0.6 1_10002:Downstream bw lock request succeeded∞_„Rs
```

Related Commands

Command	Description
debug mpls traffic-eng link-management admission-control	Prints information about traffic engineering LSP admission control on traffic engineering interfaces.
debug mpls traffic-eng link-management errors	Prints information about errors encountered during any traffic engineering link management procedure.

debug mpls traffic-eng link-management errors

To print information about errors encountered during any traffic engineering link management procedure, use the **debug mpls traffic-eng link-management errors** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management errors [detail]

no debug mpls traffic-eng link-management errors [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(3)T	This command was introduced.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about errors encountered during a traffic engineering link management procedure:

```
Router# debug mpls traffic-eng link-management errors detail
00:04:48 TE-LM-ROUTING: link Et1/1/1: neighbor 0010.0000.0012.01: add to IP peer db failed
```

Related Commands

Command	Description
debug mpls traffic-eng link-management admission-control	Prints information about traffic engineering LSP admission control on traffic engineering interfaces.
debug mpls traffic-eng link-management advertisements	Prints information about resource advertisements for traffic engineering interfaces.

Command	Description
debug mpls traffic-eng link-management bandwidth-allocation	Prints information about bandwidth allocation for traffic engineering LSPs.
debug mpls traffic-eng link-management events	Prints information about traffic engineering link management system events.
debug mpls traffic-eng link-management igp-neighbors	Prints information about changes to the link management databases of IGP neighbors.
debug mpls traffic-eng link-management links	Prints information about traffic engineering link management interface events.

debug mpls traffic-eng link-management events

To print information about traffic engineering link management system events, use the **debug mpls traffic-eng link-management events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management events [detail]

no debug mpls traffic-eng link-management events [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T and the **detail** keyword was added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering link management system events:

```
Router# debug mpls traffic-eng link-management events detail
TE-LM-EVENTS:stopping MPLS TE Link Management process
TE-LM-EVENTS:MPLS TE Link Management process dying now
```

debug mpls traffic-eng link-management igp-neighbors

To print information about changes to the link management database of Interior Gateway Protocol (IGP) neighbors, use the **debug mpls traffic eng link-management igp-neighbors** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management igp-neighbors [detail]

no debug mpls traffic-eng link-management igp-neighbors [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T and the **detail** keyword was added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about changes to the link management database of IGP neighbors:

```
Router# debug mpls traffic-eng link-management igp-neighbors detail
TE-LM-NBR:link AT0/0.2:neighbor 0001.0000.0002.00:created (isis level-1, 10.42.0.10, Up)[total
 2]
```

Related Commands

Command	Description
debug mpls traffic-eng link-management events	Prints information about traffic engineering-related ISIS events.

debug mpls traffic-eng link-management links

To print information about traffic engineering link management interface events, use the **debug mpls traffic-eng link-management links** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management links [detail]

no debug mpls traffic-eng link-management links [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T and the **detail** keyword was added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering link management interface events:

```
Router# debug mpls traffic-eng link-management links detail
TE-LM-LINKS:link AT0/0.2:RSVP enabled
TE-LM-LINKS:link AT0/0.2:increasing RSVP bandwidth from 0 to 5000000
TE-LM-LINKS:link AT0/0.2:created [total 2]
TE-LM-LINKS:Binding MPLS TE LM Admission Control as the RSVP Policy Server on ATM0/0.2
TE-LM-LINKS:Bind attempt succeeded
TE-LM-LINKS:link AT0/0.2:LSP tunnels enabled
```

debug mpls traffic-eng link-management preemption

To print information about traffic engineering label-switched path (LSP) preemption, use the **debug mpls traffic-eng link-management preemption** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management preemption [detail]

no debug mpls traffic-eng link-management preemption [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(3)T	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering LSP preemption:

```
Router# debug mpls traffic-eng link-management preemption detail
TE-LM-BW:preempting Downstream bandwidth, 1000000, for tunnel 10.106.0.6 2_2
TE-LM-BW:building preemption list to get bandwidth, 1000000, for tunnel 10.106.0.6 2_2
(priority 0)
TE-LM-BW:added bandwidth, 3000000, from tunnel 10.106.0.6 1_2 (pri 1) to preemption list
TE-LM-BW:preemption list build to get bw, 1000000, succeeded (3000000)
TE-LM-BW:preempting bandwidth, 1000000, using plist with 1 tunnels
TE-LM-BW:tunnel 10.106.0.6 1_2:being preempted on AT0/0.2 by 10.106.0.6 2_2
TE-LM-BW:preemption of Downstream bandwidth, 1000000, succeeded
```

debug mpls traffic-eng link-management routing

To print information about traffic engineering link management routing resolutions that can be performed to help Resource Reservation Protocol (RSVP) interpret explicit route objects, use the **debug mpls traffic-eng link-management routing** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management routing [detail]

no debug mpls traffic-eng link-management routing [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T and the **detail** keyword was added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering link management routing resolutions that can be performed to help RSVP interpret explicit route objects:

```
Router# debug mpls traffic-eng link-management routing detail
TE-LM-ROUTING:route options to 10.42.0.10:building list (w/ nhop matching)
TE-LM-ROUTING:route options to 10.42.0.10:adding {AT0/0.2, 10.42.0.10}
TE-LM-ROUTING:route options to 10.42.0.10:completed list has 1 links
```

Related Commands

Command	Description
debug ip rsvp	Prints information about RSVP signalling events.

debug mpls traffic-eng load-balancing

To print information about unequal cost load balancing over traffic engineering tunnels, use the **debug mpls traffic-eng load-balancing** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng load-balancing

no debug mpls traffic-eng load-balancing

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)ST	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about unequal cost load balancing over traffic engineering tunnels:

```
Router# debug mpls traffic-eng load-balancing
TE-Load:10.210.0.0/16, 2 routes, loadbalancing based on MPLS TE bandwidth
TE-Load:10.200.0.0/16, 2 routes, loadbalancing based on MPLS TE bandwidth
```

debug mpls traffic-eng lsd-client

To display the Application Programming Interface (API) messages sent to the Label Switching Database (LSD) from the Traffic Engineering (TE) client, use the debug mpls traffic-eng lsd-client command in privileged EXEC mode. To disable the display of these messages, use the **no** form of this command.

debug mpls traffic-eng lsd-client

no debug mpls traffic-eng lsd-client

Syntax Description

This command has no arguments or keywords.

Command Default

Debugging is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(25)S	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB and implemented on the Cisco 10000 series routers.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(28)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(28)SXH.

Examples

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand enable TE globally:

```
00:10:23: TE-LSD-CLIENT: register with LSD OK; conn_id = 23, recov time = 60000 s
00:10:23: TE-LSD-CLIENT: LSD is now up
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand disable TE globally:

```
00:09:50: TE-LSD-CLIENT: unregister LSD client; result = OK; conn_id 23
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand enable TE on specific interfaces on Cisco 7500 series routers:

```
00:10:23: TE-LSD-CLIENT: enabled TE LSD client on Ethernet1/0; status = OK
00:10:23: TE-LSD-CLIENT: enabled TE LSD client on Serial2/0; status = OK
00:10:23: TE-LSD-CLIENT: enabled TE LSD client on Serial3/0; status = OK
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand disable TE on specific interfaces on Cisco 7500 series routers:

```
00:09:50: TE-LSD-CLIENT: disabled TE LSD client on Ethernet1/0; status = OK
00:09:50: TE-LSD-CLIENT: disabled TE LSD client on Serial2/0; status = OK
00:09:50: TE-LSD-CLIENT: disabled TE LSD client on Serial3/0; status = OK
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand enable TE on specific interfaces on Cisco 10000 series routers:

```
00:10:23: TE-LSD-CLIENT: enabled TE LSD client on GigabitEthernet1/0/0; status = OK
00:10:23: TE-LSD-CLIENT: enabled TE LSD client on Serial2/0/0; status = OK
00:10:23: TE-LSD-CLIENT: enabled TE LSD client on Serial3/0/0; status = OK
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand disable TE on specific interfaces on Cisco 10000 series routers:

```
00:09:50: TE-LSD-CLIENT: disabled TE LSD client on GigabitEthernet1/0/0; status = OK
00:09:50: TE-LSD-CLIENT: disabled TE LSD client on Serial2/0/0; status = OK
00:09:50: TE-LSD-CLIENT: disabled TE LSD client on Serial3/0/0; status = OK
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**command, allocate labels on tunnel midpoints, and create tunnel midpoint rewrites on Cisco 7500 series routers:

```
00:14:04: TE-LSD-CLIENT: label alloc OK; label = 16, conn_id = 23
00:14:04: TE-LSD-CLIENT: Create TE mid rewrite for 10.100.100.100 1 [5], Result: OK
00:14:04:          In: Serial3/0, 16 Out: Serial2/0, 3
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**command,allocate labels on tunnel midpoints, and create tunnel midpoint rewrites on a Cisco 10000 series router:

```
00:14:04: TE-LSD-CLIENT: label alloc OK; label = 16, conn_id = 23
00:14:04: TE-LSD-CLIENT: Create TE mid rewrite for 10.100.100.100 1 [5], Result: OK
00:14:04:          In: Serial3/0/0, 16 Out: Serial2/0/0, 3
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**command, free labels on tunnel midpoints, and delete tunnel midpoints on a Cisco 7500 series router:

```
00:13:13: TE-LSD-CLIENT: Delete TE mid rewrite for iou-100_t1, Result: OK
00:13:13: In: Serial3/0, 16 Out: Serial2/0, 1
00:13:13: TE-LSD-CLIENT: free label 16 result =  OK; conn_id = 23
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**command, free labels on tunnel midpoints, and delete tunnel midpoints on a Cisco 10000 series router:

```
00:13:13: TE-LSD-CLIENT: Delete TE mid rewrite for iou-100_t1, Result: OK
00:13:13: In: Serial3/0/0, 16 Out: Serial2/0/0, 1
00:13:13: TE-LSD-CLIENT: free label 16 result =  OK; conn_id = 23
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand create tunnel headend rewrites on a Cisco 7500 series router:

```
00:09:10: TE-LSD-CLIENT: Create TE he rewrite for iou-100_t1, Result = OK
00:09:10: tun_inst: 7  Out: Serial3/0, 16  Dest: 10.0.0.2
ps_flags: 0x60003
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand create tunnel headend rewrites on a Cisco 10000 series router:

```
00:09:10: TE-LSD-CLIENT: Create TE he rewrite for iou-100_t1, Result = OK
00:09:10: tun_inst: 7  Out: Serial3/0/0, 16  Dest: 10.0.0.2
ps_flags: 0x60003
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand delete tunnel headend rewrites on a Cisco 7500 series router:

```
00:09:15: TE-LSD-CLIENT: Delete TE he rewrite for iou-100_t1, Result: OK
00:09:15: tun_inst: 7  Out: Serial3/0, 16  ps_flags: 0x60003
```

The following messages are displayed when you issue the **debug mpls traffic-eng lsd-client**commandand delete tunnel headend rewrites on a Cisco 10000 series router:

