debug management event through debug mpls ldp bindings

debug management event mib

To monitor the activities of the Event MIB in real time on your routing device, use the **debug management event mib**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug management event mib
no debug management event mib

Syntax Description

This command has no arguments or keywords.

Command Default

Debugging output is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.1(3)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.

Usage Guidelines

The **debug management event mib** command prints messages to the screen whenever the Event MIB evaluates a specified trigger. These messages are given in real-time, and are intended to be used by technical support engineers for troubleshooting purposes. Definitions for the OID (object identifier) fields can be found in the EVENT-MIB.my file, available for download from the Cisco MIB website on http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.

Examples

The following is sample output from the **debug management event mib**command:

```
Router# debug management event mib
Event Process Bool: Owner aseem, Trigger 01
 Event Bool process: invoke event
 Event Bool process: no wildcarding
Event: OID ifEntry.10.3
Event getValue abs: 69847284
 Event Bool process: Trigger Fired !
 mteSetNotifyObjects:
 Event execOnFiring: sending notification
Event: OID ifEntry.10.1
Event add_objects: Owner , Trigger
Event add_objects: Owner aseem, Trigger sethi
Event Found Owner: aseem
Event Found Name: sethi
Event: OID ifEntry.10.1
 Event: sending trap with 7 OIDs
Event: OID mteHotTrigger.0
Event: OID mteHotTargetName.0
Event: OID mteHotContextName.0
Event: OID ifEntry.10.3
Event: OID mteHotValue.0
Event: OID ifEntry.10.1
Event: OID ifEntry.10.1
```

```
Event mteDoSets: setting oid
 Event mteDoSets: non-wildcarded oid
Event: OID ciscoSyslogMIB.1.2.1.0
Event Thresh Process: Owner aseem, Trigger 01
 Event Thresh process: invoke rising event
 Event Thresh process: invoke falling event
 Event Thresh process: no wildcarding
Event: OID ifEntry.10.3
Event getValue abs: 69847284
Event Existence Process: Owner aseem, Trigger 01
 Event Exist process: invoke event
 Event Exist process: no wildcarding
Event: OID ifEntry.10.3
Event getValue abs: 69847284
 Event Check ExistTrigger for Absent
 Event Check ExistTrigger for Changed
Router# no debug management event mib
```

Related Commands

Command	Description
show management event	Displays the SNMP Event values that have been configured on your routing device through the use of the Event MIB.

debug management expression

To monitor the activities of the Expression MIB in real time on your routing device, use the **debug management expression**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug management expression {**evaluator** | **mib** | **parser**}
debug management expression {**evaluator** | **mib** | **parser**}

Syntax Description

evaluator	Specifies the Expression MIB evaluator.
mib	Specifies the Expression MIB SNMP operations.
parser	Specifies the Expression MIB parsing.

Command Default

By default, debugging is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(1)	This command was introduced in a release earlier than Cisco IOS Release 12.2(1).
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.
12.2SR	This command is supported in the Cisco IOS Release 12.2SR train. Support in a specific 12.2SR release of this train depends on your feature set, platform, and platform hardware.
12.2SB	This command is supported in the Cisco IOS Release 12.2SB train. Support in a specific 12.2SB Release of this train depends on your feature set, platform, and platform hardware.

Examples

The following example shows how to enable debugging options for Expression MIB:

```
Router# debug management expression mib
Expression MIB SNMP operations debugging is on
```

Related Commands

Command	Description
show management expression	Displays the SNMP Expression values that have been configured on your routing device through the use of the Expression MIB.

debug mdns

To enable the debugging of multicast Domain Name System (mDNS) service discovery information, use the **debug mdns** command in privileged EXEC mode. To disable the debugging output, use the **no** form of this command.

debug mdns {**all** | **error** | **event** | **packet** | **verbose**}
no debug mdns

Syntax Description

all	Enables logging of the information about the mDNS service discovery processes.
error	Enables logging of the information about the errors encountered by the mDNS responder.
event	Enables logging of the information about the various events such as free, memory-allocated, packet, request, and timer.
packet	Enables logging of the information about the hex dump (byte by byte printing of packet traffic information) moving in and out of the mDNS responder.
verbose	Enables logging of detailed mDNS service discovery information.

Command Default

Debugging of mDNS service discovery is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.3(2)S	This command was introduced.

Examples

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

```
Device> enable
Device# debug mdns event
Device# mDNS event debugs debugging is on
Device# sh log
Syslog logging: enabled (0 messages dropped, 14 messages rate-limited, 0 flushes, 0 overruns,
 xml disabled, filtering disabled)

No Active Message Discriminator.



No Inactive Message Discriminator.


    Console logging: disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 4762561 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level informational, 176 message lines logged
        Logging Source-Interface:       VRF Name:
Log Buffer (4096 bytes):
er cache hit!
*Mar 15 03:01:38.234: SISF[CLA]: Interested feature:
*Mar 15 03:01:38.234: SISF[CLA]:                Snooping
*Mar 15 03:01:38.234: SISF[SWI]: Gi0/0/1 vlan 0 Feature_0  Snooping priority 128
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0 Parse msg ND_ROUTER_ADVERT. len 48
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0 Found 3 options
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0            option 1 : ND_OPT_SOURCE_LINKADDR
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0            option 3 :
ND_OPT_PREFIX_INFORMATION
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0            option 5 : ND_OPT_MTU
*Mar 15 03:01:38.234: SISF[PRS]:
*Mar 15 03:01:38.234: SISF[GLN]: Gi0/0/1 vlan 0 IPv6 snooping Gleaner setting sec level to
 2
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0 Sec level is Guard
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0 Advertise from access: default action is
update entry
*Mar 15 03:01:38.234: SISF[PRS]: Gi0/0/1 vlan 0 Unallowed RA/Redir: default action is delete
 entry
*Mar 15 03:01:38.234: SISF[GLN]: Gi0/0/1 vlan 0 Unauthorized packet
*Mar 15 03:01:38.234: SISF[SWI]: Gi0/0/1 vlan 0 Feature Snooping rc 1
*Mar 15 03:01:38.235: SISF[SWI]: Gi0/0/1 vlan 0 Feature drop
*Mar 15 03:01:38.235: SISF[MEM]: Unlocking, count is now 0
*Mar 15 03:01:38.235: SISF[MEM]:  3BB56338 semaphore system unlocked
*Mar 15 03:01:38.485: SISF[SWI]: SISF IPv6 enqueue FE80::217:95FF:FE73:9600
*Mar 15 03:01:40.716: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:01:40.866: SISF[SWI]: SISF IPv6 enqueue FE80::213:80FF:FE3E:8B25
*Mar 15 03:01:41.466: SISF[SWI]: SISF IPv6 enqueue FE80::213:80FF:FE3E:8B24
*Mar 15 03:01:41.644: SISF[SWI]: SISF IPv6 enqueue FE80::221:D8FF:FECD:5F40
*Mar 15 03:01:45.376: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:01:49.732: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:01:50.463: SISF[SWI]: Match ACL for incoming packet on Gi0/0/1
*Mar 15 03:01:50.463: SISF[SWI]: SISF IPv6 highjack L3-IF Gi0/0/1
*Mar 15 03:01:50.463: SISF[MEM]: Owner is this process
*Mar 15 03:01:50.463: SISF[MEM]: semaphore 3BB56338 (re)locked
*Mar 15 03:01:50.463: SISF[MEM]: Locking, count is now 1
*Mar 15 03:01:50.463: SISF[CLA]: Building interested feature list
*Mar 15 03:01:50.463: SISF[CLA]: Interest on target Gi0/0/1
*Mar 15 03:01:50.463: SISF[CLA]: Classifier cache hit!
*Mar 15 03:01:50.463: SISF[CLA]: Interested feature:
*Mar 15 03:01:50.463: SISF[CLA]:                Snooping
*Mar 15 03:01:50.463: SISF[SWI]: Gi0/0/1 vlan 0 Feature_0  Snooping priority 128
*Mar 15 03:01:50.463: SISF[PRS]: Gi0/0/1 vlan 0 Parse msg ND_ROUTER_ADVERT. len 48
*Mar 15 03:01:50.463: SISF[PRS]: Gi0/0/1 vlan 0 Found 3 options
*Mar 15 03:01:50.463: SISF[PRS]: Gi0/0/1 vlan 0            option 1 : ND_OPT_SOURCE_LINKADDR
*Mar 15 03:01:50.463: SISF[PRS]: Gi0/0/1 vlan 0            option 3 :
ND_OPT_PREFIX_INFORMATION
*Mar 15 03:01:50.463: SISF[PRS]: Gi0/0/1 vlan 0            option 5 : ND_OPT_MTU
*Mar 15 03:01:50.464: SISF[PRS]:
*Mar 15 03:01:50.464: SISF[GLN]: Gi0/0/1 vlan 0 IPv6 snooping Gleaner setting sec level to
 2
*Mar 15 03:01:50.464: SISF[PRS]: Gi0/0/1 vlan 0 Sec level is Guard
*Mar 15 03:01:50.464: SISF[PRS]: Gi0/0/1 vlan 0 Advertise from access: default action is
update entry
*Mar 15 03:01:50.464: SISF[PRS]: Gi0/0/1 vlan 0 Unallowed RA/Redir: default action is delete
 entry
```

```
*Mar 15 03:01:50.464: SISF[GLN]: Gi0/0/1 vlan 0 Unauthorized packet
*Mar 15 03:01:50.464: SISF[SWI]: Gi0/0/1 vlan 0 Feature Snooping rc 1
*Mar 15 03:01:50.464: SISF[SWI]: Gi0/0/1 vlan 0 Feature drop
*Mar 15 03:01:50.464: SISF[MEM]: Unlocking, count is now 0
*Mar 15 03:01:50.464: SISF[MEM]:  3BB56338 semaphore system unlocked
*Mar 15 03:01:54.548: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:01:57.543: SISF[SWI]: SISF IPv6 enqueue FE80::B614:89FF:FE03:2600
*Mar 15 03:01:59.428: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:02:03.896: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:02:08.500: SISF[SWI]: SISF IPv6 enqueue FE80::219:2FFF:FE53:83CE
*Mar 15 03:02:10.266: SISF[SWI]: SISF IPv6 enqueue FE80::213:80FF:FE3E:8B25
ASR1006-1#

Device# end
```

