

debug mpls ldp transport connections

To display information about the TCP connections used to support LDP sessions, use the **debug mpls ldp transport connections** command in privileged EXEC mode. To disable debugging output, 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 <i>acl</i>	(Optional) Limits the displayed information to that for LDP peers permitted by the access control list (ACL).
interface <i>interface</i>	(Optional) Limits the displayed information to that for the specified interface.

Defaults

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

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1 CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
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.

Usage Guidelines

Use this command to monitor LDP activity relating to the establishment of the 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 **debug mpls ldp transport connections** command:

```
Router# debug mpls ldp transport connections
```

```
Debug output at active peer:
```

```
ldp: Opening listen port 646 for 10.0.0.44, 34.0.0.44
ldp: Open LDP listen TCB 0x60E105BC; lport = 646; fhost = 10.0.0.44
ldp: Add listen TCB to list; tcb 0x60E105BC; addr 10.0.0.44
ldp: Incoming ldp conn 133.0.0.33:646 <-> 10.0.0.44:11042
ldp: create ptcl_adj: tp = 0x610ECD64, ipaddr = 10.0.0.44
```

Debug output at passive peer:

```
ldp: Opening ldp conn; adj 0x60BAC33C, 10.0.0.44 <-> 133.0.0.33
ldp: ldp conn is up; adj 0x60BAC33C, 10.0.0.44:11042 <-> 133.0.0.33:646
```

[Table 171](#) describes the significant fields shown in the sample display.

Table 171 *debug mpls ldp transport connections Field Descriptions*

Field	Description
ldp:	Identifies the source of the message as LDP.
adj 0xnntnnnnnn	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 LDP peer discovery mechanism, use the **debug mpls ldp transport events** command in privileged EXEC mode. To disable debugging output, 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 <i>acl</i>	(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.

Defaults

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

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1 CT	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF command syntax and terminology.
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.

Usage Guidelines

Use the command to monitor LDP discovery activity. This mechanism is used to determine the devices with which you wish to establish LDP sessions.

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 **debug mpls ldp transport events** 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 8.1.1.1/224.0.0.2, 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

```

Table 172 describes the significant fields in the sample display.

Table 172 *debug mpls ldp transport events Field Descriptions*

Field	Description
ldp:	Identifies the source of the message as LDP.
adj 0xn	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: <ul style="list-style-type: none"> • 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 CEF routes are added, changed, or removed, use the **debug mpls lfib cef** EXEC command. 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.

Defaults This command has 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 command syntax.

Usage Guidelines Several lines of output are produced for each route placed into the 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 displayed when you enter the **debug mpls lfib cef** 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

```

Table 173 lists the significant fields shown in the display.

See Table 175 for a description of special labels that appear in the output of this debug command.

Table 173 *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: <ul style="list-style-type: none"> 0x1 = dynamic 0x2 = TSP tunnels 0x3 = both
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.

Table 173 debug mpls lfib cef Field Descriptions (continued)

Field	Description
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:l	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.
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 <i>x</i> is 0, indicates that no change has occurred in the LFIB entry. If <i>x</i> 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, <i>s</i> = “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.

Table 173 *debug mpls lfib cef* Field Descriptions (continued)

Field	Description
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 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 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 **debug mpls lfib enc** 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.

Defaults This command has 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 command syntax.

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 an example of output generated when you issue the **debug mpls lfib enc** 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
```

Table 174 describes the significant fields shown in the display.

Table 174 *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 <i>x</i> is 1, this is a point-to-point adjacency. If <i>x</i> is 0, it is not.
fibidb = s	Indicates the interface of the adjacency.
linktype = x	The link type of the adjacency, as follows: <ul style="list-style-type: none"> • 7 = LINK_IP • 79 = LINK_TAG
sizes x,y,z	Indicates the following values: <ul style="list-style-type: none"> • x = length of macstring • y = length of tag encapsulation • z = tag MTU
type = x	Tag encapsulation type, as follows: <ul style="list-style-type: none"> • 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.

Table 175 describes the special labels, which sometimes appear in the debug output, and their meanings.

Table 175 *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.

Table 175 Special Labels Appearing in debug Command Output (continued)

Special Label	Meaning
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.
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 lsp

To print detailed information about label rewrites being created and deleted as LSP tunnels are added or removed, use the **debug mpls lfib lsp** EXEC command. 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.

Defaults This command has 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 command syntax.

Examples The following is sample output generated from the **debug mpls lfib lsp** 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

```

Table 176 describes the significant fields in the sample display shown above.

Table 176 *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.
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 **debug mpls lfib state EXEC** command. 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.