```
00:09:15: TE-LSD-CLIENT: Delete TE he rewrite for iou-100_t1, Result: OK
00:09:15: tun_inst: 7  Out: Serial3/0/0, 16  ps_flags: 0x60003
```

Related Commands

Command	Description
debug mpls ip iprm events	Displays events related to the MPLS IPRM.
debug mpls ip iprm ldm	Displays debugging information for interactions between the IP LDMs and the MPLS IPRM.
debug mpls ip iprm mfi	Displays debugging information for interactions between the MFI and the MPLS IPRM.

debug mpls traffic-eng path

To display information about Multiprotocol Label Switching (MPLS) traffic engineering path calculation, use the **debug mpls traffic-eng path** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng path {**api**| **dump**| **errors**| *num*| **lookup**| **spf**| **verify**}

no debug mpls traffic-eng path {**api**| **dump**| **errors**| *num*| **lookup**| **spf**| **verify**}

Syntax Description

api	Displays path calculation application programming interface (API) events.
dump	Displays detailed path calculation information.
errors	Displays path calculation error event information.
num	The specific tunnel for which path calculation information is displayed. Valid values are 0-65535.
lookup	Displays information for path lookup events.
spf	Displays information for shortest path first (SPF) calculations.
verify	Displays information for path verifications.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(5)ST	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Release	Modification
12.2(33)SRE	This command was modified. The **api**, **dump**, and **errors**keywords were added.

Examples

In the following example, information is printed about the calculation of the traffic engineering path:

```
Router# debug mpls traffic-eng path lookup
TE-PCALC:Tunnel1000 Path Setup to 10.110.0.10:FULL_PATH
TE-PCALC:bw 0, min_bw 0, metric:0
TE-PCALC:setup_pri 0, hold_pri 0
TE-PCALC:affinity_bits 0x0, affinity_mask 0xFFFF
TE-PCALC_PATH:create_path_hoplist:ip addr 10.42.0.6 unknown.
```

debug mpls traffic-eng process-restart

To display information about process restarts for reporting to your technical support representative, use the **debug mpls traffic-eng process-restart** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng process-restart

no debug mpls traffic-eng process-restart

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(33)SXH	This command was introduced.

Usage Guidelines

If you report a problem and the **show mpls traffic-eng process-restart iprouting** displays abnormal results, your technical support representative might ask you to issue the **debug mpls traffic-eng process-restart** command, then perform an IP routing process restart and capture the output for analysis.

Examples

The following example shows partial output from an IP routing process restart:

```
Router# debug mpls traffic-eng process-restart
02:24:22: SM: ---TE ION Process Restart 0x78EF9050: process restart (3)
02:24:22: SM:    NORM (1) --> AWAIT-CFG (3)
02:24:22: TE ION Restart timer started, proc_idx:0 delay:120000
02:24:22: SM: ---TE ION Process Restart 0x78EF9050: process cfg replay start (4)
02:24:22: SM:    AWAIT-CFG (3) --> CFG (4)
02:24:22: TE ION Restart timer started, proc_idx:0 delay:300000
02:24:22: SM: ---TE ION Process Restart 0x78EF9050: reg invoke succeeded (2)
02:24:22: SM:    CFG (4) --> CFG (4)
02:24:22: SM: ---TE ION Process Restart 0x78EF9050: process cfg replay done (5)
02:24:22: SM:    CFG (4) --> SYNC (5)
02:24:22: TE ION Restart timer started, proc_idx:0 delay:900000
```

The output shows typical process restart information that your technical support representative might request if you report a problem after an IP routing process restart. The information displayed can vary, depending on the conditions that caused the restart.

Related Commands

Command	Description
show mpls traffic-eng process-restart iprouting	Displays the status of IP routing and MPLS traffic engineering synchronization after an IP routing process restarts.

debug mpls traffic-eng topology change

To print information about traffic engineering topology change events, use the **debug mpls traffic-eng topology change** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng topology change

no debug mpls traffic-eng topology change

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)ST	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about traffic engineering topology change events:

```
Router# debug mpls traffic-eng topology change
TE-PCALC_LSA:NODE_CHANGE_UPDATE isis level-1
        link flags:LINK_CHANGE_BW
        system_id:0001.0000.0001.00, my_ip_address:10.42.0.6
        nbr_system_id:0001.0000.0002.00, nbr_ip_address 10.42.0.10
```

debug mpls traffic-eng topology lsa

To print information about traffic engineering topology link state advertisement (LSA) events, use the **debug mpls traffic-eng topology lsa** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng topology lsa

no debug mpls traffic-eng topology lsa

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)ST	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, information is printed about traffic engineering topology LSA events:

```
Router# debug mpls traffic-eng topology lsa
TE-PCALC_LSA:node_lsa_add:Received a LSA:flags 0x1 !
IGP Id:0001.0000.0001.00, MPLS TE Id:10.106.0.6 is VALID has 2 links (frag_id 0)
      link[0 ]:Nbr IGP Id:0001.0000.0001.02
          frag_id 0, Intf Address:0.0.0.0
          admin_weight:10, attribute_flags:0x0
      link[1 ]:Nbr IGP Id:0001.0000.0002.00
          frag_id 0, Intf Address:10.42.0.6, Nbr Intf Address:10.42.0.10
          admin_weight:100, attribute_flags:0x0
TE-PCALC_LSA:(isis level-1):Received lsa:
IGP Id:0001.0000.0001.00, MPLS TE Id:10.106.0.6 Router Node  id 8
      link[0 ]:Nbr IGP Id:0001.0000.0002.00, nbr_node_id:9, gen:114
          frag_id 0, Intf Address:10.42.0.6, Nbr Intf Address:10.42.0.10
          admin_weight:100, attribute_flags:0x0
          physical_bw:155520 (kbps), max_reservable_bw:5000 (kbps)
               allocated_bw   reservable_bw      allocated_bw   reservable_bw
               ------------   -------------      ------------   -------------
        bw[0]:0              5000         bw[1]:3000           2000
        bw[2]:0              2000         bw[3]:0              2000
```

```
bw[4]:0             2000        bw[5]:0             2000
bw[6]:0             2000        bw[7]:0             2000
```

debug mpls traffic-eng tunnels errors

To print information about errors encountered during any traffic engineering tunnel management procedure, use the **debug mpls traffic-eng tunnels errors** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels errors [detail]

no debug mpls traffic-eng tunnels errors [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(3)T	This command was introduced.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about errors encountered during a traffic engineering tunnel management procedure:

```
Router# debug mpls traffic-eng tunnels errors

00:04:14: LSP-TUNNEL-SIG: Tunnel10012[1]: path verification failed (unprotected) [Can't use
 link 10.12.4.4 on node 10.0.0.4]
```

debug mpls traffic-eng tunnels events

To print information about traffic engineering tunnel management system events, use the **debug mpls traffic-eng tunnels events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels events [detail]

no debug mpls traffic-eng tunnels events [detail]

Syntax Description

detail	(Optional) Prints detailed debugging information.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T and the **detail** keyword was added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering tunnel management system events:

```
Router# debug mpls traffic-eng tunnels events detail
LSP-TUNNEL:received event:interface admin. down [Ethernet4/0/1]
LSP-TUNNEL:posting action(s) to all-tunnels:
              check static LSPs
LSP-TUNNEL:scheduling pending actions on all-tunnels
LSP-TUNNEL:applying actions to all-tunnels, as follows:
              check static LSPs
```

debug mpls traffic-eng tunnels labels

To print information about Multiprotocol Label Switching (MPLS) label management for traffic engineering tunnels, use the **debug mpls traffic-eng tunnels labels** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels labels [detail] [*acl-number*]

no debug mpls traffic-eng tunnels labels [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information only about traffic engineering tunnels that match the access list.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T, and the **detail** keyword and the *acl-number*argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about MPLS label management for traffic engineering tunnels:

```
Router# debug mpls traffic-eng tunnels labels detail
LSP-TUNNEL-LABELS:tunnel 10.106.0.6 1 [2]:fabric PROGRAM request
LSP-TUNNEL-LABELS:tunnel 10.106.0.6 1 [2]:programming label 16 on output interface ATM0/0.2
LSP-TUNNEL-LABELS:descriptor 71FA64:continuing “Program” request
LSP-TUNNEL-LABELS:descriptor 71FA64:set “Interface Point Out State” to, allocated
LSP-TUNNEL-LABELS:# of resource points held for “default” interfaces:2
LSP-TUNNEL-LABELS:descriptor 71FA64:set “Fabric State” to, enabled
LSP-TUNNEL-LABELS:descriptor 71FA64:set “Fabric Kind” to, default (LFIB)
```

```
LSP-TUNNEL-LABELS:descriptor 71FA64:set “Fabric State” to, set
LSP-TUNNEL-LABELS:tunnel 10.106.0.6 1 [2]:fabric PROGRAM reply
```

To restrict output to information about a single tunnel, you can configure an access list and supply it to the **debug** command. Configure the access list as follows:

```
Router(config-ext-nacl)# permit udp host scr_address host dst_address eq tun intfc
```

For example, if tunnel 10012 has destination 10.0.0.11 and source 10.0.0.4, as determined by the **show mpls traffic-eng tunnels command**, the following access list could be configured and added to the **debug** command:

```
Router(config-ext-nacl)# permit udp host 10.0.0.4 10.0.0.11 eq 10012
```

debug mpls traffic-eng tunnels reoptimize

To print information about traffic engineering tunnel reoptimizations, use the **debug mpls traffic-eng tunnels reoptimize** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels reoptimize [detail] [*acl-number*]

no debug mpls traffic-eng tunnels reoptimize [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information about only those traffic engineering tunnel reoptimizations that match the access list.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T, and the **detail** keyword and the *acl-number*argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering tunnel reoptimizations that match access list number 101:

```
Router# debug mpls traffic-eng tunnels reoptimize detail 101
LSP-TUNNEL-REOPT:Tunnel1 curr option 2 (0x6175CF8C), activate new option 2
LSP-TUNNEL-REOPT:Tunnel1 new path:option 2 [10002], weight 20
LSP-TUNNEL-REOPT:Tunnel1 old path:option 2 [2], weight 110
LSP-TUNNEL-REOPT:Tunnel1 [10002] set as reopt
LSP-TUNNEL-REOPT:Tunnel1 path option 2 [10002] installing as current
LSP-TUNNEL-REOPT:Tunnel1 [2] removed as current
```

```
LSP-TUNNEL-REOPT:Tunnel1 [2] set to delayed clean
LSP-TUNNEL-REOPT:Tunnel1 [10002] removed as reopt
LSP-TUNNEL-REOPT:Tunnel1 [10002] set to current
```

debug mpls traffic-eng tunnels signalling

To print information about traffic engineering tunnel signalling operations, use the **debug mpls traffic-eng tunnels signalling** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels signalling [detail] [*acl-number*]

no debug mpls traffic-eng tunnels signalling [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information about only those traffic engineering tunnel signalling operations that match the access list.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T, and the **detail** keyword and the *acl-number*argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering tunnel signalling operations that match access list number 101:

```
Router# debug mpls traffic-eng tunnels signalling detail 101
LSP-TUNNEL-SIG:tunnel Tunnel1 [2]:RSVP head-end open
LSP-TUNNEL-SIG:tunnel Tunnel1 [2]:received Path NHOP CHANGE
LSP-TUNNEL-SIG:Tunnel1 [2]:first hop change:0.0.0.0 --> 10.1.0.10
LSP-TUNNEL-SIG:received ADD RESV request for tunnel 10.106.0.6 1 [2]
LSP-TUNNEL-SIG:tunnel 10.106.0.6 1 [2]:path next hop is 10.1.0.10 (Et4/0/1)
```

```
LSP-TUNNEL-SIG:Tunnel1 [2] notified of new label information
LSP-TUNNEL-SIG:sending ADD RESV reply for tunnel 10.106.0.6 1 [2]
```

debug mpls traffic-eng tunnels state

To print information about state maintenance for traffic engineering tunnels, use the **debug mpls traffic-eng tunnels state** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels state [detail] [*acl-number*]

no debug mpls traffic-eng tunnels state [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information about state maintenance for traffic engineering tunnels that match the access list.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(3)T	This command was introduced.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about state maintenance for traffic engineering tunnels that match access list number 99:

```
Router# debug mpls traffic-eng tunnels state detail 99
LSP-TUNNEL:tunnel 10.106.0.6 1 [2]: “Connected” -> “Disconnected”
LSP-TUNNEL:Tunnel1 received event:LSP has gone down
LSP-TUNNEL:tunnel 10.106.0.6 1 [2]: “Disconnected” -> “Dead”
LSP-TUNNEL-SIG:Tunnel1:changing state from up to down
LSP-TUNNEL:tunnel 10.106.0.6 1 [2]: “Dead” -> “Connected”
```

debug mpls traffic-eng tunnels timers

To print information about traffic engineering tunnel timer management, use the **debug mpls traffic-eng tunnels timers** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng tunnels timers [detail] [*acl-number*]

no debug mpls traffic-eng tunnels timers [detail] [*acl-number*]

Syntax Description

detail	(Optional) Prints detailed debugging information.
acl-number	(Optional) Uses the specified access list to filter the debugging information. Prints information about traffic engineering tunnel timer management that matches the access list.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)S	This command was introduced.
12.1(3)T	This command was integrated into Cisco IOS Release 12.1(3)T, and the **detail** keyword and the *acl-number*argument were added.
12.0(10)ST	This command was integrated into Cisco IOS Release 12.0(10)ST.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

In the following example, detailed debugging information is printed about traffic engineering tunnel timer management:

```
Router# debug mpls traffic-eng tunnels timers detail
LSP-TUNNEL-TIMER:timer fired for Action Scheduler
LSP-TUNNEL-TIMER:timer fired for Tunnel Head Checkup
```

debug mpls vpn ha

To enable the display of Virtual Private Network (VPN) high availability (HA) debugging information, use the **debug mpls vpn ha**command in privileged EXEC mode. To disable the display of VPN HA debugging information, use the **no** form of this command.

debug mpls vpn ha

no debug mpls vpn ha

Syntax Description

This command has no arguments or keywords.

Command Default

VPN HA debugging is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(33)SRA	This command was introduced.
12.2(31)SB	This command was integrated into Cisco IOS Release 12.2(31)SB.

Examples

The following example shows sample output from the debug mpls vpn ha command:

```
Router# debug mpls vpn ha
VPN HA debugging is on.
```

debug mpls xtagatm cross-connect

Note Effective Cisco IOS Release 12.4(20)T, the **debugmplsxtagatmcross-connect** command is not available in Cisco IOS software.

To display requests and responses for establishing and removing cross-connects on the controlled ATM switch, use the **debugmplsxtagatmcross-connect**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls xtagatm cross-connect

no debug mpls xtagatm cross-connect

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.2(4)T	This command was updated to reflect the Multiprotocol Label Switching (MPLS) Internet Engineering Task Force (IETF) terminology.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(20)T	This command was removed.

Usage Guidelines

This command monitors requests to establish or remove cross-connects from XmplsATM interfaces to the Virtual Switch Interface (VSI) master, as well as the VSI master responses to these requests.

Note Use this command with care, because it generates output for each cross-connect operation performed by the label switch controller (LSC). In a network configuration with many label virtual circuits (LVCs), the volume of output generated can interfere with system timing and the proper operation of other router functions. Use this command only in situations in which the LVC setup or teardown rate is low.

Examples

The following is sample output from the **debugmplsxtagatmcross-connect** command:

```
Router# debug mpls xtagatm cross-connect
XTagATM: cross-conn request; SETUP, userdata 0x17, userbits 0x1, prec 7
        0xC0100 (Ctl-If) 1/32 <-> 0xC0200 (XTagATM0) 0/32
XTagATM: cross-conn response; DOWN, userdata 0x60CDCB5C, userbits 0x2, result
OK
        0xC0200 1/37 --> 0xC0300 1/37
```

The following table describes the significant fields shown in the display.

Table 21: debug mpls xtagatm cross-connect Field Descriptions

Field	Description
XTagATM	The source of the debugging message as an XmplsATM interface.
cross-conn	An indicator that the debugging message pertains to a cross-connect setup or teardown operation.
request	A request from an XmplsATM interface to the VSI master to set up or tear down a cross-connect.
response	Response from the VSI master to an XmplsATM interface that a cross-connect was set up or removed.
SETUP	A request for the setup of a cross-connect.
TEARDOWN	A request for the teardown of a cross-connect.
UP	The cross-connect is established.
DOWN	The cross-connect is not established.
userdata, userbits	Values passed with the request that are returned in the corresponding fields in the matching response.
prec	The precedence for the cross-connect.
result	The status of the completed request.

Field	Description
0xC0100 (Ctl-If) 1/32	Information about the interface: • One endpoint of the cross-connect is on the interface whose logical interface number is 0xC0100. • The interface is the VSI control interface. • The virtual path identifier (VPI) value at this endpoint is 1. • The virtual channel identifier (VCI) value at this end of the cross-connect is 32.
<->	The type of cross-connect (unidirectional or bidirectional).
0xC0200 (XTagATM0) 0/32	Information about the interface: • The other endpoint of the cross-connect is on the interface whose logical interface number is 0xC0200. • The interface is associated with XmplsATM interface 0. • The VPI value at this endpoint is 0. • The VCI value at this end of the cross-connect is 32.
->	The response pertains to a unidirectional cross-connect.

Related Commands

Command	Description
show xtagatm cross-connect	Displays information about remotely connected ATM switches.

debug mpls xtagatm errors

Note Effective with Cisco IOS Release 12.4(20)T, the **debug mpls xtagatm errors** command is not available in Cisco IOS software.

To display information about error and abnormal conditions that occur on XmplsATM interfaces, use the **debug mpls xtagatm errors** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls xtagatm errors

no debug mpls xtagatm errors

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.2(4)T	This command was updated to reflect the Multiprotocol Label Switching (MPLS) Internet Engineering Task Force (IETF) terminology.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(20)T	This command was removed.

Usage Guidelines

Use the **debug mpls xtagatm errors** command to display information about abnormal conditions and events that occur on XmplsATM interfaces.

Examples

The following is sample output from the **debug mpls xtagatm errors** command:

```
Router# debug mpls xtagatm errors
XTagATM VC: XTagATM0 1707 2/352 (ATM1/0 1769 3/915): Cross-connect setup
failed NO_RESOURCES
```

This message indicates a failed attempt to set up a cross-connect for a terminating a virtual circuit (VC) on XmplsATM0. The reason for the failure was a lack of resources on the controlled ATM switch.

debug mpls xtagatm events

Note Effective with Cisco IOS Release 12.4(20)T, the**debugmplsxtagatmevents** command is not available in Cisco IOS software.

To display information about major events that occur on XmplsATM interfaces, not including events for specific XmplsATM virtual circuits (VCs) and switch cross-connects, use the **debugmplsxtagatmevents**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls xtagatm events

no debug mpls xtagatm events

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Command	Modification
12.0(5)T	This command was introduced.
12.2(4)T	This command was updated to reflect the Multiprotocol Label Switching (MPLS) Internet Engineering Task Force (IETF) terminology.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.4(20)T	This command was removed.

Usage Guidelines

Use the **debugmplsxtagatmevents** command to monitor major events that occur on XmplsATM interfaces. This command monitors events that pertain only to XmplsATM interfaces as a whole and does not include any events that pertain to individual XmplsATM VCs or individual switch cross-connects. The specific events that are monitored when the **debugmplsxtagatmevents** command is in effect include the following:

- Receiving asynchronous notifications that the VSI master sent through the external ATM application programming interface (ExATM API) to an XmplsATM interface.

- Resizing of the table that is used to store switch cross-connect information. This table is resized automatically as the number of cross-connects increases.
- Marking of XmplsATM VCs as stale when an XmplsATM interface shuts down, thereby ensuring that the stale interfaces are refreshed before new XmplsATM VCs can be created on the interface.

Examples

The following is sample output from the **debugmplsxtagatmevents** command:

```
Router# debug mpls xtagatm events
XTagATM: desired cross-connect table size set to 256
XTagATM: ExATM API intf event Up, port 0xA0100 (None)
XTagATM: ExATM API intf event Down, port 0xA0100 (None)
XTagATM: marking all VCs stale on XTagATM0
```

The following table describes the significant fields shown in the display.

Table 22: debug mpls xtagatm events Field Descriptions

Field	Description
XTagATM	The source of the debugging message.
desired cross-connect table size set to 256	The table of cross-connect information has been set to hold 256 entries. A single cross-connect table is shared among all XmplsATM interfaces. The cross-connect table is automatically resized as the number of cross-connects increases.
ExATM API	The information in the debug output pertains to an asynchronous notification sent by the Virtual Switch Interface (VSI) master to the XmplsATM driver.
event Up/Down	The specific event that was sent by the VSI master to the XmplsATM driver.
port 0xA0100 (None)	The event pertains to the VSI interface whose logical interface number is 0xA0100, and that this logical interface is not bound to an XmplsATM interface.
marking all VCs stale on XTagATM0	All existing XmplsATM VCs on interface XmplsATM0 are marked as stale, and that XmplsATM0 remains down until all of these VCs are refreshed.

debug mpls xtagatm vc

Note Effective with Cisco IOS Release 12.4(20)T, the **debugmplsxtagatmvc** command is not available in Cisco IOS software.

To display information about events that affect individual XmplsATM terminating virtual circuits (VCs), use the **debugmplsxtagatmvc**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls xtagatm vc

no debug mpls xtagatm vc

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.2(4)T	This command was updated to reflect the Multiprotocol Label Switching (MPLS) Internet Engineering Task Force (IETF) terminology.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.4(20)T	This command was removed.

Usage Guidelines

Use the **debugmplsxtagatmvc** command to display detailed information about all events that affect individual XmplsATM terminating VCs.

Note Use this command with care, because it results in extensive output when many XmplsATM VCs are set up or torn down. This output can interfere with system timing and normal operation of other router functions. Use the **debugmplsxtagatmvc** command only when a few XmplsATM VCs are created or removed.

Examples

The following is sample output from the **debugmplsxtagatmvc** command:

```
Router# debug mpls xtagatm vc
XTagATM VC: XTagATM1 18 0/32 (ATM1/0 0 0/0):  Setup,  Down --> UpPend
XTagATM VC: XTagATM1 18 0/32 (ATM1/0 88 1/32):  Complete,  UpPend --> Up
XTagATM VC: XTagATM1 19 1/33 (ATM1/0 0 0/0):  Setup,  Down --> UpPend
XTagATM VC: XTagATM0 43 0/32 (ATM1/0 67 1/84):  Teardown,  Up --> DownPend
```

The following table describes the significant fields shown in the display.

Table 23: debug mpls xtagatm vc Field Descriptions

Field	Description
XTagATM VC	The source of the debugging message.
XTagATM <ifnum>	The particular XmplsATM interface number for the terminating VC.
vcd vpi/vci	The virtual circuit descriptor (VCD) and virtual path identifier/virtual channel identifier (VPI/VCI) values for the terminating VC.
(ctl-if vcd vpi/vci)	The control interface, the VCD, and the VPI and VCI values for the private VC corresponding to the XmplsATM VC on the control interface.
Setup, Complete, Teardown	The name of the event that occurred for the indicated VC.
oldstate -> newstate	The state of the terminating VC before and after the processing of the event.

debug mpoa client

Note Effective with Cisco IOS Release 15.1M, the **debug mpoa client**command is not available in Cisco IOS software.

To display Multiprotocol over ATM (MPOA) client debug information, use the **debug mpoa client**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

MPOA:**debug mpoa client commanddebug mpoa client commanddebug mpoa client** {**all**| **data**| **egress**| **general**| **ingress**| **keep-alives**| **platform-specific**} [**name** *mpc-name*]

no debug mpoa client {**all**| **data**| **egress**| **general**| **ingress**| **keep-alives**| **platform-specific**} [**name** *mpc-name*]

Syntax Description

all	Displays debugging information for all MPC activity.
data	Displays debugging information for data plane activity only. This option applies only to routers.
egress	Displays debugging information for egress functionality only.
general	Displays general debugging information only.
ingress	Displays debugging information for ingress functionality only.
keep-alives	Displays debugging information for keep-alive activity only.
platform-specific	Displays debugging information for specific platforms only. This option applies only to the Catalyst 5000 series ATM module.
name *mpc-name*	(Optional) Specifies the name of the MPC with the specified name.

Command Default

Debugging is turned on for all MPOA Clients (MPCs).

Command Modes

Privileged EXEC

Command History

Release	Modification
11.3	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.1M	This command was removed.

Examples

The following shows how to turn on debugging for the MPC ip_mpc:

```
ATM# debug mpoa client all name ip_mpc
```

Related Commands

Command	Description
debug mpoa server	Displays information about the MPOA server.

debug mpoa server

Note Effective with Cisco IOS Release 15.1M, the **debug mpoa server**command is not available in Cisco IOS software.

To display information about the Multiprotocol over ATM (MPOA) server, use the **debug mpoa server** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

dMPOA:**debug mpoa server commanddebug mpoa server commandebug mpoa server** [**name** *mps-name*]

no dMPOA:**debug mpoa server commanddebug mpoa server commandebug mpoa server** [**name** *mps-name*]

Syntax Description

name *mps-name*	(Optional) Specifies the name of an MPOA server.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.3	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.1M	This command was removed.

Usage Guidelines

The **debug mpoa server**command optionally limits the output only to the specified MPOA Server (MPS).

Examples

The following turns on debugging only for the MPS named ip_mps:

```
Router# debug mpoa server name ip_mps
```

debug mrcp

To display debugging messages for Media Resource Control Protocol (MRCP) operations, use the **debug mrcp**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mrcp {**all**| **api**| **detail**| **error**| **pmh**| **session**| **socket**| **state**}

no debug mrcp {**all**| **api**| **detail**| **error**| **pmh**| **session**| **socket**| **state**}

Syntax Description

all	Displays all MRCP debugging messages.
api	Displays messages between the application and the MRCP stack.
detail	Displays detailed MRCP version 2 (MRCP v2) debugging messages.
error	Displays MRCP error messages.
pmh	Displays protocol message handler (PMH) messages.
session	Displays messages about active MRCP sessions.
socket	Displays MRCP v2 socket debugging messages
state	Displays Finite State Machine (FSM) messages.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced on the Cisco 3640, Cisco 3660, Cisco AS5300, Cisco AS5350, and Cisco AS5400.
12.4(15)T	This command was modified to support MRCP v2. The **detail** and **socket** keywords were added.

Examples

The following example shows output from the **debug mrcp api**command:

```
Router# debug mrcp api
```

The first four lines show Real Time Streaming Protocol (RTSP) socket commands for Text-To-Speech (TTS) operations:

```
*Apr 17 16:31:16.323:mrcp_add_param:param:Kill-On-Barge-In:
*Apr 17 16:31:16.323:mrcp_add_param:param:Speech-Language:
*Apr 17 16:31:16.323:mrcp_add_param:param:Logging-Tag:
*Apr 17 16:31:16.323:mrcp_add_param:param:Content-Base:
*Apr 17 16:31:16.323:mrcp_create_session:same host/port
*Apr 17 16:31:16.323:mrcp_associate_call 5 10
*Apr 17 16:31:16.323:mrcp_associate_call 5 10
*Apr 17 16:31:16.323:mrcp_synth_speak 5
*Apr 17 16:31:16.323:mrcp_add_param:param:Content-Base:
*Apr 17 16:31:16.323:mrcp_recognizer_define_grammar 5
```

The following lines show RTSP socket commands for Automatic Speech Recognition (ASR) operations:

```
*Apr 17 16:31:16.323:mrcp_add_param:param:Confidence-Threshold:
*Apr 17 16:31:16.323:mrcp_add_param:param:Sensitivity-Level:
*Apr 17 16:31:16.323:mrcp_add_param:param:Speed-Vs-Accuracy:
*Apr 17 16:31:16.323:mrcp_add_param:param:Dtmf-Interdigit-Timeout:
*Apr 17 16:31:16.323:mrcp_add_param:param:Dtmf-Term-Timeout:
*Apr 17 16:31:16.323:mrcp_add_param:param:Dtmf-Term-Char:
*Apr 17 16:31:16.323:mrcp_add_param:param:No-Input-Timeout:
*Apr 17 16:31:16.323:mrcp_add_param:param:Logging-Tag:
*Apr 17 16:31:16.327:mrcp_add_param:param:Content-Base:
*Apr 17 16:31:16.327:mrcp_add_param:param:Recognizer-Start-Timers:
*Apr 17 16:31:16.327:mrcp_recognizer_start 5
*Apr 17 16:31:26.715:mrcp_add_param:param:Kill-On-Barge-In:
*Apr 17 16:31:26.715:mrcp_add_param:param:Speech-Language:
*Apr 17 16:31:26.715:mrcp_add_param:param:Logging-Tag:
*Apr 17 16:31:26.715:mrcp_add_param:param:Content-Base:
*Apr 17 16:31:26.715:mrcp_synth_speak 5
*Apr 17 16:31:30.451:mrcp_destroy_session 5 type:SYNTHESIZER
*Apr 17 16:31:30.451:mrcp_destroy_session 5 type:RECOGNIZER
```

The following examples show output from the **debug mrcp error** command:

```
Router# debug mrcp error
```

This output shows an error when the response from the server is incorrect:

```
*May  9 20:29:09.936:Response from 10.1.2.58:554 failed
*May  9 20:29:09.936:MRCP/1.0 71 422 COMPLETE
```

This output shows an error when the RTSP connection to the server fails:

```
*May  9 20:29:09.936:Connecting to 10.1.2.58:554 failed
```

This output shows an error when the recognize request comes out of sequence:

```
*May  9 20:29:09.936:act_idle_recognize:ignoring old recognize request
```

The following example shows output from the **debug mrcp pmh** command:

```
Router# debug mrcp pmh
*Apr 17 16:32:51.777:param:Kill-On-Barge-In: true
*Apr 17 16:32:51.777:param:Speech-Language: en-US
*Apr 17 16:32:51.777:param:Logging-Tag: 14:14
*Apr 17 16:32:51.777:param:Content-Base: http://server-asr/
*Apr 17 16:32:51.777:param:Content-Base: http://server-asr/
*Apr 17 16:32:51.777:param:Confidence-Threshold: 50
*Apr 17 16:32:51.781:param:Sensitivity-Level: 50
*Apr 17 16:32:51.781:param:Speed-Vs-Accuracy: 50
*Apr 17 16:32:51.781:param:Dtmf-Interdigit-Timeout: 10000
*Apr 17 16:32:51.781:param:Dtmf-Term-Timeout: 10000
*Apr 17 16:32:51.781:param:Dtmf-Term-Char: #
*Apr 17 16:32:51.781:param:No-Input-Timeout: 10000
*Apr 17 16:32:51.781:param:Logging-Tag: 14:14
*Apr 17 16:32:51.781:param:Content-Base: http://server-asr/
*Apr 17 16:32:51.781:param:Recognizer-Start-Timers: false
*Apr 17 16:32:51.877:GRAMMAR-CONTENT-HEADER
```

```
*Apr 17 16:32:51.877:Content-Type:application/grammar+xml
Content-Id:field2@field.grammar
Content-Length:356
*Apr 17 16:32:51.885:GRAMMAR-CONTENT-HEADER
*Apr 17 16:32:51.885:Content-Type:text/uri-list
Content-Length:30
*Apr 17 16:32:51.885:Total-Length=365
*Apr 17 16:32:51.885:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
*Apr 17 16:32:51.885:RECOGNIZE  20 MRCP/1.0
Confidence-Threshold:50
Sensitivity-Level:50
Speed-Vs-Accuracy:50
Dtmf-Interdigit-Timeout:10000
Dtmf-Term-Timeout:10000
Dtmf-Term-Char:#
No-Input-Timeout:10000
Logging-Tag:14:14
Content-Base:http://server-asr/
Recognizer-Start-Timers:false



*Apr 17 16:32:51.885:Content-Type:text/uri-list
Content-Length:30
*Apr 17 16:32:51.885:session:field2@field.grammar
*Apr 17 16:32:51.885:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
*Apr 17 16:32:51.889:SPEECH-MARKUP-TYPE-HEADER
*Apr 17 16:32:51.889:Content-Type:application/synthesis+ssml
Content-Length:126
*Apr 17 16:32:51.889:Total-Length=313
*Apr 17 16:32:51.889:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
*Apr 17 16:32:51.889:SPEAK  18 MRCP/1.0
Kill-On-Barge-In:true
Speech-Language:en-US
Logging-Tag:14:14
Content-Base:http://server-asr/