Related Commands

Command	Description
show mdns cache	Displays information about the resource records in the mDNS cache during the mDNS service discovery process.
show mdns requests	Displays information about the browse requests, pending service requests, and pending host resolve requests during the mDNS service discovery process.
show mdns statistics	Displays information about the number of packets sent, received, and dropped in the device during the mDNS service discovery process.

debug mdss

To display the run-time errors and sequence of events for the multicast distributed switching services (MDSS), use the **debug mdss**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mdss commanddebug mdss {**all** | **error** | **event**}
no debug mdss {**all** | **error** | **event**}

Syntax Description

all	Displays both errors and sequence of events for MDSS.
error	Displays the run-time errors for MDSS.
event	Displays the run-time sequence of events for MDSS.

Command Default

Debugging is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following example shows output using the **debug mdss**command with the **all** keyword:

```
Router# debug mdss all
mdss all debugging is on
Router# clear ip mroute *
Router#
01:31:03: MDSS: got MDFS_CLEARALL
01:31:03: MDSS: --> mdss_flush_all_sc
01:31:03: MDSS: enqueue a FE_GLOBAL_DELETE
01:31:03: MDSS: got MDFS_MROUTE_ADD for (0.0.0.0, 224.0.1.40)
01:31:03: MDSS: --> mdss_free_scmdb_cache
01:31:03: MDSS: got MDFS_MROUTE_ADD for (0.0.0.0, 239.255.158.197)
01:31:03: MDSS: got MDFS_MROUTE_ADD for (192.1.21.6, 239.255.158.197)
01:31:03: MDSS: got a MDFS_MIDB_ADD for (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan22
01:31:03: MDSS: -- mdss_add_oif
01:31:03: MDSS: enqueue a FE_OIF_ADD (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan22
01:31:03: MDSS: mdb (192.1.21.6, 239.255.158.197) fast_flags |
MCACHE_MTU
01:31:03: MDSS: got a MDFS_MIDB_ADD for (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan23
01:31:03: MDSS: -- mdss_add_oif
01:31:03: MDSS: enqueue a FE_OIF_ADD (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan23
01:31:03: MDSS: mdb (192.1.21.6, 239.255.158.197) fast_flags |
MCACHE_MTU
01:31:03: MDSS: got a MDFS_MIDB_ADD for (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan24
01:31:03: MDSS: -- mdss_add_oif
01:31:03: MDSS: enqueue a FE_OIF_ADD (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan24
01:31:03: MDSS: mdb (192.1.21.6, 239.255.158.197) fast_flags |
MCACHE_MTU
01:31:03: MDSS: got a MDFS_MIDB_ADD for (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan25
01:31:03: MDSS: -- mdss_add_oif
01:31:03: MDSS: enqueue a FE_OIF_ADD (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan25
01:31:03: MDSS: mdb (192.1.21.6, 239.255.158.197) fast_flags |
MCACHE_MTU
01:31:03: MDSS: got a MDFS_MIDB_ADD for (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan26
01:31:03: MDSS: -- mdss_add_oif
01:31:03: MDSS: enqueue a FE_OIF_ADD (192.1.21.6, 239.255.158.197,
Vlan21) +Vlan26
01:31:03: MDSS: mdb (192.1.21.6, 239.255.158.197) fast_flags |
MCACHE_MTU
01:31:03: MDSS: got a MDFS_MIDB_ADD for (192.1.21.6, 239.255.158.197,u
Vlan21) +Vlan27
```

Related Commands

Command	Description
debug mls rp ip multicast	Displays information about MLSP.

debug media resource provisioning all

To display debugging messages related to all media resource provisioning, use the **debug media resource provisioning all** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug media resource provisioning all
no debug media resource provisioning all

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(8)T	This command was introduced.

Examples

The following is sample output from the **debug media resource provisioning all** command:

```
Router# debug media resource provisioning all

.
.
.
Media resource provisioning all debugging is on.
Disabling profile will disconnect active CONFERENCING calls,
do you want to continue ? [yes/no]
*Jul  8 18:46:36: rpm_if_profile_exist ::profile id 10, service  TRANSCODING
*Jul  8 18:46:36: rpm_get_rscid_profile_info Profile with profile id :10, service :TRANSCODING
 does not exist
*Jul  8 18:46:36: rpm_if_profile_exist ::profile id 10, service  CONFERENCING
*Jul  8 18:46:36: rpm_if_profile_exist ::profile id 10, service  TRANSCODING
*Jul  8 18:46:36: rpm_get_rscid_profile_info Profile with profile id :10, service :TRANSCODING
 does not exist
*Jul  8 18:46:36: rpm_if_profile_exist ::profile id 10, service  CONFERENCING
 Must be yes or no
Router(config-dspfarm-profile)#
Router(config-dspfarm-profile)#
Router(config-dspfarm-profile)#
Router(config-dspfarm-profile)# no shutdown

Router(config-dspfarm-profile)#
*Jul  8 18:46:42: rpm_user_enable_profile ::profile id 10, service CONFERENCING
*Jul  8 18:46:44:%DSPRM-5-UPDOWN:DSP 10 in slot 1, changed state to up
*Jul  8 18:46:44: rpm_rscprv_update ::provider_id 1 rsc_id 2 rsc_grp_state 4num_channel_delta
 0
*Jul  8 18:46:44: rpm_rscprv_update resource update from resource provider 1 is successful
Router(config-dspfarm-profile)#
Router(config-dspfarm-profile)# exit

Router(config)# exit
```

Related Commands

Command	Description
debug media resource provisioning errors	Displays debugging messages related to media resource provisioning errors.
debug media resource provisioning events	Displays debugging messages related to media resource provisioning events.

debug media resource provisioning errors

To display debugging messages related to media resource provisioning errors, use the **debug media resource provisioning errors** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug media resource provisioning errors
no debug media resource provisioning errors

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.3(8)T	This command was introduced.
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.

Examples

The following is sample output from the **debug media resource provisioning errors** command:

```
Router# debug media resource provisioning errors

Media resource provisioning errors debugging is on
Router# no debug media resource provisioning errors

Media resource provisioning errors debugging is off
```

Related Commands

Command	Description
debug media resource provisioning all	Displays debugging messages related to all media resource provisioning.
debug media resource provisioning events	Displays debugging messages related to media resource provisioning events.

debug media resource provisioning events

To display debugging messages related to media resource provisioning events, use the **debug media resource provisioning events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug media resource provisioning events
no debug media resource provisioning events

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.3(8)T	This command was introduced.
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.

Examples

The following is sample output from the **debug media resource provisioning events** command:

```
Router# debug media resource provisioning events

Media resource provisioning events debugging is on
Router# no debug media resource provisioning events

Media resource provisioning events debugging is off
Router#
```

Related Commands

Command	Description
debug media resource provisioning all	Displays debugging messages related to all media resource provisioning.
debug media resource provisioning errors	Displays debugging messages related to media resource provisioning errors.

debug mediacard

To display Digital Signal Processor Resource Manager (DSPRM) debugging information, use the **debug mediacard** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mediacard {**all** | **errors** | **events** | **message**}
no debug mediacard {**all** | **errors** | **events** | **message**}

Syntax Description

all	Debugs DSPRM errors, events, and messages.
errors	Debugs DSPRM errors.
events	Debugs DSPRM events.
message	Debugs DSPRM messages.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(8)XY	This command was introduced on the Communication Media Module.
12.3(14)T	This command was integrated into Cisco IOS Release 12.3(14)T.
12.4(3)	This command was integrated into Cisco IOS Release 12.4(3).

Usage Guidelines

Use the **debug mediacard errors** command to debug active calls. You should use the **debug mediacard all** command during minimum traffic periods only; using the **debug mediacard all** command during active calls can significantly impact system performance.