Defaults This command has 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 command syntax.

Usage Guidelines Use this command when you wish to trace what happens to the LFIB when you issue the **mpls ip** or the **mpls tsp-tunnel** command.

Examples The following is sample output generated from the **debug mpls lfib state** 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
```

Table 177 describes the significant fields shown in the display.

Table 177 *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 <i>x</i> 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: <ul style="list-style-type: none"> • status byte • maximum transmission unit (<i>tmtu</i>) • range of ATM VPs • control VP • control VC (if this is a TC-ATM interface) • encapsulation type (<i>et</i>) • encapsulation information • tunnel interface number (<i>lc</i>) • 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.
cleanup: tfib[x] still non-0	Indicates that the LFIB is being deleted, but that slot <i>x</i> is still active.
disable lc msg rcvd, 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 rcvd,type=0x1	Indicates that the line card received a message to enable label switching type 1 (dynamic).

Table 177 *debug mpls lfib state Field Descriptions (continued)*

Field	Description
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: <ul style="list-style-type: none"> • 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 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 struct

To trace the allocation and freeing of LFIB-related data structures, such as the LFIB itself, label rewrites, and label_info data, use the **debug mpls lfib struct** EXEC command. 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.

Defaults This command has 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 command syntax.

Examples The following is sample output generated from the **debug mpls lfib struct** 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
```

Table 178 describes the significant fields shown in the display.

Table 178 *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 <i>x</i> is the lowest numbered slot still in use.
TFIB freed	The LFIB table has been freed.
enable, TFIB allocated, size <i>x</i> bytes, maxtag = <i>y</i>	Label switching has been enabled and an LFIB of <i>x</i> bytes has been allocated. The largest legal label is <i>y</i> .
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.
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 packets

To display labeled packets switched by the host router, use the **debug mpls packets EXEC** command. To disable debugging output, use the **no** form of this command.

debug mpls packets [*interface*]

no debug mpls packets [*interface*]

Syntax Description	<i>interface</i>	(Optional.) The interface or subinterface name.
--------------------	------------------	---

Defaults	Displays all labeled packets regardless of interface.
----------	---

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 command syntax.	

Usage Guidelines	The optional <i>interface</i> parameter restricts the display to only those packets received or sent on the indicated interface.
------------------	--



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 debug mpls packets 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
```

Table 179 describes the significant fields shown in the display.

Table 179 *debug mpls packets Field Descriptions*

Field	Description
Hs0/0	The identifier for the interface on which the packet was received or sent.
rcvd	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 popped off the packet and were sent unlabeled.
Tag(s)	A list of labels on the packet, ordered from the top of the stack to the bottom.

Related Commands

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

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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Release	Modification
12.0(5)ST	This command was introduced.

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

```
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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.0(5)ST	This command was introduced.

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

```
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 link-management admission-control

To print information about traffic engineering 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] [aclnum]
```

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

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(Optional) Uses the specified access list to filter the debugging information. Prints information only for those LSPs that match the access list.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword and the <i>aclnum</i> argument were added.

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

```
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 **debug mpls traffic-eng link-management advertisements** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls traffic-eng link-management advertisements [detail] [aclnum]

no debug mpls traffic-eng link-management advertisements [detail] [aclnum]

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(Optional) Uses the specified access list to filter the debugging information.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword was added.

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

```
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

```

Table 180 describes the significant fields shown in the display.

Table 180 *debug isis mpls traffic-eng link-management advertisements Field Descriptions*

Field	Description
Flooding 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 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**] [*aclnum*]

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

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(Optional) Uses the specified access list to filter the debugging information. Prints information only for those LSPs that match the access list.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword and the <i>aclnum</i> argument were added.

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

```
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 succeededx_„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.														
Defaults	No default behavior or values.														
Command Modes	Privileged EXEC														
Command History	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>12.1(3)T</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	12.1(3)T	This command was introduced.										
Release	Modification														
12.1(3)T	This command was introduced.														
Examples	<p>In the following example, detailed debugging information is printed about errors encountered during a traffic engineering link management procedure:</p> <pre>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</pre>														
Related Commands	<table border="1"> <thead> <tr> <th>Command</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>debug mpls traffic-eng link-management admission-control</td> <td>Prints information about traffic engineering LSP admission control on traffic engineering interfaces.</td> </tr> <tr> <td>debug mpls traffic-eng link-management advertisements</td> <td>Prints information about resource advertisements for traffic engineering interfaces.</td> </tr> <tr> <td>debug mpls traffic-eng link-management bandwidth-allocation</td> <td>Prints information about bandwidth allocation for traffic engineering LSPs.</td> </tr> <tr> <td>debug mpls traffic-eng link-management events</td> <td>Prints information about traffic engineering link management system events.</td> </tr> <tr> <td>debug mpls traffic-eng link-management igp-neighbors</td> <td>Prints information about changes to the link management databases of IGP neighbors.</td> </tr> <tr> <td>debug mpls traffic-eng link-management links</td> <td>Prints information about traffic engineering link management interface events.</td> </tr> </tbody> </table>	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.	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.
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.														
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.
---------------------------	---