*Apr 17 16:32:51.889:Content-Type:application/synthesis+ssml
Content-Length:126
*Apr 17 16:32:51.889:<?xml version="1.0"?><speak> Who do you want speak to?? Joe, Carl,
Alex?. And I am extending the length of the text</speak>
*Apr 17 16:32:51.889:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
*Apr 17 16:32:51.925:mrcp_pmh_parse_response:Length:28
 Apr 17 16:32:51.925:mrcp_pmh_get_request_line:Line:MRCP/1.0 19 200 COMPLETE
*Apr 17 16:32:51.925:Request-tag:19 resp-code:200 Status:COMPLETE
*Apr 17 16:32:51.925:No Of Properties:0
*Apr 17 16:32:51.925:mrcp_process_recog_response:
*Apr 17 16:32:51.933:mrcp_pmh_parse_response:Length:31
 Apr 17 16:32:51.933:mrcp_pmh_get_request_line:Line:MRCP/1.0 20 200 IN-PROGRESS
*Apr 17 16:32:51.933:Request-tag:20 resp-code:200 Status:IN-PROGRESS
*Apr 17 16:32:51.933:No Of Properties:0
*Apr 17 16:32:51.933:mrcp_process_recog_response:
*Apr 17 16:32:53.413:mrcp_pmh_parse_response:Length:31
 Apr 17 16:32:53.413:mrcp_pmh_get_request_line:Line:MRCP/1.0 18 200 IN-PROGRESS
*Apr 17 16:32:53.413:Request-tag:18 resp-code:200 Status:IN-PROGRESS
*Apr 17 16:32:53.413:No Of Properties:0
*Apr 17 16:32:53.413:mrcp_process_synth_response:
*Apr 17 16:33:01.685:mrcp_pmh_parse_response:Length:100
 Apr 17 16:33:01.689:mrcp_pmh_get_event_line:Line:SPEAK-COMPLETE 18 COMPLETE MRCP/1.0
*Apr 17 16:33:01.689:Request-tag:18 resp-code:200 Status:COMPLETE
*Apr 17 16:33:01.689:No Of Properties:2
*Apr 17 16:33:01.689:mrcp_process_synth_events:
*Apr 17 16:33:01.689: COMPLETION-CAUSE:1
*Apr 17 16:33:01.689:mrcp_send_synth_app_response:
*Apr 17 16:33:01.689:mrcp_pmh_parse_response:Length:61
 Apr 17 16:33:01.689:mrcp_pmh_get_event_line:Line:START-OF-SPEECH 20 IN-PROGRESS MRCP/1.0
*Apr 17 16:33:01.689:Request-tag:20 resp-code:200 Status:IN-PROGRESS
```

```
*Apr 17 16:33:01.689:No Of Properties:1
*Apr 17 16:33:01.689:mrcp_process_recog_events:
*Apr 17 16:33:02.653:mrcp_pmh_parse_response:Length:815
 Apr 17 16:33:02.653:mrcp_pmh_get_event_line:Line:RECOGNITION-COMPLETE 20 COMPLETE MRCP/1.0
*Apr 17 16:33:02.653:Request-tag:20 resp-code:200 Status:COMPLETE
*Apr 17 16:33:02.653:No Of Properties:2
*Apr 17 16:33:02.653:mrcp_process_recog_events:
*Apr 17 16:33:02.653: COMPLETION-CAUSE:0
*Apr 17 16:33:02.653:mrcp_send_recog_app_response:
*Apr 17 16:33:02.661:param:Kill-On-Barge-In: true
*Apr 17 16:33:02.661:param:Speech-Language: en-US
*Apr 17 16:33:02.661:param:Logging-Tag: 14:14
*Apr 17 16:33:02.661:param:Content-Base: http://server-asr/
*Apr 17 16:33:02.665:SPEECH-MARKUP-TYPE-HEADER
*Apr 17 16:33:02.665:Content-Type:application/synthesis+ssml
Content-Length:57
*Apr 17 16:33:02.665:Total-Length=243
*Apr 17 16:33:02.665:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
*Apr 17 16:33:02.665:SPEAK  22 MRCP/1.0
Kill-On-Barge-In:true
Speech-Language:en-US
Logging-Tag:14:14
Content-Base:http://server-asr/
```

```
*Apr 17 16:33:02.665:Content-Type:application/synthesis+ssml
Content-Length:57
*Apr 17 16:33:02.665:<?xml version="1.0"?><speak> You have joe mails</speak>
*Apr 17 16:33:02.665:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
*Apr 17 16:33:02.833:mrcp_pmh_parse_response:Length:31
 Apr 17 16:33:02.833:mrcp_pmh_get_request_line:Line:MRCP/1.0 22 200 IN-PROGRESS
*Apr 17 16:33:02.833:Request-tag:22 resp-code:200 Status:IN-PROGRESS
*Apr 17 16:33:02.833:No Of Properties:0
*Apr 17 16:33:02.833:mrcp_process_synth_response:
*Apr 17 16:33:06.382:mrcp_pmh_parse_response:Length:98
 Apr 17 16:33:06.382:mrcp_pmh_get_event_line:Line:SPEAK-COMPLETE 22 COMPLETE MRCP/1.0
*Apr 17 16:33:06.382:Request-tag:22 resp-code:200 Status:COMPLETE
*Apr 17 16:33:06.382:No Of Properties:2
*Apr 17 16:33:06.382:mrcp_process_synth_events:
*Apr 17 16:33:06.382: COMPLETION-CAUSE:0
*Apr 17 16:33:06.382:mrcp_send_synth_app_response:
```

The following example shows output from the **debug mrcp session**command:

```
Router# debug mrcp session
*Apr 17 16:34:07.851:mrcp_create_session:
*Apr 17 16:34:07.851:mrcp_create_session:New SCB creation
*Apr 17 16:34:07.851:mrcp_create_svr_session_url:
*Apr 17 16:34:07.851:mrcp_create_session:
*Apr 17 16:34:07.851:mrcp_create_session:Already an  SCB is created for this call
*Apr 17 16:34:07.851:mrcp_process_events:event:LIB_CONNECT SYNTHESIZERCONN-STATUS=0
*Apr 17 16:34:07.855:mrcp_process_events:event:SPEAK SYNTHESIZER
*Apr 17 16:34:07.855:mrcp_process_events:event:SPEAK defered
*Apr 17 16:34:07.855:mrcp_process_events:event:LIB_CONNECT RECOGNIZERCONN-STATUS=0
*Apr 17 16:34:07.855:mrcp_process_events:event:DEFINE_GRAMMAR RECOGNIZER
*Apr 17 16:34:07.855:mrcp_process_events:event:DEFINE_GRAMMAR defered
*Apr 17 16:34:07.855:mrcp_process_events:event:LIB_CONNECT RECOGNIZERCONN-STATUS=0
*Apr 17 16:34:07.855:mrcp_process_events:event:RECOGNIZE RECOGNIZER
*Apr 17 16:34:07.855:mrcp_process_events:event:RECOGNIZE defered
*Apr 17 16:34:07.855:mrcp_response_handler:status=RTSPLIB_STATUS_SERVER_CONNECTED
*Apr 17 16:34:07.855:mrcp_process_events:event:LIB_CONNECTED SYNTHESIZERCONN-STATUS=4
*Apr 17 16:34:07.947:mrcp_response_handler:status=RTSPLIB_STATUS_RTP_RECORD_SETUP
*Apr 17 16:34:07.947:mrcp_process_events:event:RECOG_RTP_SETUP RECOGNIZER
*Apr 17 16:34:07.947:mrcp_process_defered_events:event:DEFINE_GRAMMAR
*Apr 17 16:34:07.947:mrcp_process_defered_events:event:RECOGNIZECONN-STATUS=2
*Apr 17 16:34:07.971:mrcp_response_handler:status=RTSPLIB_STATUS_RECORD_ASSOCIATED
*Apr 17 16:34:07.971:mrcp_response_handler:status=RTSPLIB_STATUS_RTP_PLAY_SETUP
*Apr 17 16:34:07.975:mrcp_process_events:event:RECOGNIZER_ASSOCIATED RECOGNIZER
*Apr 17 16:34:07.975:mrcp_process_events:event:SYNTH_RTP_SETUP SYNTHESIZER
```

```
*Apr 17 16:34:07.975:mrcp_process_defered_events:event:SPEAKCONN-STATUS=1
*Apr 17 16:34:07.975:mrcp_response_handler:status=RTSPLIB_STATUS_PLAY_ASSOCIATED
*Apr 17 16:34:07.975:mrcp_process_events:event:SYNTHESIZER_ASSOCIATED SYNTHESIZER
*Apr 17 16:34:08.007:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:08.019:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:08.059:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:17.611:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:17.611:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:17.611:mrcp_process_events:event:SPEECH_COMPLETE SYNTHESIZER
*Apr 17 16:34:17.611:mrcp_process_events:event:START_OF_SPEECH RECOGNIZER
*Apr 17 16:34:18.575:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:18.575:mrcp_process_events:event:RECOGNITION_COMPLETE RECOGNIZER
*Apr 17 16:34:18.583:mrcp_process_events:event:SPEAK SYNTHESIZER
*Apr 17 16:34:18.587:mrcp_response_handler:status=RTSPLIB_STATUS_PLAY_ASSOCIATED
*Apr 17 16:34:18.587:mrcp_process_events:event:SYNTHESIZER_ASSOCIATED SYNTHESIZER
*Apr 17 16:34:18.763:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:22.279:mrcp_response_handler:status=RTSPLIB_STATUS_RESP_OK
*Apr 17 16:34:22.283:mrcp_process_events:event:SPEECH_COMPLETE SYNTHESIZER
*Apr 17 16:34:22.307:mrcp_process_events:event:LIB_DESTROY SYNTHESIZERCONN-STATUS=12
*Apr 17 16:34:22.311:mrcp_process_events:event:LIB_DESTROY RECOGNIZERCONN-STATUS=12
*Apr 17 16:34:22.311:mrcp_response_handler:status=RTSPLIB_STATUS_CLEANUP
*Apr 17 16:34:22.315:mrcp_free_fsm:
*Apr 17 16:34:22.315:mrcp_free_scb:
*Apr 17 16:34:22.315:mrcp_create_session_history:scb=0x62C712F4
*Apr 17 16:34:22.315:mrcp_insert_session_history_record:current=0x62999544, callID=0x12
*Apr 17 16:34:22.315:mrcp_insert_session_history_record:count = 3
*Apr 17 16:34:22.315:mrcp_insert_session_history_record:starting history record deletion_timer
 of 10 minutes
```

The following example shows output from the **debug mrcp state** command:

```
Router# debug mrcp state
*Apr 17 16:35:25.141:mrcp_add_synthesizer_fsm:adding synthesizer fsm
*Apr 17 16:35:25.141:mrcp_add_connection_fsm:adding connection fsm
*Apr 17 16:35:25.141:mrcp_add_rtpsetup_fsm:adding rtpsetup fsm
*Apr 17 16:35:25.145:hash_get:  key=7
*Apr 17 16:35:25.145:mrcp_add_recognizer_fsm:adding recognizer fsm
*Apr 17 16:35:25.145:mrcp_add_connection_fsm:adding connection fsm
*Apr 17 16:35:25.145:mrcp_add_rtpsetup_fsm:adding rtpsetup fsm
*Apr 17 16:35:25.145:mrcp_fsm_execute:type=SYNTHESIZER
```

The following lines show the gateway connecting to the TTS server:

```
*Apr 17 16:35:25.145:    curr[CONNECT_IDLE]  ev-id[LIB_CONNECT]
    next[CONNECTING] action=610B8FD00
*Apr 17 16:35:25.145:act_idle_libconnect
*Apr 17 16:35:25.145:mrcp_shortcut_connection_fsm
*Apr 17 16:35:25.149:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:25.149:    curr[CONNECTING]  ev-id[LIB_CONNECT_PENDING]
    next[CONNECTING] action=610B90F80
*Apr 17 16:35:25.149:act_connecting_libpending
*Apr 17 16:35:25.149:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:25.149:    curr[CONNECTING]  ev-id[LIB_CONNECT]
    next[CONNECTING] action=610B8D480
*Apr 17 16:35:25.149:act_connectfsm_error
*Apr 17 16:35:25.149:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:25.149:    curr[CONNECTING]  ev-id[LIB_CONNECT]
```

The following lines show the gateway successfully connected to the TTS server:

```
    next[CONNECTING] action=610B8D480
*Apr 17 16:35:25.149:act_connectfsm_error
*Apr 17 16:35:25.149:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:25.149:    curr[CONNECTING]  ev-id[LIB_CONNECTED]
    next[CONNECTED] action=610B913C0
*Apr 17 16:35:25.149:act_connecting_libconnected
*Apr 17 16:35:25.149:act_rtpsetupfsm_libdescribed
*Apr 17 16:35:25.237:mrcp_fsm_execute:type=RESOURCE_NONE
*Apr 17 16:35:25.237:    curr[RTP_IDLE]  ev-id[RECOG_RTP_SETUP]
    next[RTP_RECOG_SETUP_DONE] action=610B94F40
*Apr 17 16:35:25.237:act_idle_recog_rtpsetup
*Apr 17 16:35:25.237:mrcp_fsm_execute:type=RECOGNIZER
*Apr 17 16:35:25.237:    curr[RECOG_IDLE]  ev-id[DEFINE_GRAMMAR]
```

```
    next[RECOG_IDLE] action=610B99340
*Apr 17 16:35:25.237:act_idle_define_grammar:
*Apr 17 16:35:25.237:hash_add:  key=31
*Apr 17 16:35:25.237:mrcp_fsm_execute:type=RECOGNIZER
*Apr 17 16:35:25.237:    curr[RECOG_IDLE]  ev-id[RECOGNIZE]
    next[RECOG_ASSOCIATING] action=610B98400
*Apr 17 16:35:25.237:act_idle_recognize:
*Apr 17 16:35:25.245:mrcp_fsm_execute:type=RECOGNIZER
*Apr 17 16:35:25.245:    curr[RECOG_ASSOCIATING]  ev-id[RECOGNIZER_ASSOCIATED]
    next[RECOGNIZING] action=610B9AB40
*Apr 17 16:35:25.245:act_associating_recognizer_associated:
*Apr 17 16:35:25.249:hash_add:  key=32
*Apr 17 16:35:25.249:mrcp_fsm_execute:type=RESOURCE_NONE
*Apr 17 16:35:25.249:    curr[RTP_IDLE]  ev-id[SYNTH_RTP_SETUP]
    next[RTP_SYNTH_SETUP_DONE] action=610B93D40
*Apr 17 16:35:25.249:act_idle_synth_rtpsetup
*Apr 17 16:35:25.249:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:25.249:    curr[SYNTH_IDLE]  ev-id[SPEAK]
    next[SYNTH_ASSOCIATING] action=610BA5540
*Apr 17 16:35:25.249:act_idle_speak
*Apr 17 16:35:25.249:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:25.249:    curr[SYNTH_ASSOCIATING]  ev-id[SYNTHESIZER_ASSOCIATED]
```

The following lines show the TTS server performing speech synthesis:

```
    next[SPEAKING] action=610BA7B40
*Apr 17 16:35:25.249:act_associating_speak_associated
*Apr 17 16:35:25.249:hash_add:  key=30
*Apr 17 16:35:25.285:hash_get:  key=31
*Apr 17 16:35:25.285:hash_delete:  key=31
*Apr 17 16:35:25.293:hash_get:  key=32
*Apr 17 16:35:25.293:hash_get:  key=30
*Apr 17 16:35:32.805:hash_get:  key=30
*Apr 17 16:35:32.805:hash_delete:  key=30
*Apr 17 16:35:32.805:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:32.805:    curr[SPEAKING]  ev-id[SPEECH_COMPLETE]
    next[SYNTH_IDLE] action=610BAA680
*Apr 17 16:35:32.805:act_speaking_speech_complete
*Apr 17 16:35:32.809:hash_get:  key=32
*Apr 17 16:35:32.809:mrcp_fsm_execute:type=RECOGNIZER
*Apr 17 16:35:32.809:    curr[RECOGNIZING]  ev-id[START_OF_SPEECH]
    next[RECOGNIZING] action=610B9F3C0
*Apr 17 16:35:32.809:act_recognizing_start_of_speech
*Apr 17 16:35:33.781:hash_get:  key=32
*Apr 17 16:35:33.781:hash_delete:  key=32
*Apr 17 16:35:33.781:mrcp_fsm_execute:type=RECOGNIZER
*Apr 17 16:35:33.781:    curr[RECOGNIZING]  ev-id[RECOGNITION_COMPLETE]
    next[RECOGNIZED] action=610B9D240
*Apr 17 16:35:33.781:act_recognizing_recognition_complete:
*Apr 17 16:35:33.789:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:33.789:    curr[SYNTH_IDLE]  ev-id[SPEAK]
    next[SYNTH_ASSOCIATING] action=610BA5540
*Apr 17 16:35:33.789:act_idle_speak
*Apr 17 16:35:33.793:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:33.793:    curr[SYNTH_ASSOCIATING]  ev-id[SYNTHESIZER_ASSOCIATED]
    next[SPEAKING] action=610BA7B40
*Apr 17 16:35:33.793:act_associating_speak_associated
*Apr 17 16:35:33.793:hash_add:  key=34
*Apr 17 16:35:33.949:hash_get:  key=34
*Apr 17 16:35:37.221:hash_get:  key=34
*Apr 17 16:35:37.221:hash_delete:  key=34
*Apr 17 16:35:37.221:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:37.221:    curr[SPEAKING]  ev-id[SPEECH_COMPLETE]
    next[SYNTH_IDLE] action=610BAA680
*Apr 17 16:35:37.221:act_speaking_speech_complete
*Apr 17 16:35:37.245:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:37.249:    curr[CONNECTED]  ev-id[LIB_DESTROY]
    next[CONNECTED] action=610B8DD00
*Apr 17 16:35:37.249:act_connected_libdestroy
*Apr 17 16:35:37.249:mrcp_fsm_execute:type=SYNTHESIZER
*Apr 17 16:35:37.249:    curr[CONNECTED]  ev-id[LIB_DESTROY]
    next[CONNECTED] action=610B8DD00
*Apr 17 16:35:37.249:act_connected_libdestroy
```

The following example shows output from the **debug mrcp detail**command:

```
Router# debug mrcp detail
*Sep  1 21:37:53.652: //68//MRCP:/mrcpv2_allocate_scb:
   scb=0xC07318C8, root_scb=0x661BDD54
*Sep  1 21:37:53.708: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=FALSE;TotalLength=165
*Sep  1 21:37:53.708: //-1//MRCP:/MRCPV2_ADD_HEADER:
   TotalLength=87
*Sep  1 21:37:53.708: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=TRUE;TotalLength=535:
MRCP/2.0 535      DEFINE-GRAMMAR  1
Channel-Identifier: 0000251844F8ACAD@speechrecog
:
Speech-Language: en-US
Content-Base: http://http-server1/php/
:
Content-Type: application/srgs+xml
Content-Id: field24@field.grammar
Content-Length: 290
:
<?xml version="1.0"?><grammar mode="voice" version="1.0" root="xxx"
xmlns="http://www.w3.org/2001/06/grammar" xml:lang="en-US">
       <rule id="xxx" scope="public">
         <one-of>
           <item>one</item>
           <item>two</item>
         </one-of>
       </rule>
      </grammar>
*Sep  1 21:37:53.708: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=FALSE;TotalLength=160
*Sep  1 21:37:53.708: //-1//MRCP:/MRCPV2_ADD_HEADER:
   TotalLength=82
*Sep  1 21:37:53.708: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=TRUE;TotalLength=499:
MRCP/2.0 499      RECOGNIZE  2
Channel-Identifier: 0000251844F8ACAD@speechrecog
:
Speech-Language: en-US
Confidence-Threshold: 0.50
Sensitivity-Level: 0.50
Speed-Vs-Accuracy: 0.50
Dtmf-Interdigit-Timeout: 10000
Dtmf-Term-Timeout: 0
Dtmf-Term-Char: #
No-Input-Timeout: 20000
N-Best-List-Length: 1
Logging-Tag: 68:68
Content-Base: http://http-server1/php/
Media-Type: audio/basic
Start-Input-Timers: false
:
Content-Type: text/uri-list
Content-Length: 31
:
session:field24@field.grammar
:
MRCP/2.0 80 1 200 COMPLETE
Channel-Identifier: 0000251844F8ACAD@speechrecog
:
MRCP/2.0 83 2 200 IN-PROGRESS
Channel-Identifier: 0000251844F8ACAD@speechrecog
*Sep  1 21:37:57.404: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=FALSE;TotalLength=169
*Sep  1 21:37:57.404: //-1//MRCP:/MRCPV2_ADD_HEADER_CR:
   TotalLength=93
*Sep  1 21:37:57.404: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=TRUE;TotalLength=93:
MRCP/2.0 93       START-INPUT-TIMERS  3
Channel-Identifier: 0000251844F8ACAD@speechrecog
:
```