Examples

The following is sample output from the **debug mediacard** command:

```
Router# debug mediacard messages
Media Card service messages debugging is on
*Mar  1 07:45:06.362: >  CREATE_CONFERENCE (0x1) , pktLen 56, confId 1, instId 1
7483, seqNo 27983, Payload (24 bytes): confType 3, agcMode 1, spkrUpdateReportMo
de 1, maxActSpkr 3
*Mar  1 07:45:06.362: >  CREATE_CHANNEL (0x64) , pktLen 100, confId 1, instId 26
625, seqNo 27984, Payload (68 bytes): rxCodecType 1, suppressRx 1, rxCNG 2, rxPL
C 2, rxVAD 2, rxToneDet 1, rxSpkrPriority 1, rxInactiveTimeOut 7200, rxPacketSiz
e 20, rxRTPPayloadType 0
*Mar  1 07:45:06.362:    txCodecType 2, suppressTx 1, txVAD 1, AGC 1, txSSRC 167
860472, txPacketSize 20, txRTPPayloadType 0
*Mar  1 07:45:06.362:  < CREATE_CONFERENCE_ACK (0x4001) , pktLen 116, confId 1,
instId 0, seqNo 27983, Payload (84 bytes): status 0 (Normal Completion), param1
3, param2 0
*Mar  1 07:45:06.362:  < CREATE_CHANNEL_ACK (0x4064) , pktLen 116, confId 1, ins
tId 26625, seqNo 27984, Payload (84 bytes): status 0 (Normal Completion), param1
 0, param2 0
```

```
*Mar  1 07:45:06.362: >  CREATE_CONFERENCE (0x1) , pktLen 56, confId 2, instId
All possible debugging has been turned off
MTP#26625, seqNo 27985, Payload (24 bytes): confType 3, agcMode 1, spkrUpdateRep
ortMode 1, maxActSpkr 3
*Mar  1 07:45:06.362: >  CREATE_CHANNEL (0x64) , pktLen 100, confId 2, instId 26
626, seqNo 27986, Payload (68 bytes): rxCodecType 2, suppressRx 1, rxCNG 2, rxPL
C 2, rxVAD 2, rxToneDet 1, rxSpkrPriority 1, rxInactiveTimeOut 7200, rxPacketSiz
e 20, rxRTPPayloadType 0
*Mar  1 07:45:06.366:    txCodecType 1, suppressTx 1, txVAD 1, AGC 1, txSSRC 167
858296, txPacketSize 20, txRTPPayloadType 0
*Mar  1 07:45:06.366:  < CREATE_CONFERENCE_ACK (0x4001) , pktLen 116, confId 2,
instId 0, seqNo 27985, Payload (84 bytes): status 0 (Normal Completion), param1
3, param2 0
Router# debug mediacard events
Media Card service events debugging is on
*Mar  1 07:47:53.926: ms_ac_open_rtp_sockets: loc_ipaddr = 10.1.80.24 loc_mac<00
03.feac.c842> rem_ip<0.0.0.0> rem_port<0>
*Mar  1 07:47:53.926: ms_ac_get_unique_udp_port: rtcp_socket = 6255F490
*Mar  1 07:47:53.926: ms_ac_get_unique_udp_port: SLOT3 Port<3450> is assigned!
*Mar  1 07:47:53.926: ms_ac_open_local_rtp: rtpinfo 64382A3C, local_port =23930
*Mar  1 07:47:53.926: ms_ac_rtp_enq: Sent msg 101 to DSPFARM
*Mar  1 07:47:53.926: ms_ac_open_remote_rtp: rtpinfo 64382A3C, loc_ipaddr = 10.1
.80.24 loc_udp_prt <23930> ,loc_mac<0003.feac.c842>
*Mar  1 07:47:53.926: ms_ac_open_remote_rtp: remote_ipaddr = 10.1.2.15 remote_ud
p_prt <17932>
*Mar  1 07:47:53.926: ms_ac_nexthop_macaddr idb<630BDFCC> nexthop<10.1.80.1>
*Mar  1 07:47:53.926: ms_ac_nexthop_macaddr ptr<6301F5AC> through<GigabitEtherne
t1/0> nexthop<10.1.80.1>
*Mar  1 07:47:53.926: ms_ac_after_found_mac <10.1.2.15>'s mac <00d0.002a.7400> f
ound
*Mar  1 07:47:53.926: ms_ac_check_xcode_rem_ip: rtpinfo <64382A3C> other_rtpinfo
<0>
*Mar  1 07:47:53.926: ms_ac_rtp_enq: Sent msg 103 to DSPFARM
*Mar  1 07:47:53.942: ms_ac_open_rtp_sockets: loc_ipaddr = 10.1.80.24 loc_mac<00
03.feac.c842> rem_ip<0.0.0.0> rem_port<0>
*Mar  1 07:47:53.942: ms_ac_get_unique_udp_port: rtcp_socket = 6256C9B4
*Mar  1 07:47:53.942: ms_ac_get_unique_udp_port: SLOT3 Port<1778> is assigned!
*Mar  1 07:47:53.942: ms_ac_open_local_rtp: rtpinfo 6438353C, local_port =22258
*Mar  1 07:47:53.942: ms_ac_rtp_enq: Sent msg 101 to DSPFARM
*Mar  1 07:47:53.942: ac_validate_xcode_params: codeDec<2> codeEnc<1> decDur<20>
 encDur<20>
*Mar  1 07:47:53.942: ac_open_xcode_channel: codeDec<1> codeEnc<2> decDur<20> en
cDur<20> VADen<0> prf_id<4>
*Mar  1 07:47:53.942: reserve_xcode_resource: reserve xcode resource:codecDec<1>
 codecEnc<2>
*Mar  1 07:47:53.942: al
```

Related Commands

Command	Description
show mediacard	Displays information about the media card.

debug memory

To enable debugging on memory, use the **debug memory**command in privileged EXEC mode. To disable memory debugging, use the **no** form of this command.

debug memory [**rmi**]
no debug memory

Syntax Description

rmi	(Optional) Displays debug information related to memory Remote Method Invocation (RMI).

Command Default

Memory debugging is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(2)T	This command was introduced.

Usage Guidelines

The **debug memory** command is used when debugging memory manager operations such as allocating and reallocating memory.

Examples

The following example shows how to enable memory debugging:

```
Router# debug memory
Memory debugging is on
```

The following example shows how to enable memory RMI debugging:

```
Router# debug memory rmi
Memory RMI debugging is on
```

Related Commands

Command	Description
show debug	Displays the types of debugging that are enabled.

debug metadata

To enable debugging for metadata flow information, use the **debug metadata** command in privileged EXEC mode. To disable debugging for metadata flow information, use the **no** form of this command.

debug metadata {**encode-decode** {**details** | **errors** | **events**} | **flow** {**all** | **core** | **table**} | **nbar**}
no debug metadata {**encode-decode** {**details** | **errors** | **events**} | **flow** {**all** | **core** | **table**} | **nbar**}

Syntax Description

encode-decode	Debugs information related to the metadata encoding and decoding mechanism.
details	Debugs details that occurred during the encode-decode process.
errors	Debugs errors that occurred during the encode-decode process.
events	Debugs events that occurred during the encode-decode process.
flow	Debugs details related to metadata flow.
all	Debugs all metadata flow information.
core	Debugs core metadata events information.
table	Debugs metadata flow table information.
nbar	Debugs Network-Based Application Recognition (NBAR) as a source for metadata.

Command Default

Debugging for metadata flow information is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.2(1)T	This command was introduced.
15.2(4)M	This command was modified. The **nbar** keyword was added.

Examples

The following is sample output from the **debug metadata encode-decode details** command. The debug output shows the process for creating the IP information export (IPFIX) template and decoding the metadata information. The last two lines indicate the length, Variable Length Information ID (VLIE), and metadata application name.

```
Device# debug metadata encode-decode details

*Jul 14 03:24:50.395: MED-IPFIX: Hdr: Ver 10 msg len 66
*Jul 14 03:24:50.395: MED-IPFIX: Hdr: Export time = Thu
Jul 14 08:54:50 2011
*Jul 14 03:24:50.395: MED-IPFIX: Hdr: Seq num = 4
*Jul 14 03:24:50.395: MED-IPFIX: Hdr: Obs dom ID = 0
*Jul 14 03:24:50.395: MED-IPFIX: Creating IP FIX Template, 79CD778
```

```
*Jul 14 03:24:50.395: MED-IPFIX: Decoded and saved ID 256 Templates Address 79CD778
*Jul 14 03:24:50.395: MED-IPFIX: Decoding 2 Template fields
*Jul 14 03:24:50.395: MED-IPFIX: len=4 936750775487430656
*Jul 14 03:24:50.395: MED-IPFIX: VLIE len 17 [telepresence-data]
```

The following is sample output from the **debug metadata flow all** command. The first few lines in the output display the addition of an event. Then, the output shows details of ingress and egress interfaces. Next, the display shows various application names and the associated application IDs. Then, Classification types and the matching applications follow.