Defaults	No default behavior or values.
-----------------	--------------------------------

Command Modes	Privileged EXEC
----------------------	-----------------

Command History	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>12.05(S)</td> <td>This command was introduced.</td> </tr> <tr> <td>12.1(3)T</td> <td>The detail keyword was added.</td> </tr> </tbody> </table>	Release	Modification	12.05(S)	This command was introduced.	12.1(3)T	The detail keyword was added.
Release	Modification						
12.05(S)	This command was introduced.						
12.1(3)T	The detail keyword was added.						

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

```
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 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.
--------------------	--------	---

Defaults	No default behavior or values.
----------	--------------------------------

Command Modes	Privileged EXEC
---------------	-----------------

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword was added.

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

```
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.
---------------------------	---

Defaults	No default behavior or values.
-----------------	--------------------------------

Command Modes	Privileged EXEC
----------------------	-----------------

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword was added.

Examples	<p>In the following example, detailed debugging information is printed about traffic engineering link management interface events:</p>
-----------------	--

```
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 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.				
Defaults	No default behavior or values.				
Command Modes	Privileged EXEC				
Command History	<table border="1"> <thead> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">Release</th> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">Modification</th> </tr> </thead> <tbody> <tr> <td style="border-bottom: 1px solid black;">12.1(3)T</td> <td style="border-bottom: 1px solid black;">This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	12.1(3)T	This command was introduced.
Release	Modification				
12.1(3)T	This command was introduced.				

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

```
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 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.
---------------------------	---

Defaults	No default behavior or values.
-----------------	--------------------------------

Command Modes	Privileged EXEC
----------------------	-----------------

Command History	<table border="1"> <thead> <tr> <th style="border: none;">Release</th> <th style="border: none;">Modification</th> </tr> </thead> <tbody> <tr> <td style="border: none;">12.05(S)</td> <td style="border: none;">This command was introduced.</td> </tr> <tr> <td style="border: none;">12.1(3)T</td> <td style="border: none;">The detail keyword was added.</td> </tr> </tbody> </table>	Release	Modification	12.05(S)	This command was introduced.	12.1(3)T	The detail keyword was added.
Release	Modification						
12.05(S)	This command was introduced.						
12.1(3)T	The detail keyword was added.						

Examples	<p>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:</p>
-----------------	---

```
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	<table border="1"> <thead> <tr> <th style="border: none;">Command</th> <th style="border: none;">Description</th> </tr> </thead> <tbody> <tr> <td style="border: none;">debug ip rsvp</td> <td style="border: none;">Prints information about RSVP signalling events.</td> </tr> </tbody> </table>	Command	Description	debug ip rsvp	Prints information about RSVP signalling events.
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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Release	Modification
12.0(5)ST	This command was introduced.

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

```
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 path

To print information about 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 { *num* | **lookup** | **spf** | **verify** }

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

Syntax Description	<i>num</i>	Prints path calculation information only for the local tunneling interface with unit number <i>num</i> .
	lookup	Prints information for path lookups.
	spf	Prints information for shortest path first (SPF) calculations.
	verify	Prints information for path verifications.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.0(5)ST	This command was introduced.

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

```
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 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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Release	Modification
12.0(5)ST	This command was introduced.

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

```
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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.0(5)ST	This command was introduced.

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

```
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.
--------------------	--------	---

Defaults	No default behavior or values.
----------	--------------------------------

Command Modes	Privileged EXEC
---------------	-----------------

Command History	Release	Modification
	12.1(3)T	This command was introduced.

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

```
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.
---------------------------	---

Defaults	No default behavior or values.
-----------------	--------------------------------

Command Modes	Privileged EXEC
----------------------	-----------------

Command History	Release	Modification
	12.05(S)	This command was introduced.
12.1(3)T	The detail keyword was added.	

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

```
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 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] [aclnum]
```

```
no debug mpls traffic-eng tunnels labels [detail] [aclnum]
```

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(Optional) Uses the specified access list to filter the debugging information. Prints information only about traffic engineering tunnels that match the access list.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword and the <i>aclnum</i> argument were added.