```
MRCP/2.0 80 3 200 COMPLETE
Channel-Identifier: 0000251844F8ACAD@speechrecog
:
MRCP/2.0 148 START-OF-INPUT 2 IN-PROGRESS
Channel-Identifier: 0000251844F8ACAD@speechrecog
Proxy-Sync-Id: 0F1F813000000148
Input-Type: speech
:
MRCP/2.0 589 RECOGNITION-COMPLETE 2 COMPLETE
Channel-Identifier: 0000251844F8ACAD@speechrecog
Proxy-Sync-Id: 0F1F813000000148
Completion-Cause: 000 success
Content-Type: application/nlsml+xml
Content-Length: 369
<?xml version="1.0" encoding="UTF-8"?>
<result grammar="session:field24@field.grammar">
 <interpretation grammar="session:field24@field.grammar" confidence="0.646043">
  <instance confidence="0.646043">
   one
  </instance>
  <input mode="speech" confidence="0.646043">
   one
   <input confidence="0.646043">
    one
   </input>
  </input>
 </interpretation>
</result>
*Sep  1 21:37:59.588: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=FALSE;TotalLength=165
*Sep  1 21:37:59.588: //-1//MRCP:/MRCPV2_ADD_HEADER:
   TotalLength=87
*Sep  1 21:37:59.588: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=TRUE;TotalLength=566:
MRCP/2.0 566      DEFINE-GRAMMAR  1
Channel-Identifier: 00001FEC44F8AA93@speechrecog
:
Speech-Language: en-US
Content-Base: http://http-server1/php/
:
Content-Type: application/srgs+xml
Content-Id: field25@field.grammar
Content-Length: 321
:
<?xml version="1.0"?><grammar mode="voice" version="1.0" root="xxx"
xmlns="http://www.w3.org/2001/06/grammar" xml:lang="en-US">
       <rule id="xxx" scope="public">
         <one-of>
           <item>three</item>
           <item>four</item>
           <item>one</item>
         </one-of>
       </rule>
      </grammar>
*Sep  1 21:37:59.588: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=FALSE;TotalLength=160
*Sep  1 21:37:59.588: //-1//MRCP:/MRCPV2_ADD_HEADER:
   TotalLength=82
*Sep  1 21:37:59.588: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=TRUE;TotalLength=499:
MRCP/2.0 499      RECOGNIZE  2
Channel-Identifier: 00001FEC44F8AA93@speechrecog
:
Speech-Language: en-US
Confidence-Threshold: 0.50
Sensitivity-Level: 0.50
Speed-Vs-Accuracy: 0.50
Dtmf-Interdigit-Timeout: 10000
Dtmf-Term-Timeout: 0
Dtmf-Term-Char: #
No-Input-Timeout: 10000
N-Best-List-Length: 1
Logging-Tag: 68:68
```

```
Content-Base: http://http-server1/php/
Media-Type: audio/basic
Start-Input-Timers: false
:
Content-Type: text/uri-list
Content-Length: 31
:
session:field25@field.grammar
:
MRCP/2.0 80 1 200 COMPLETE
Channel-Identifier: 00001FEC44F8AA93@speechrecog
:
MRCP/2.0 83 2 200 IN-PROGRESS
Channel-Identifier: 00001FEC44F8AA93@speechrecog
*Sep  1 21:38:00.044: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=FALSE;TotalLength=169
*Sep  1 21:38:00.044: //-1//MRCP:/MRCPV2_ADD_HEADER_CR:
   TotalLength=93
*Sep  1 21:38:00.044: //-1//MRCP:/MRCPV2_ADD_REQUEST_LINE:
   IsFinal=TRUE;TotalLength=93:
MRCP/2.0 93       START-INPUT-TIMERS  3
Channel-Identifier: 00001FEC44F8AA93@speechrecog
:
MRCP/2.0 80 3 200 COMPLETE
Channel-Identifier: 00001FEC44F8AA93@speechrecog
:
MRCP/2.0 148 START-OF-INPUT 2 IN-PROGRESS
Channel-Identifier: 00001FEC44F8AA93@speechrecog
Proxy-Sync-Id: 0925248800000011
Input-Type: speech
:
MRCP/2.0 589 RECOGNITION-COMPLETE 2 COMPLETE
Channel-Identifier: 00001FEC44F8AA93@speechrecog
Proxy-Sync-Id: 0925248800000011
Completion-Cause: 000 success
Content-Type: application/nlsml+xml
Content-Length: 369
<?xml version="1.0" encoding="UTF-8"?>
<result grammar="session:field25@field.grammar">
 <interpretation grammar="session:field25@field.grammar" confidence="0.701971">
  <instance confidence="0.701971">
   one
  </instance>
  <input mode="speech" confidence="0.701971">
   one
   <input confidence="0.701971">
    one
   </input>
  </input>
 </interpretation>
</result>
```

The following example shows output from the **debug mrcp socket**command:

```
Router# debug mrcp socket
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_tcp_socket_connect:
   Socket=0,Dest=10.1.2.201:51001
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_connect_to_server:
   SocketConnectStatus[MRCPV2_SOCK_CONNECT_PENDING(2)], SocketId=0,
   ServerSession=0xC0732278, Dest=10.1.2.201:51001
*Sep  1 21:52:58.392: //-1//MRCP:/mrcpv2_handle_socket_read:
   Before Execute: Socket=0, SocketStatus=MRCPV2_SOCK_CONNECT_PENDING(2)
*Sep  1 21:52:58.392: //-1//MRCP:/mrcpv2_handle_socket_read:
   After Execute: Socket=0, SocketStatus=MRCPV2_SOCK_CONNECTED(1)
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_partial_socket_send:
   (Socket:0 Length:87) 600 bytes of data
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_partial_socket_send:
   Buffer Sent Successfully; fd=0, Sent=87
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_partial_socket_send:
   (Socket:0 Length:64) 600 bytes of data
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_partial_socket_send:
   Buffer Sent Successfully; fd=0, Sent=64
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_partial_socket_send:
```

```
   (Socket:0 Length:94) 600 bytes of data
*Sep  1 21:52:58.392: //74//MRCP:/mrcpv2_partial_socket_send:
   Buffer Sent Successfully; fd=0, Sent=94
```

Related Commands

Command	Description
show mrcp client session active	Displays information about active MRCP sessions.
show mrcp client session history	Displays information about past MRCP sessions.
show mrcp client statistics hostname	Displays statistics about MRCP sessions.

debug mspi receive

Note Effective with release 12.3(8)T, the **debug mspi receive**command is replaced by the **debug fax mspi**command. See the **debug fax mspi**command for more information.

To display debugging messages for the receiving mail Service Provider Interface (MSPI), use the **debug mspi receive**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mspi receive

no debug mspi receive

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(3)XI	This command was introduced on the Cisco AS5300 universal access server.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.
12.2(4)T	This command was implemented on the Cisco 1750 access router.
12.3(8)T	This command was replaced by the **debug fax mspi**command.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following is sample output from the **debug mspi receive**command:

```
Router# debug mspi receive
Jan  1 05:09:33.890: mspi_tel_num_trans: from: Radhika,
ph#in: fax=5271714 ph#dial: 5271714
Jan  1 05:09:33.890:  incoming destPat(5271714), matched(7), tag(22)
Jan  1 05:09:33.890:  out destPat(5......), tag(20), dgt strip enabled
Jan  1 05:09:33.890: mspi_off_new_rcpt: envlp_to [fax=5271714@smith.abccompany.com], 30
Jan  1 05:09:33.890:  tel_numb_dial: 5271714, subaddr:[], cover page
Jan  1 05:09:39.122: mspi_offramp_rfc822_header: msgType=0
Jan  1 05:09:39.122:  envlp_from: [Radhika], 8
Jan  1 05:09:39.122: mspi_off_put_buff: ignore mime type=1, st=CONNECTING, len=0
Jan  1 05:09:39.122: moff_save_buffer: cid=0x1F, mime=9, len=4
Jan  1 05:09:39.122:  offramp disabled receiving!
Dec 31 21:09:44.078: %ISDN-6-CONNECT: Interface Serial0:22 is now connected to 5271714
Jan  1 05:09:52.154: mspi_bridge: cid=0x1F, dst cid=0x22, data dir=OFFRAMP, conf dir=DEST
```

```
Jan  1 05:09:52.154: mspi_offramp_send_buffer: cid=0x1F, mime=9
Jan  1 05:09:52.154:  buffer with only CR/LF - set buff_len=0
Jan  1 05:09:52.154: mspi_offramp_send_buffer: cid=0x1F, mime=9 rx BUFF_END_OF_PART, offramp
 rcpt enabled
Jan  1 05:09:54.126: mspi_offramp_send_buffer: cid=0x1F, mime=11
Jan  1 05:09:54.134: mspi_offramp_send_buffer: cid=0x1F, mime=11
```

Related Commands

Command	Description
debug mspi send	Displays debugging messages for MSPI send.

debug mspi send

Note Effective with Cisco IOS Release 12.3(8)T, the **debug mspi send**command is replaced by the **debug fax mspi**command. See the **debug fax mspi**command for more information.

To display debugging messages for the sending mail Service Provider Interface (MSPI), use the **debug mspi send**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mspi send

no debug mspi send

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(3)XI	This command was introduced on the Cisco AS5300 universal access server.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.
12.2(4)T	This command was introduced on the Cisco 1750 access router.
12.3(8)T	This command was replaced by the **debug fax mspi**command.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following is sample output from the **debug mspi send** command:

```
Router# debug mspi send
*Oct 16 08:40:27.515: mspi_bridge: cid=0x21, dst cid=0x26, data dir=OFFRAMP, conf
dir=DEST
*Oct 16 08:40:29.143: mspi_setup_req: for cid=0x27
*Oct 16 08:40:29.147:   envelope_from=5??????@fax.cisco.com
*Oct 16 08:40:29.147:   envelope_to=ilyau@cisco.com
*Oct 16 08:40:30.147: mspi_chk_connect: cid=0x27, cnt=0,
*Oct 16 08:40:30.147:  SMTP connected to the server !
*Oct 16 08:40:30.147: mspi_bridge: cid=0x27, dst cid=0x28, data dir=ONRAMP, conf dir=SRC
*Oct 16 08:40:38.995: mspi_xmit: cid=0x27, st=CONFERENCED, src_cid=0x28, buf cnt=0
```

Related Commands

Command	Description
debug mspi receive	Displays debugging messages for MSPI receive.

debug mta receive all

Note Effective with release 12.3(8)T, the **debug mta receiv all**command is replaced by the **debug fax mta**command. See the **debug fax mta**command for more information.

To show output relating to the activity on the Simple Mail Transfer Protocol (SMTP) server, use the **debug mta receive all** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mta receive all

no debug mta receive all

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(4)T	This command was introduced.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.
12.2(4)T	This command was implemented on the Cisco 1750 access router.
12.2(8)T	This command was implemented on the Cisco 1751 access routers, Cisco 3725 access routers, and Cisco 3745 access routers.
12.2(13)T	This feature was implemented on the Cisco 7200 series routers.
12.3(8)T	This command was replaced by the **debug fax mta**command.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following example shows the messages exchanged (for example, the handshake) between the e-mail server and the off-ramp gateway:

```
Router# debug mta receive all
Jan  1 05:07:41.314: esmtp_server_work: calling helo
Jan  1 05:07:43.354: esmtp_server_work: calling mail
Jan  1 05:07:45.386: esmtp_server_work: calling rcpt
```