The last line, "DB Addition Succeded" indicates that an appropriate match was detected and the control plane classification completed successfully.

```
Device# debug metadata flow all

*Jul 14 08:07:23.155: FMD SIG: Process RSVP Event RSVP_FMD_EVENT_PAYLOAD_RECEIVED(1)
*Jul 14 08:07:23.155: FMD : fmd_post_events: posting event 0
*Jul 14 08:07:23.167: FMD Process Event - FMD_RSVP_TRANSPORT_ADD
*Jul 14 08:07:23.167: (fmd_add_event_process): For Source IP/Port : 67372036/1000
*Jul 14 08:07:23.167: FMD DB Lookup: Hash 391
*Jul 14 08:07:23.167: FMD Event for Ingress Interface Ethernet0/0 , Egress Interface
Ethernet0/1
*Jul 14 08:07:23.167: FMD Classification Src Type 96, Len 17, Value telepresence-data
*Jul 14 08:07:23.167: FMD Classification Dest Type 95, Len 4, Value
*Jul 14 08:07:23.167: App name telepresence-data id 218104286 in Metadata local app table
*Jul 14 08:07:23.167: FMD Classification Src Type 96, Len 11, Value webex-audio
*Jul 14 08:07:23.167: FMD Classification Dest Type 95, Len 4, Value
*Jul 14 08:07:23.167: App name webex-audio id 12 in Metadata local app table
*Jul 14 08:07:23.167: FMD Classification Src Type 96, Len 11, Value webex-audio
*Jul 14 08:07:23.167: FMD Classification Dest Type 96, Len 17, Value telepresence-data *Jul
 14 08:07:23.167: FMD Classification Src Type 96, Len 11, Value webex-audio
*Jul 14 08:07:23.167: FMD Classification Dest Type 0, Len 0, Value
*Jul 14 08:07:23.167: FMD Classification: Match Passed for type 95 value Router-201
*Jul 14 08:07:23.167: FMD Classification: Found 1 filters matching
*Jul 14 08:07:23.167: FMD Event: Input policy Matched, Add flow to CFT
*Jul 14 08:07:23.167: FMD Event: PPCP Binding Succeeded
*Jul 14 08:07:23.167: FMD fmd_add_update_ingress_cft_fo : fid 4
*Jul 14 08:07:23.167: FMD Event: Local Flow ID 0
*Jul 14 08:07:23.167: (fmd_add_event_process): Update with Template Address 79CD778, Md
Addr 947F810
*Jul 14 08:07:23.167: fmd_add_ipv4_flow_node_to_hash: Hash 391
*Jul 14 08:07:23.167: FMD Event: DB Addition Succeeded
```

The following is sample output from the **debug metadata nbar** command. The fields are self-explanatory.

```
Device# debug metadata nbar

*May 21 10:22:02.655: FMD NBAR: Successfully activated NBAR for proto id: 64
*May 21 10:22:02.656: FMD NBAR: fmd filter "application telepresence-media"
*May 21 10:22:02.656: FMD NBAR: Match application command found
*May 21 10:22:02.656: FMD NBAR: Successfully activated NBAR for proto id: 113
*May 21 10:22:02.656: FMD NBAR: class_id 0 name class-default
*May 21 10:22:02.656: FMD NBAR: Non Metadata filter type 26. Skipping
```

Related Commands

Command	Description
metadata application-params	Creates new metadata application parameters.
show metadata application table	Displays a list of metadata applications defined on a device.

Command	Description
show metadata flow	Displays the metadata flow information.

debug mgcp

To enable debug traces for Media Gateway Control Protocol (MGCP) errors, events, media, packets, parser, and Call Admission Control (CAC), use the **debug mgcp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp [{**all** | **errors** [**endpoint** *endpoint-name*] | **events** [**endpoint** *endpoint-name*] | **media** [**endpoint** *endpoint-name*] | **nas** | **packets** [{**endpoint** *endpoint-name* | **input-hex**}] | **parser** | **src** | **voipcac**}]
no debug mgcp [{**all** | **errors** | **events** | **media** | **nas** | **packets** | **parser** | **src** | **voipcac**}]

Syntax Description

all	(Optional) Debugs MGCP errors, events, media, packets, parser and builder, and CAC.
errors	(Optional) Debugs MGCP errors.
endpoint *endpoint-name*	(Optional) Debugs MGCP errors, events, media, or packets per endpoint.
events	(Optional) Debugs MGCP events.
media	(Optional) Debugs MGCP tone and signal events.
nas	(Optional) Debugs MGCP network access server (NAS) (data) events.
packets	(Optional) Debugs MGCP packets.
input-hex	(Optional) Debugs MGCP input packets in hexadecimal values.
parser	(Optional) Debugs MGCP parser and builder.
src	(Optional) Debugs MGCP System Resource Check (SRC) CAC information.
voipcac	(Optional) Turns on debugging messages for the Voice over IP (VoIP) CAC process at the MGCP application layer.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	Additional information was displayed for the gateways.
12.1(5)XM, 12.2(2)T	The output was modified to display parameters for the MGCP channel-associated signaling (CAS) PBX and ATM adaptation layer 2 (AAL2) permanent virtual circuit (PVC) features.
12.2(2)XA	The **media** keyword was added. The **endpoint** *endpoint-name*keyword and argument were added as options for the **errors**, **events**, **media,** and **packets** keywords. The **input-hex** keyword option was added for the **packets**keyword.

Release	Modification
12.2(2)XB	The **nas** keyword and the **src** and **voipcac** keywords were added. (Refer to MGCP VoIP Call Admission Control in Cisco IOS Release 12.2(2)XB.)
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T. **Note** The **nas** keyword was not integrated into Cisco IOS Release 12.2(8)T.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.

Usage Guidelines

There is always a performance penalty when using debug commands.

Examples

The following is sample output from the **debug mgcp errors**, **debug mgcp events**, **debug mgcp media**, **debug mgcp nas**, **debug mgcp packets**, **debug mgcp parser**, and **debug mgcp src** commands and keywords. The **debug mgcp all** command and keyword would show a compilation of all this output, including the **debug mgcp voipcac** command and keyword output. Note that using the **debug mgcp all** command and keyword may severely impact network performance.

The following is sample output from the **debug mgcp errors** command and keyword:

```
Router# debug mgcp errors
Unknown network interface type
```

The following is sample output from the **debug mgcp events** command and keyword:

```
Router# debug mgcp events
Media Gateway Control Protocol events debugging is on
Router#
1w1d: MGC stat - 172.19.184.65, total=44, succ=7, failed=21
1w1d: MGCP msg 1
1w1d: remove_old_under_specified_ack:
1w1d: MGC stat - 172.19.184.65, total=44, succ=8, failed=21
1w1d: updating lport with 2427setup_ipsocket: laddr=172.29.248.193, lport=2427,
faddr=172.19.184.65, fport=2427
1w1d: enqueue_ack: ackqhead=0, ackqtail=0, ackp=1DC1D38, msg=21A037C
```

The following is sample output from the **debug mgcp media** command and keyword:

```
Router# debug mgcp media
Media Gateway Control Protocol media events debugging is on
Router#
DYNAMIC payload type
DYNAMIC payload type
*Jan  1 03:02:13.159:mgcp_verify_supp_reqdet_ev
*Jan  1 03:02:13.159:mgcp_verify_supp_signal_ev
*Jan  1 03:02:13.159:process_request_ev- callp 635368FC, voice_if 6353C1F8
*Jan  1 03:02:13.159:process_detect_ev- callp 635368FC, voice_if 6353C1F8
*Jan  1 03:02:13.159:process_signal_ev- callp 635368FC, voice_ifp 6353C1F8
*Jan  1 03:02:13.159:mgcp_process_quarantine_mode- callp 635368FC, voice_if 6353C1F8
*Jan  1 03:02:13.159:mgcp_process_quarantine_mode- new q mode:process=0, loop=0
*Jan  1 03:02:13.179:process_deferred_request_events
*Jan  1 03:02:13.479:mgcp_verify_supp_reqdet_ev
*Jan  1 03:02:13.479:mgcp_verify_supp_signal_ev
```

```
*Jan  1 03:02:13.479:process_request_ev- callp 6353BCCC, voice_if 638C3094
*Jan  1 03:02:13.479:process_detect_ev- callp 6353BCCC, voice_if 638C3094
*Jan  1 03:02:13.479:process_signal_ev- callp 6353BCCC, voice_ifp 638C3094
*Jan  1 03:02:13.479:mgcp_process_quarantine_mode- callp 6353BCCC, voice_if 638C3094
*Jan  1 03:02:13.479:mgcp_process_quarantine_mode- new q mode:process=0, loop=0
*Jan  1 03:02:13.499:process_deferred_request_events
*Jan  1 03:02:13.827:mgcp_verify_supp_reqdet_ev
*Jan  1 03:02:13.827:mgcp_verify_supp_signal_ev
*Jan  1 03:02:13.827:process_request_ev- callp 635368FC, voice_if 6353C1F8
*Jan  1 03:02:13.827:process_detect_ev- callp 635368FC, voice_if 6353C1F8
*Jan  1 03:02:13.827:process_signal_ev- callp 635368FC, voice_ifp 6353C1F8
*Jan  1 03:02:13.827:mgcp_process_quarantine_mode- callp 635368FC, voice_if 6353C1F8
*Jan  1 03:02:13.827:mgcp_process_quarantine_mode- new q mode:process=0, loop=0
*Jan  1 03:02:13.831:process_deferred_request_events
*Jan  1 03:02:23.163:mgcp_cr_and_init_evt_node:$$$ the node pointer 63520B14
*Jan  1 03:02:23.163:mgcp_insert_node_to_preprocess_q:$$$enq to preprocess, qhead=63520B14,
 qtail=63520B14, count 1, evtptr=63520B14
*Jan  1 03:02:23.479:mgcp_cr_and_init_evt_node:$$$ the node pointer 63520BA8
*Jan  1 03:02:23.479:mgcp_insert_node_to_preprocess_q:$$$enq to preprocess, qhead=63520BA8,
 qtail=63520BA8, count 1, evtptr=63520BA8
```