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

```
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 **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 re-optimizations, 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] [aclnum]
```

```
no debug mpls traffic-eng tunnels reoptimize [detail] [aclnum]
```

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword and the <i>aclnum</i> argument were added.

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

```
debug mpls traffic-eng tunnels reoptimize detail 101
```

```
LSP-TUNNEL-REOPT:Tunnell curr option 2 (0x6175CF8C), activate new option 2
LSP-TUNNEL-REOPT:Tunnell new path:option 2 [10002], weight 20
LSP-TUNNEL-REOPT:Tunnell old path:option 2 [2], weight 110
LSP-TUNNEL-REOPT:Tunnell [10002] set as reopt
LSP-TUNNEL-REOPT:Tunnell path option 2 [10002] installing as current
LSP-TUNNEL-REOPT:Tunnell [2] removed as current
LSP-TUNNEL-REOPT:Tunnell [2] set to delayed clean
LSP-TUNNEL-REOPT:Tunnell [10002] removed as reopt
LSP-TUNNEL-REOPT:Tunnell [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] [aclnum]

no debug mpls traffic-eng tunnels signalling [detail] [aclnum]

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword and the <i>aclnum</i> argument were added.

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

```
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] [aclnum]
```

```
no debug mpls traffic-eng tunnels state [detail] [aclnum]
```

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(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.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.1(3)T	This command was introduced.

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

```
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] [aclnum]

no debug mpls traffic-eng tunnels timers [detail] [aclnum]

Syntax Description	detail	(Optional) Prints detailed debugging information.
	<i>aclnum</i>	(Optional) Uses the specified access list to filter the debugging information. Prints information about traffic engineering tunnel timer management that matches the access list.

Defaults No default behavior or values.

Command Modes Privileged EXEC

Command History	Release	Modification
	12.05(S)	This command was introduced.
	12.1(3)T	The detail keyword and the <i>aclnum</i> argument were added.

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

```
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 xmplsatm cross-connect

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

debug mpls xmplsatm cross-connect

no debug mpls xmplsatm cross-connect

Syntax Description This command has no arguments or keywords.

Defaults 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.

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 **debug mpls xmplsatm cross-connect** command:

```
Router# debug mpls xmplsatm 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 0x60CDBC5C, userbits 0x2, result
OK
          0xC0200 1/37 --> 0xC0300 1/37
```

Table 181 describes the significant fields shown in the display.

Table 181 *debug mpls xmplsatm cross-connect Command Field Descriptions*

Field	Description
XTagATM	The source of the debug message as an XmplsATM interface.
cross-conn	An indicator that the debug 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.
0xC0100 (Ctl-If) 1/32	Information about the interface: <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • 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 xmplsatm cross-connect	Displays information about remotely connected ATM switches.

debug mpls xmplsatm errors

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

debug mpls xmplsatm errors

no debug mpls xmplsatm errors

Syntax Description This command has no arguments or keywords.

Defaults No default behavior or values.

Command Modes Privileged EXEC

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.

Usage Guidelines Use the **debug mpls xmplsatm 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 xmplsatm errors** command:

```
Router# debug mpls xmplsatm 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 xmplsatm events

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 **debug mpls xmplsatm events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls xmplsatm events

no debug mpls xmplsatm events

Syntax Description This command has no arguments or keywords.

Defaults 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.

Usage Guidelines Use the **debug mpls xmplsatm events** 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 **debug mpls xmplsatm events** command is in effect include the following:

- Receiving asynchronous notifications that the VSI master sent through the external ATM application programming interface (API) (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 **debug mpls xmplsatm events** command:

```
Router# debug mpls xmplsatm 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
```

Table 182 describes the significant fields shown in the display.

Table 182 *debug mpls xmplsatm events Command Field Descriptions*

Field	Description
XTagATM	The source of the debug 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 xmplsatm vc

To display information about events that affect individual XmplsATM terminating virtual circuits (VCs), use the **debug mpls xmplsatm vc** command. To disable debugging output, use the **no** form of this command.

debug mpls xmplsatm vc

no debug mpls xmplsatm vc

Syntax Description This command has no arguments or keywords.

Defaults 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.

Usage Guidelines Use the **debug mpls xmplsatm vc** 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 **debug mpls xmplsatm vc** command only when a few XmplsATM VCs are created or removed.

Examples The following is sample output from the **debug mpls xmplsatm vc** command:

```
Router# debug mpls xmplsatm 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
```

Table 183 describes the significant fields shown in the display.

Table 183 *debug mpls xmplsatm vc Command Field Descriptions*

Field	Description
XTagATM VC	The source of the debug 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.