```
Jan  1 05:07:47.426: esmtp_server_work: calling data
Jan  1 05:07:49.514: (S)R: 'Content-Type: multipart/mixed;
boundary="------------11F7CD9D2EB3E8B8D5627C62"'
Jan  1 05:07:49.514: (S)R: ''
Jan  1 05:07:49.514: esmtp_server_engine_new_part:
Jan  1 05:07:49.514: (S)R: 'Content-Type: text/plain; charset=us-ascii'
Jan  1 05:07:49.514: (S)R: 'Content-Transfer-Encoding: 7bit'
Jan  1 05:07:49.514: (S)R: ''
Jan  1 05:07:49.514: esmtp_server_engine_new_part:
Jan  1 05:07:49.514: esmtp_server_work: freeing temp header
Jan  1 05:07:49.514: (S)R: 'Content-Type: image/tiff; name="DevTest.8.1610.tif"'
Jan  1 05:07:49.514: (S)R: 'Content-Transfer-Encoding: base64'
Jan  1 05:07:49.514: (S)R: 'Content-Disposition: inline; filename="DevTest.8.1610.tif"'
Jan  1 05:07:49.514: (S)R: ''
Jan  1 05:07:49.514: esmtp_server_engine_update_recipient_status: status=6
Jan  1 05:07:49.514: esmtp_server_engine_new_part:
Jan  1 05:07:49.518: esmtp_server_work: freeing temp header
Jan  1 05:08:03.014: esmtp_server_engine_update_recipient_status: status=7
Jan  1 05:08:04.822: esmtp_server_engine_update_recipient_status: status=6
Jan  1 05:08:33.042: esmtp_server_engine_update_recipient_status: status=7
Jan  1 05:08:34.906: esmtp_server_engine_getline: Unexpected end of file on socket 1
Jan  1 05:08:34.906: esmtp_server_work: error occurred with ctx=0x61FFF710, socket=1
```

Related Commands

Command	Description
debug mta send all	Displays output for all the on-ramp client connections.

debug mta send all

Note Effective with release 12.3(8)T, the **debug mta send all**command is replaced by the **debug fax mta**command. See the **debug fax mta**command for more information.

To display output for all of the on-ramp client connections, use the **debug mta send all** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mta send all

no debug mta send all

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(4)T	This command was introduced.
12.2(4)T	This command was implemented on the Cisco 1750 access router.
12.2(8)T	This command was implemented on the Cisco 1751 access routers, Cisco 3725 access routers, and Cisco 3745 access routers.
12.2(13)T	This feature was implemented on the Cisco 7200 series routers.
12.3(8)T	This command was replaced by the **debug fax mta** command.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following example shows the messages exchanged (for example, the handshake) between the e-mail server and the on-ramp gateway:

```
Router# debug mta send all
*Oct 16 09:04:13.055: esmtp_client_engine_open: from=5551212@fax.cisco.com,
to=madeup@abccompany.com
*Oct 16 09:04:13.055: esmtp_client_engine_add_headers: from_comment=
*Oct 16 09:04:13.111: esmtp_client_work: socket 0 attempting to connect to IP address
171.71.154.56
*Oct 16 09:04:13.111: esmtp_client_work: socket 0 readable for first time
*Oct 16 09:04:13.135: esmtp_client_work: socket 0 readable for first time
*Oct 16 09:04:13.135: (C)R: 220 madeup.abccompany.com ESMTP Sendmail 8.8.4-Cisco.1/8.6.5
```

```
ready at Wed, 27 Sep 2000 11:45:46 -0700 (PDT)
*Oct 16 09:04:13.135: (C)S: EHLO mmoip-c.cisco.com
*Oct 16 09:04:13.183: (C)R: 250-madeup.abccompany.com Hello [172.22.95.16], pleased to meet
 you
*Oct 16 09:04:13.183: (C)R: 250-EXPN
*Oct 16 09:04:13.183: (C)R: 250-VERB
```

Related Commands

Command	Description
debug mta send rcpt-to	Displays output for a specific on-ramp SMTP client connection during an e-mail transmission.

debug mta send rcpt-to

Note Effective with release 12.3(8)T, the **debug mta send rcpt-to**command is no longer available in Cisco IOS.

To display output for a specific on-ramp Simple Mail Transfer Protocol (SMTP) client connection during an e-mail transmission, use the **debug mta send rcpt-to** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mta send rcpt-to *string*

no debug mta send rcpt-to *string*

Syntax Description

string	E-mail address.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(4)T	This command was introduced.
12.2(4)T	This command was implemented on the Cisco 1750 access router.
12.2(8)T	This command was implemented on the Cisco 1751 access routers, Cisco 3725 access routers, and Cisco 3745 access routers.
12.2(13)T	This feature was implemented on the Cisco 7200 series routers.
12.3(8)T	This command was removed and is no longer available in Cisco IOS software.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following example shows debugging information displayed when the **debug mta send rcpt-to** command has been enabled and the SMTP client is sending an e-mail message:

```
Router# debug mta send rcpt-to 5551212

Router# socket 0 attempting to connect to IP address 100.00.00.00
```

```
socket 0 readable for first time - let's try to read it
R:220 madeup.abc.com ESMTP Sendmail 8.8.4-abc.1/8.6.5 ready at Tue, 6
Apr 1999 13:35:39 -0700 (PDT)
S:EHLO mmoip-c.abc.com
R:250-quisp.cisco.com Hello [100.00.00.00], pleased to meet you
R:250-EXPN
R:250-VERB
R:250-8BITMIME
R:250-SIZE
R:250-DSN
R:250-ETRN
R:250-XUSR
R:250 HELP
S:MAIL FROM:<testing@> RET=HDRS
R:250 <testing@>... Sender ok
S:RCPT TO:<madeup@abc.com> NOTIFY=SUCCESS ORCPT=rfc822;testing@
R:250 <madeup@abc.com>... Recipient ok
R:354 Enter mail, end with "." on a line by itself
S:Received:(Cisco Powered Fax System) by mmoip-c.cisco.com for
<madeup@abc.com> (with Cisco NetWorks); Fri, 17 Oct 1997 14:54:27 +0800
S:To: <madeup@abc.com>
S:Message-ID:<000F1997145427146@mmoip-c.cisco.com>
S:Date:Fri, 17 Oct 1997 14:54:27 +0800
S:Subject:mmoip-c subject here
S:X-Mailer:IOS (tm) 5300 Software (C5300-IS-M)
S:MIME-Version:1.0
S:Content-Type:multipart/mixed;
S: boundary="yradnuoB=_000E1997145426826.mmoip-ccisco.com"
S:From:"Test User" <testing@>
S:--yradnuoB=_000E1997145426826.mmoip-ccisco.com
S:Content-ID:<00101997145427150@mmoip-c.cisco.com>
S:--yradnuoB=_000E1997145426826.mmoip-ccisco.com--
Sending terminating dot ...(socket=0)
S:.
R:250 NAA09092 Message accepted for delivery
S:QUIT
R:221 madeup@abc.com closing connection
Freeing SMTP ctx at 0x6121D454
returned from work_routine, context freed
```

Related Commands

Command	Description
debug mta send all	Displays output for all the on-ramp client connections.

debug mvrp

To display debugging information for Multiple VLAN Registration Protocol (MVRP) configurations, use the **debug mvrp** command in privileged EXEC mode. To disable debugging of MVRP configurations, use the **no** form of this command.

debug mvrp [**all**| **config**| **error**| **event**| **ha**| **packets**| **switch**]

no debug mvrp

Syntax Description

all	(Optional) Enables all levels of debugging
config	(Optional) Displays user configuration information.
error	(Optional) Enables error-level debugging.
event	(Optional) Enables event-level debugging.
ha	(Optional) Enables high availability-level debugging.
packets	(Optional) Enables packet-level debugging.
switch	(Optional) Enables switch-level debugging.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(33)SXI	This command was introduced.

Usage Guidelines

Conditional interface debugging can be used to limit the scope of output messages related to an interface.

Cisco Catalyst 6000 Series Platforms

On switches with a Switch Processor (SP) or Route Processor (RP), this command can be used only on the SP console.

Examples

The following example shows switch-level debugging enabled:

```
Router# debug mvrp switch
```

Related Commands

Command	Description
clear mvrp statistics	Clears statistics related to MVRP and recorded on one (or all) MVRP-enabled ports.
show mvrp	Displays statistics for configured MVRP attributes on a device or specified ports on a device.

debug mwi relay errors

To debug message waiting indication (MWI) relay errors, use the **debug mwi relay errors**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mwi relay errors

no debug mwi relay errors

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XT	This command was introduced on the following platforms: Cisco 1750, Cisco 1751, Cisco 2600 series and Cisco 3600 series multiservice routers; and Cisco IAD2420 series Integrated Access Devices (IADs).
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T and implemented on the Cisco 3725 and Cisco 3745 routers.
12.2(8)T1	This command was implemented on the Cisco 2600-XM and Cisco 2691 routers.
12.2(11)T	This command was implemented on the Cisco 1760 routers.

Usage Guidelines

The **debug mwi relay errors** command provides a debug monitor display of any error messages, when MWI Relay Server (Cisco IOS Telephony Server) is trying to do MWI Relay to extensions on remote Cisco IOS Telephony Service (ITS).

Examples

The following examples show errors when MWI Relay Server tries to do an MWI Relay to extension 7004, but location of 7004 is not known to the MWI Relay Server:

```
Router#
debug mwi relay errors

mwi-relay error info debugging is on
01:46:48: MWI-APP: mwi_notify_status: No ClientID (7004) registered
```

Related Commands

Command	Description
debug ephone mwi	Sets MWI debugging for the Cisco IOS Telephony Service router.
debug mwi relay events	Sets MWI relay events debugging for the Cisco IOS Telephony Service router.

debug mwi relay events

To set message waiting indication (MWI) relay events debugging, use the **debug mwi relay events**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mwi relay events

no debug mwi relay events

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XT	This command was introduced on the following platforms: Cisco 1750, Cisco 1751, Cisco 2600 series and Cisco 3600 series multiservice routers; and Cisco IAD2420 series Integrated Access Devices (IADs).
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T and implemented on the Cisco 3725 and Cisco 3745 routers.
12.2(8)T1	This command was implemented on the Cisco 2600-XM and Cisco 2691 routers.
12.2(11)T	This command was implemented on the Cisco 1760 routers.

Usage Guidelines

The **debug mwi relay events** command provides a debug monitor display of events, when MWI Relay Server (Cisco IOS Telephony Server) is trying to do MWI Relay to extensions on remote Cisco IOS Telephony Services (ITS).

Examples

The following debugging messages are shown when the MWI Relay server tries to send MWI Information to remote client 7001 and the location of 7001 is known by the MWI Relay Server:

```
Router# debug mwi relay events

mwi-relay events info debugging is on
01:45:34: mwi_notify_status: Queued event for mwi_app_queue
01:45:34: MWI-APP: mwi_app_process_event:
01:45:34: MWI-APP: mwi_app_process_event: MWI Event for ClientID(7001)@(1.8.17.22)
```

Related Commands

Command	Description
debug ephone mwi	Sets MWI debugging for the Cisco IOS Telephony Service router.
debug mwi relay errors	Sets MWI relay errors debugging for the Cisco IOS Telephony Service router.