The following is sample output for the **debug mgcp nas** command and keyword, with the **debug mgcp packets** command and keyword enabled as well:

```
Router# debug mgcp nas
Media Gateway Control Protocol nas pkg events debugging is on
Router# debug mgcp packets
Media Gateway Control Protocol packets debugging is on
Router#
01:49:14:MGCP Packet received -
CRCX 58 S7/DS1-0/23 MGCP 1.0
X:57
 M:nas/data
 C:3

 L:b:64, nas/bt:modem, nas/cdn:3000, nas/cgn:1000

mgcp_parse_conn_mode :string past nas = data
mgcp_chq_nas_pkg:Full string:nas/bt:modem
mgcp_chq_nas_pkg:string past slash:bt
mgcp_chq_nas_pkg:string past colon:modem
mgcp_chq_nas_pkg:Full string:nas/cdn:3000
mgcp_chq_nas_pkg:string past slash:cdn
mgcp_chq_nas_pkg:string past colon:3000
mgcp_chq_nas_pkg:Full string:nas/cgn:1000
c5400#
mgcp_chq_nas_pkg:string past slash:cgn
mgcp_chq_nas_pkg:string past colon:1000
CHECK DATA CALL for S7/DS1-0/23
 mgcpapp_xcsp_get_chan_cb -Found - Channel state  Idle
 CRCX Recv
mgcpapp_endpt_is_data:endpt S7/DS1-0/23, slot 7, port 0 chan 23
 mgcpapp_data_call_hnd:mgcpapp_xcsp_get_chan_cb -Found - Channel state  Idle
bw=64, bearer=E1,cdn=3000,cgn=1000
```

The following is sample output from the **debug mgcp packets** command and keyword:

```
Router# debug mgcp packets
Media Gateway Control Protocol packets debugging is on
Router#
1w1d: MGCP Packet received -
DLCX 408631346 * MGCP 0.1
1w1d: send_mgcp_msg, MGCP Packet sent --->
```

```
1w1d: 250 408631346
<---
```

The following is sample output from the **debug mgcp parser** command and keyword:

```
Router# debug mgcp

parser
Media Gateway Control Protocol parser debugging is on
Router#
1w1d: -- mgcp_parse_packet() - call mgcp_parse_header
- mgcp_parse_header()- Request Verb FOUND DLCX
- mgcp_parse_packet() - out mgcp_parse_header
- SUCCESS: mgcp_parse_packet()- MGCP Header parsing was OK
- mgcp_val_mandatory_parms()
- SUCCESS: mgcp_parse_packet()- END of Parsing
1w1d: -- mgcp_build_packet()-
1w1d: - mgcp_estimate_msg_buf_length() - 87 bytes needed for header
- mgcp_estimate_msg_buf_length() - 87 bytes needed after checking parameter lines
- mgcp_estimate_msg_buf_length() - 87 bytes needed after checking SDP lines
- SUCCESS: MGCP message building OK
- SUCCESS: END of building
```

The following is sample output from the **debug mgcp src** command and keyword:

```
Router# debug mgcp src
Media Gateway Control Protocol System Resource Check CAC debugging is on
Router#
00:14:08: setup_indication: Set incoming_call flag=TRUE in voice_if
00:14:08: send_mgcp_msg, MGCP Packet sent --->
00:14:08: NTFY 11 aaln/S1/1@Router MGCP 0.1
N: emu@[1.4.173.1]:51665
X: 35
O: hd
<---
00:14:08: MGCP Packet received -
200 11 hello
00:14:08: MGCP Packet received -
RQNT 42 aaln/S1/1 MGCP 0.1
N: emu@[1.4.173.1]:51665
X: 41
R: D/[0-9*#T](d), hu
S: dl
D: (911|xxxx)
00:14:08: send_mgcp_msg, MGCP Packet sent --->
00:14:08: 200 42 OK
<---
00:14:12: send_mgcp_msg, MGCP Packet sent --->
00:14:12: NTFY 12 aaln/S1/1@Router MGCP 0.1
N: emu@[1.4.173.1]:51665
X: 41
O: D/2222
<---
00:14:12: MGCP Packet received -
200 12 phone-number ok
00:14:12: MGCP Packet received -
CRCX 44 aaln/S1/1 MGCP 0.1
N: emu@[1.4.173.1]:51665
C: 3
X: 43
R: hu(n)
M: recvonly
L: a:G.711u,p:5,e:off,s:off
```

```
00:14:12: mgcp_setup_conn_check_system_resource: System resource check successful
00:14:12: mgcp_voice_crcx: System resource is available
00:14:12: mgcp_set_call_counter_control: Incoming call with 1 network leg, flag=FALSE
00:14:12: send_mgcp_msg, MGCP Packet sent --->
00:14:12: 200 44
I: 4
v=0
o=- 4 0 IN IP4 1.4.120.1
s=Cisco SDP 0
c=IN IP4 1.4.120.1
t=0 0
m=audio 16404 RTP/AVP 0
<---
00:14:13: MGCP Packet received -
MDCX 48 aaln/S1/1 MGCP 0.1
N: emu@[1.4.173.1]:51665
C: 3
I: 4
X: 47
M: recvonly
R: hu
L: a:G.711u,p:5,e:off,s:off
v=0
o=- 4 0 IN IP4 1.4.120.3
s=Cisco SDP 0
c=IN IP4 1.4.120.3
t=0 0
m=audio 16384 RTP/AVP 0
00:14:13: mgcp_modify_conn_check_system_resource: System resource check successful
00:14:13: mgcp_modify_connection: System resource is available
00:14:13: send_mgcp_msg, MGCP Packet sent --->
00:14:13: 200 48 OK
<---
00:14:20: MGCP Packet received -
MDCX 52 aaln/S1/1 MGCP 0.1
N: emu@[1.4.173.1]:51665
C: 3
I: 4
X: 51
M: sendrecv
R: hu
L: a:G.711u,p:5,e:off,s:off
00:14:20: mgcp_modify_conn_check_system_resource: System resource check successful
00:14:20: mgcp_modify_connection: System resource is available
00:14:20: send_mgcp_msg, MGCP Packet sent --->
00:14:20: 200 52 OK
<---
00:14:34: MGCP Packet received -
DLCX 56 aaln/S1/1 MGCP 0.1
X: 55
N: emu@[1.4.173.1]:51665
C: 3
I: 4
R: hu
00:14:34: send_mgcp_msg, MGCP Packet sent --->
00:14:34: 250 56
P: PS=1382, OS=110180, PR=1378, OR=109936, PL=63484, JI=520, LA=2
<---
00:14:36: mgcp_reset_call_direction: Reseting incoming_call flag=FALSE in voice_if
00:14:36: send_mgcp_msg, MGCP Packet sent --->
00:14:36: NTFY 13 aaln/S1/1@tlkrgw1 MGCP 0.1
N: emu@[1.4.173.1]:51665
X: 55
```

```
O: hu
<---
```

debug mgcp all

To enable all debug traces for Media Gateway Control Protocol (MGCP), use the **debug mgcp all**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp all [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp all

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	Additional information was displayed for the gateways.
12.1(5)XM, 12.2(2)T	The output was modified to display parameters for the MGCP channel-associated signaling (CAS) PBX and ATM adaptation layer 2 (AAL2) permanent virtual circuit (PVC) features.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

This command enables the following MGCP debug commands:

- **debug mgcp endptdb**
- **debug mgcp errors**
- **debug mgcp events**
- **debug mgcp gcfm**
- **debug mgcp inout**
- **debug mgcp media**

- **debug mgcp nas**
- **debug mgcp packets**
- **debug mgcp parser**
- **debug mgcp src**
- **debug mgcp state**
- **debug mgcp voipcac**

Caution Using the **debug mgcp all** command may severely impact network performance.

Examples

The following is sample output from the **debug mgcp all**command:

```
Router# debug mgcp all
This may severely impact network performance. Continue[confirm]
Media Gateway Control Protocol all debugging is on, trace-level Verbose
Router#
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_count_active_mgc_msg_stat(240):[lvl=1]MGC
 stat - 192.168.1.200, total=8, succ=5, failed=1
*Sep 10 17:20:24.408: MGCP Packet received from 192.168.1.200:7979--->
CRCX 6 aaln/S2/SU1/1 MGCP 1.0
M: recvonly
C: 1
<---
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcpapp_process_mgcp_msg(3318):[lvl=0] : <NEW
 MGCP MSG From CA>
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(316):[lvl=0]call
mgcp_parse_header
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(320):[lvl=0]out
mgcp_parse_header
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(360):[lvl=1]SUCCESS: - MGCP
 Header parsing was OK
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_string_parse(186):[lvl=0]return code=1.
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_parameter_lines(725):[lvl=1]return
 parse function in mgcp_parm_rules_array[6].
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4762):[lvl=0](in_ptr:
recvonly)
*Sep 10 17:20:24.408:
//-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4780):[lvl=0]tmp_ptr:(recvonly)
*Sep 10 17:20:24.408:
//-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4816):[lvl=0]tmp_ptr:(recvonly)
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4822):[lvl=0]match recvonly
 recvonly
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4830):[lvl=0]case
MODE_RECVONLY
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4894):[lvl=0]SUCCESS:
Connection Mode parsing is OK
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_string_parse(186):[lvl=0]return code=1.
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_parameter_lines(725):[lvl=1]return
 parse function in mgcp_parm_rules_array[1].
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_call_id(840):[lvl=0]in_ptr: 1
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_call_id(883):[lvl=1]SUCCESS: Call
ID string(1) parsing is OK
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_mandatory_parms(12428):[lvl=0]Entered
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_comp_mp_parms(14923):[lvl=0]Entered
```

```
*Sep 10 17:20:24.408: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_comp_mp_parms(14928):[lvl=1] -
lcon_opt_ptr could not be obtained
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(378):[lvl=2]SUCCESS: END of
 Parsing
*Sep 10 17:20:24.412:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_a(1339):[lvl=0]aaln/S2/SU1/1
*Sep 10 17:20:24.412:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_slot(1632):[lvl=0]2/SU1/1
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]2/SU1/1
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_slot(1641):[lvl=0]
 : ifn 0x665449A8, slot:2
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_su(1773):[lvl=0]1/1
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]1/1
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_port(1807):[lvl=0]1
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]1
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_endpt_offset(2590):[lvl=0]endpt
 NULL
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_by_ifn(1326):[lvl=0]Entered
*Sep 10 17:20:24.412:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_tree_link_by_ifn(1145):[lvl=0]Entered
*Sep 10 17:20:24.412: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_compute_key(196):[lvl=0]type 2 slot
 0002 subunit 0001
*Sep 10 17:20:24.412:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_tree_link_by_ifn(1157):[lvl=0]computed key 0x2081FF01
```

Related Commands

Command	Description
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show debugging	Displays the types of debugging that are enabled.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp endpoint

To enable debug traces for a specific Media Gateway Control Protocol (MGCP) endpoint, use the **debug mgcp endpoint** command in privileged EXEC mode. To disable debugging output for the endpoint, use the **no** form of this command.

debug mgcp endpoint *endpoint-name* {**all** [**tracelevel** {**critical** | **moderate** | **verbose**}] | **errors** | **events** [**tracelevel** {**critical** | **moderate** | **verbose**}] | **media** [**tracelevel** {**critical** | **moderate** | **verbose**}] | **packets**}
no debug mgcp endpoint *endpoint-name* {**all** | **errors** | **events** | **media** | **packets**}

Syntax Description

endpoint-name	Name of the MGCP endpoint for which to enable debugging. Must be a fully specified and supported endpoint.
all	Displays MGCP errors, events, media, and packets for the specified endpoint.
errors	Displays MGCP errors for the specified endpoint.
events	Displays MGCP events for the specified endpoint.
media	Displays MGCP tone and signal events for the specified endpoint.
packets	Displays MGCP packets for the specified endpoint.
tracelevel	(Optional) Sets the priority level for the **all**, **events**, or **media** debug trace. • **critical** --Displays only high-priority debug information. • **moderate** --Displays medium and high-priority debug information. • **verbose** --Displays all debug information. This is the default level. **Note** This keyword is not available for **errors** or **packets** debugging.

Command Default

Debugging for specific endpoints is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(4)T	This command was introduced.

Usage Guidelines

This command enables debugging for a specific MGCP endpoint. You can enable the same type of debugging globally for all endpoints by using the **debug mgcp all**, **debug mgcp errors**, **debug mgcp events**, **debug mgcp media**, or **debug mgcp packets** commands.

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

This command sets the trace level for the specific endpoint. You can set the trace level globally for all MGCP debug commands and endpoints by using the **debug mgcp tracelevel-default** command. Setting the endpoint-specific trace level takes precedence over the global trace-level.

Note Trace levels are not supported for errors or packets debugging because all of the output from those commands is set to high priority.

Examples

The following is sample output from the **debug mgcp endpoint**command:

```
Router# debug mgcp endpoint aaln/S2/SU1/1 events tracelevel critical
Media Gateway Control Protocol events debugging for endpoint aaln/S2/SU1/1 is on, trace-level
 Critical
Router#
*Sep 10 17:46:13.100:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_idle_crcx(4875):[lvl=2]callp(0x63E313E0),
current state CALL_IDLE, event EV_CREATE_CONN
*Sep 10 17:46:13.100:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E313E0), new state CALL_CONNECTING, event EV_CREATE_CONN
*Sep 10 17:46:13.104:
//8/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_call_pre_conference(223):[lvl=2]callp(0x63E311D0),
 current state CALL_CONNECTING, event EV_CALL_CONNECT
*Sep 10 17:46:13.104:
//8/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_call_connect(7331):[lvl=2]callp(0x63E311D0),
 current state CALL_CONNECTING, event EV_CALL_CONNECT
*Sep 10 17:46:13.104:
//8/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E311D0), new state CALL_CONFERENCING, event EV_CALL_CONNECT
*Sep 10 17:46:13.104:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_call_proceeding(6306):[lvl=2]callp(0x63E313E0),
 current state CALL_CONNECTING, event EV_CALL_PROCEED
*Sep 10 17:46:13.104:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_call_connect(7331):[lvl=2]callp(0x63E313E0),
 current state CALL_CONNECTING, event EV_CALL_PROCEED
*Sep 10 17:46:13.104:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E313E0), new state CALL_CONFERENCING, event EV_CALL_PROCEED
*Sep 10 17:46:13.108:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_ignore_ccapi_ev(4316):[lvl=2]callp(0x63E313E0),
 current state CALL_CONFERENCING, event EV_CONF_RDY
*Sep 10 17:46:13.108:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E313E0), new state CALL_CONFERENCING, event EV_CONF_RDY
*Sep 10 17:46:13.108:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_call_modified(7710):[lvl=2]callp(0x63E313E0),
 current state CALL_CONFERENCING, event EV_MODIFY_DONE
*Sep 10 17:46:13.108:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E313E0), new state CALL_CONFERENCING, event EV_MODIFY_DONE
*Sep 10 17:46:13.108:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_voice_mode_done(7994):[lvl=2]callp(0x63E313E0),
 current state CALL_CONFERENCING, event EV_VOICE_MODE_DONE, minor ev(d): 138, minor ev
*Sep 10 17:46:13.112:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E313E0), new state CALL_ACTIVE, event EV_VOICE_MODE_DONE
*Sep 10 17:46:23.104:
//7/9D04EB218005/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_invoke_app_sm(570):[lvl=2]MGCP:FSM
done- callp(63E313E0), new state CALL_ACTIVE, event EV_MEDIA_EVT
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp errors	Enables debug traces for MGCP errors.
debug mgcp events	Enables debug traces for MGCP events.
debug mgcp media	Enables debug traces for MGCP tone and signal events.
debug mgcp packets	Enables debug traces for MGCP packets.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show debugging	Displays the types of debugging that are enabled.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp endptdb

To enable debug traces for all Media Gateway Control Protocol (MGCP) endpoints, use the **debug mgcp endptdb** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp endptdb [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp endptdb

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP debugging for endpoints is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XA	This command was introduced.
12.2(11)T	This command was integrated into Cisco IOS Release 12.2(11)T.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

This command enables debugging globally for all MGCP endpoints. You can limit debugging to a specific endpoint by using the **debug mgcp endpoint** command.

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp endptdb** command used with the **debug mgcp packets**command:

```
Router# debug mgcp packets
Media Gateway Control Protocol packets debugging for all endpoints is on
Router# debug mgcp endptdb
Media Gateway Control Protocol endpoint database debugging for all endpoints is on,
trace-level Verbose
Router#
*Sep 10 11:39:16.467: MGCP Packet received from 192.168.1.200:7979--->
CRCX 27 aaln/S2/SU1/1 MGCP 1.0
M: recvonly
C: 1
<---
```

```
*Sep 10 11:39:16.467:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_a(1339):[lvl=0]aaln/S2/SU1/1
*Sep 10 11:39:16.467:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_slot(1632):[lvl=0]2/SU1/1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]2/SU1/1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_slot(1641):[lvl=0]
 : ifn 0x665449A8, slot:2
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_su(1773):[lvl=0]1/1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]1/1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_port(1807):[lvl=0]1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_by_ifn(1326):[lvl=0]Entered
*Sep 10 11:39:16.467:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_tree_link_by_ifn(1145):[lvl=0]Entered
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_compute_key(196):[lvl=0]type 2 slot
 0002 subunit 0001
*Sep 10 11:39:16.467:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_tree_link_by_ifn(1157):[lvl=0]computed key 0x2081FF01
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_state(3758):[lvl=0]endpt
aaln/S2/SU1/1
*Sep 10 11:39:16.467: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_default_get_state(3779):[lvl=0]endpt
 aaln/S2/SU1/1
*Sep 10 11:39:16.479: MGCP Packet sent to 192.168.1.200:7979--->
200 27 OK
I: D
v=0
c=IN IP4 192.168.1.79
m=audio 16870 RTP/AVP 0 8 99 101 102 2 15 103 4 104 105 106 107 18 100
a=rtpmap:99 G.729a/8000
a=rtpmap:101 G.726-16/8000
a=rtpmap:102 G.726-24/8000
a=rtpmap:103 G.723.1-H/8000
a=rtpmap:104 G.723.1-L/8000
a=rtpmap:105 G.729b/8000
a=rtpmap:106 G.723.1a-H/8000
a=rtpmap:107 G.723.1a-L/8000
a=rtpmap:100 X-NSE/8000
a=fmtp:100 200-202
a=X-sqn:0
a=X-cap: 1 audio RTP/AVP 100
a=X-cpar: a=rtpmap:100 X-NSE/8000
a=X-cpar: a=fmtp:100 200-202
a=X-cap: 2 image udptl t38
<---
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp errors

To enable debug traces for Media Gateway Control Protocol (MGCP) errors, use the **debug mgcp errors**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp errors
no debug mgcp errors

Syntax Description

This command has no arguments or keywords.

Command Default

MGCP error debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	Additional information was displayed for the gateways.
12.1(5)XM, 12.2(2)T	The output was modified to display parameters for the MGCP channel-associated signaling (CAS) PBX and ATM adaptation layer 2 (AAL2) permanent virtual circuit (PVC) features.
12.2(2)XA	The **endpoint** *endpoint-name*keyword and argument were added.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **endpoint** *endpoint-name*keyword and argument were removed and replaced by the **debug mgcp endpoint** command.

Usage Guidelines

This command enables error debugging globally for all MGCP endpoints. You can limit debugging to a specific endpoint by using the **debug mgcp endpoint** command.

Examples

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

```
Router# debug mgcp errors
*Oct 16 12:09:05.538: MGC stat - 10.208.237.83, total=1029, succ=998, failed=0-
mgcp_parse_header()- Request Verb FOUND AUEP
- mgcp_parse_request_header()- MGCP_V10, start check for profile
- mgcp_parse_header: mgcp_parse_request_header returns status: 0
*Oct 16 12:09:05.538: MGCP Packet received from 10.208.237.83-
AUEP 9634549 S0/DS1-0/1@AS5300 MGCP 1.0
F: I
*Oct 16 12:09:05.542: -- mgcp_parse_packet() - call mgcp_parse_header
- mgcp_parse_header()- Request Verb FOUND AUEP
- mgcp_parse_request_header()- MGCP_V10, start check for profile
- mgcp_parse_header: mgcp_parse_request_header returns status: 0
- mgcp_parse_packet() - out mgcp_parse_header
- SUCCESS: mgcp_parse_packet()-MGCP Header parsing was OK
- mgcp_parse_parameter_lines(), code_str:: I, code_len:2, str:F: I
- mgcp_parse_parameter_lines(str:F: I) -num_toks: 28
- mgcp_parse_parameter_lines() check NULL str(I), in_ptr(F: I)
- mgcp_parse_parameter_lines() return Parse function in mgcp_parm_rules_array[14]
- mgcp_parse_req_info(I) is called
- mgcp_parse_req_info() - tmp_ptr:(I)
- SUCCESS: Request Info parameter line (F:) parsing OK
- mgcp_val_mandatory_parms()
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp events

To enable debug traces for Media Gateway Control Protocol (MGCP) events, use the **debug mgcp events**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp events [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp events

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP events debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	Additional information was displayed for the gateways.
12.1(5)XM, 12.2(2)T	The output was modified to display parameters for the MGCP channel-associated signaling (CAS) PBX and ATM adaptation layer 2 (AAL2) permanent virtual circuit (PVC) features.
12.2(2)XA	The **endpoint** *endpoint-name*keyword and argument were added.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **endpoint** *endpoint-name* keyword and argument were removed and replaced by the **debug mgcp endpoint** command. The **tracelevel** keyword was added.

Usage Guidelines

This command enables events debugging globally for all MGCP endpoints. You can limit debugging to a specific endpoint by using the **debug mgcp endpoint** command.

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp events** command:

```
Router# debug mgcp events
Media Gateway Control Protocol events debugging for all endpoints is on, trace-level Verbose
Router#
*Sep 10 09:22:41.276: //-1/xxxxxxxxxxxx/MGCP/mgcpapp_stw_call_back(316):[lvl=0]timer type
1
*Sep 10 09:22:41.276: //-1/xxxxxxxxxxxx/MGCP/mgcpapp_process_timers(1431):[lvl=0]timer of
type 1 expired.
*Sep 10 09:22:41.276:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_remove_old_ack(712):[lvl=1]Removing ack:
(trans ID 15) : 250 15 OK
P: PS=0, OS=0, PR=0, OR=0, PL=0, JI=0, LA=0
*Sep 10 09:22:42.300: //-1/xxxxxxxxxxxx/MGCP/mgcp_count_active_mgc_msg_stat(240):[lvl=1]MGC
 stat - 192.168.1.200, total=18, succ=14, failed=2
*Sep 10 09:22:42.300: //-1/xxxxxxxxxxxx/MGCP/mgcpapp_process_mgcp_msg(3318):[lvl=0] : <NEW
 MGCP MSG From CA>
*Sep 10 09:22:42.300: //-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_endpt_offset(2590):[lvl=0]endpt
 NULL
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcpapp_setup_per_call_data(2487):[lvl=1]mgcpapp_setup_per_call_data:
 callp: 63E313E0, vdbptr: 65822AF8, state: CALL_IDLE
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_notified_entity(439):[lvl=0]Entered
*Sep 10 09:22:42.300: //-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_notified_entity(458):[lvl=1]ne
 callagenthost:7979, ne addr 192.168.1.200:7979
*Sep 10 09:22:42.300: //-1/xxxxxxxxxxxx/MGCP/xlate_mgcp_ev(921):[lvl=1]hdr_type 1
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcpapp_process_mgcp_event(2615):[lvl=1]Processing
 Incoming Message [CRCX 16]
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcpapp_invoke_mgcp_sm(2559):[lvl=1]Msg
In-Progress(Active) [INVVERB 0], await_ev=0, queued=0x00000000
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_process_deferred_queue(3362):[lvl=0]Entered
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP/mgcp_store_endpt_and_ntfy_entity_name(4464):[lvl=0]Entered
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_invoke_app_sm(535):[lvl=0]MGCP:calling FSM-
 callp(63E313E0)
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_idle_crcx(4875):[lvl=2]callp(0x63E313E0),
current state CALL_IDLE, event EV_CREATE_CONN
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP/mgcp_init_modem_relay_params(103):[lvl=0]modem-relay-enabled=0,
mr-gw-xid=0
*Sep 10 09:22:42.300:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_compute_debugsy_hdr(274):[lvl=0]Building
Debugsy header
*Sep 10 09:22:42.300:
//-1/C537F3F38008/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_compute_debugsy_hdr(383):[lvl=0]GUID[C537F3F38008]
 assigned to call_id[-1], endpt[aaln/S2/SU1/1], mgcp_call_id[n/a], conn_id[0]
*Sep 10 09:22:42.300:
//-1/C537F3F38008/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_idle_crcx(4961):[lvl=0]calls
mgcp_allocate_if()
*Sep 10 09:22:42.300:
//-1/C537F3F38008/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_idle_crcx(5006):[lvl=1]get capability
*Sep 10 09:22:42.300:
//-1/C537F3F38008/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_compute_debugsy_hdr(274):[lvl=0]Building
 Debugsy header
*Sep 10 09:22:42.304:
//-1/C537F3F38008/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_compute_debugsy_hdr(383):[lvl=0]GUID[C537F3F38008]
 assigned to call_id[-1], endpt[aaln/S2/SU1/1], mgcp_call_id[1], conn_id[0]
*Sep 10 09:22:42.304:
//-1/C537F3F38008/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_idle_crcx(5093):[lvl=0]Default aal2
```

```
 vc = 1 1-pvc,2-svc
*Sep 10 09:22:42.304: //-1/xxxxxxxxxxxx/MGCP/mgcp_init_vox_if_record(6781):[lvl=0]reusing
records. conn_type: 2, vox if_type: 1
*Sep 10 09:22:42.304: //-1/xxxxxxxxxxxx/MGCP/mgcp_compute_debugsy_hdr(274):[lvl=0]Building
 Debugsy header
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp gcfm

To enable generic call filter module (GCFM) debug traces for Media Gateway Control Protocol (MGCP), use the **debug mgcp gcfm** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp gcfm [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp gcfm

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP GCFM debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(4)T	This command was introduced.

Usage Guidelines

This command enables GCFM debugging globally for all MGCP endpoints.

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp gcfm**command:

```
Router# debug mgcp gcfm
Media Gateway Control Protocol gcfm debugging for all endpoints is on, trace-level Verbose
Router#
*Sep 10 09:24:52.692:
//-1/12F030978009/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_gcfm_percall_register(315):[lvl=2]GCFM
 Inactive
*Sep 10 09:24:52.692:
//-1/12F030978009/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_gcfm_percall_register(315):[lvl=2]GCFM
 Inactive
Router#
```

Related Commands

Command	Description
debug call filter inout	Displays the debug trace inside the GCFM.

Command	Description
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp inout

To enable debug traces for all Media Gateway Control Protocol (MGCP) entry and exit endpoints, use the **debug mgcp inout**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp inout [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp inout

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

Debugging of MGCP entry and exit endpoints is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(4)T	This command was introduced.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp inout** command:

```
Router# debug mgcp inout
Media Gateway Control Protocol inout debugging for all endpoints is on, trace-level Verbose
Router#
*Sep 10 09:26:37.780: //-1/xxxxxxxxxxxx/MGCP/mgcp_count_active_mgc_msg_stat(240):[lvl=1]MGC
 stat - 192.168.1.200, total=22, succ=18, failed=2
*Sep 10 09:26:37.780: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(316):[lvl=0]call
mgcp_parse_header
*Sep 10 09:26:37.780: //-1/xxxxxxxxxxxx/MGCP/mgcp_string_parse(186):[lvl=0]return code=1.
*Sep 10 09:26:37.780: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4762):[lvl=0](in_ptr:
recvonly)
*Sep 10 09:26:37.780: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4894):[lvl=0]SUCCESS:
Connection Mode parsing is OK
*Sep 10 09:26:37.780: //-1/xxxxxxxxxxxx/MGCP/mgcp_string_parse(186):[lvl=0]return code=1.
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_call_id(840):[lvl=0]in_ptr: 1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_call_id(883):[lvl=1]SUCCESS: Call
ID string(1) parsing is OK
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_mandatory_parms(12428):[lvl=0]Entered
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_a(1339):[lvl=0]aaln/S2/SU1/1
```

```
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_slot(1632):[lvl=0]2/SU1/1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]2/SU1/1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_su(1773):[lvl=0]1/1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]1/1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_aaln_port(1807):[lvl=0]1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_name_parse_digit(1600):[lvl=0]1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_endpt_offset(2590):[lvl=0]endpt
 NULL
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_by_ifn(1326):[lvl=0]Entered
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_tree_link_by_ifn(1145):[lvl=0]Entered
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_compute_key(196):[lvl=0]type 2 slot
 0002 subunit 0001
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_get_state(3758):[lvl=0]endpt
aaln/S2/SU1/1
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xgcp_endpt_default_get_state(3779):[lvl=0]endpt
 aaln/S2/SU1/1
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_notified_entity(439):[lvl=0]Entered
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/mgcp_endpt_get_notified_entity(458):[lvl=1]ne
 callagenthost:7979, ne addr 192.168.1.200:7979
*Sep 10 09:26:37.784: //-1/xxxxxxxxxxxx/MGCP/xlate_mgcp_ev(921):[lvl=1]hdr_type 1
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcpapp_invoke_mgcp_sm(2559):[lvl=1]Msg
In-Progress(Active) [INVVERB 0], await_ev=0, queued=0x00000000
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_process_deferred_queue(3362):[lvl=0]Entered
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP/mgcp_store_endpt_and_ntfy_entity_name(4464):[lvl=0]Entered
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_idle_crcx(4875):[lvl=2]callp(0x63E313E0),
current state CALL_IDLE, event EV_CREATE_CONN
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP/mgcp_init_modem_relay_params(103):[lvl=0]modem-relay-enabled=0,
mr-gw-xid=0
*Sep 10 09:26:37.784:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcp_compute_debugsy_hdr(274):[lvl=0]Building
Debugsy header
*Sep 10 09:26:37.784:
//-1/5193F3E0800A/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_compute_debugsy_hdr(383):[lvl=0]GUID[5193F3E0800A]
 assigned to call_id[-1], endpt[aaln/S2/SU1/1], mgcp_call_id[n/a], conn_id[0]
*Sep 10 09:26:37.784:
//-1/5193F3E0800A/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_verify_supp_reqdet_ev(10645):[lvl=0]Entered
*Sep 10 09:26:37.784:
//-1/5193F3E0800A/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_verify_supp_signal_ev(10685):[lvl=0]Entered
*Sep 10 09:26:37.784:
//-1/5193F3E0800A/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_compute_debugsy_hdr(274):[lvl=0]Building
 Debugsy header
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.

Command	Description
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp media

To enable debug traces for Media Gateway Control Protocol (MGCP) tone and signal events, use the **debug mgcp media**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp media [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp media

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP media debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XA	This command was introduced.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **endpoint** *endpoint-name*keyword and argument were removed and replaced by the **debug mgcp endpoint** command. The **tracelevel** keyword was added.

Usage Guidelines

This command enables media debugging globally for all MGCP endpoints. You can limit debugging to a specific endpoint by using the **debug mgcp endpoint** command.

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp media** command:

```
Router# debug mgcp media
```

```
Media Gateway Control Protocol media events debugging for all endpoints is on, trace-level
 Verbose
Router#
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_verify_supp_reqdet_ev(10645):[lvl=0]Entered
*Sep 10 09:27:48.928:
```

```
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_verify_supp_signal_ev(10685):[lvl=0]Entered
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/process_request_ev(5800):[lvl=1]callp
63E313E0, voice_if 6663CA38
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/process_detect_ev(6007):[lvl=0]callp
63E313E0, voice_if 6663CA38
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/process_signal_ev(5500):[lvl=0]callp
63E313E0, voice_ifp 6663CA38
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_process_quarantine_mode(6096):[lvl=0]callp
 63E313E0, voice_if 6663CA38
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_process_quarantine_mode(6149):[lvl=0]Q
 mode not found, Reset default values
*Sep 10 09:27:48.928:
//-1/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_process_quarantine_mode(6168):[lvl=1]Q
 mode: process=0, loop=0
*Sep 10 09:27:48.936:
//19/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_process_pending_t38_port_switch(1649):[lvl=1]conn_recp->conn_id:
 0x0
*Sep 10 09:27:48.940:
//19/7BFBA9F9800B/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/process_deferred_request_events(5724):[lvl=0]Entered
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp nas

To enable network access server (NAS) (data) events for Media Gateway Control Protocol (MGCP), use the **debug mgcp nas**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp nas [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp nas

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP NAS event debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XB	This command was introduced.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output for the **debug mgcp nas** command with the **debug mgcp packets** command also enabled:

```
Router# debug mgcp nas
Media Gateway Control Protocol nas pkg events debugging for all endpoints is on, trace-level
 Verbose
Router# debug mgcp packets
Media Gateway Control Protocol packets debugging for all endpoints is on
Router#
*Sep 10 11:51:41.863: MGCP Packet received from 192.168.1.200:7979--->
CRCX 34 aaln/S2/SU1/1 MGCP 1.0
X:57
M: nas/data
C:3
```

```
L:b:64, nas/bt:modem, nas/cdn:3000, nas/cgn:1000
C: 1
<---
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(779):[lvl=0]Full string:
nas/bt:modem
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(791):[lvl=1]string past slash:
 bt
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(792):[lvl=1]string past colon:
 modem
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(779):[lvl=0]Full string:
nas/cdn:3000
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(791):[lvl=1]string past slash:
 cdn
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(792):[lvl=1]string past colon:
 3000
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(779):[lvl=0]Full string:
nas/cgn:1000
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(791):[lvl=1]string past slash:
 cgn
*Sep 10 11:51:41.863: //-1/xxxxxxxxxxxx/MGCP/mgcp_chq_nas_pkg(792):[lvl=1]string past colon:
 1000
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp packets

To enable debug traces for Media Gateway Control Protocol (MGCP) packets, use the **debug mgcp packets**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp packets [**input-hex**]
no debug mgcp packets [**input-hex**]

Syntax Description

input-hex	(Optional) Displays MGCP incoming packets in hexadecimal format.

Command Default

MGCP packets debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	Additional information was displayed for the gateways.
12.1(5)XM, 12.2(2)T	The output was modified to display parameters for the MGCP channel-associated signaling (CAS) PBX and ATM adaptation layer 2 (AAL2) permanent virtual circuit (PVC) features.
12.2(2)XA	The **endpoint** *endpoint-name*keyword and argument and the **input-hex** keyword were added.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **endpoint** *endpoint-name*keyword and argument were removed and replaced by the **debug mgcp endpoint** command.

Usage Guidelines

This command enables packet debugging globally for all MGCP endpoints. You can limit debugging to a specific endpoint by using the **debug mgcp endpoint** command.

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp packets** command:

```
Router# debug mgcp packets
Media Gateway Control Protocol packets debugging for all endpoints is on
Router#
*Sep 10 11:57:26.795: MGCP Packet received from 192.168.1.200:7979--->
CRCX 38 aaln/S2/SU1/1 MGCP 1.0
M: recvonly
```

```
C: 1
<---
*Sep 10 11:57:26.795:
//-1/xxxxxxxxxxxx/MGCP|aaln/S2/SU1/1|-1|-1/mgcpapp_invoke_mgcp_sm(2569):[lvl=0]CHECK DATA
CALL for aaln/S2/SU1/1
*Sep 10 11:57:26.807: MGCP Packet sent to 192.168.1.200:7979--->
200 38 OK
I: 10
v=0
c=IN IP4 192.168.1.79
m=audio 18876 RTP/AVP 0 8 99 101 102 2 15 103 4 104 105 106 107 18 100
a=rtpmap:99 G.729a/8000
a=rtpmap:101 G.726-16/8000
a=rtpmap:102 G.726-24/8000
a=rtpmap:103 G.723.1-H/8000
a=rtpmap:104 G.723.1-L/8000
a=rtpmap:105 G.729b/8000
a=rtpmap:106 G.723.1a-H/8000
a=rtpmap:107 G.723.1a-L/8000
a=rtpmap:100 X-NSE/8000
a=fmtp:100 200-202
a=X-sqn:0
a=X-cap: 1 audio RTP/AVP 100
a=X-cpar: a=rtpmap:100 X-NSE/8000
a=X-cpar: a=fmtp:100 200-202
a=X-cap: 2 image udptl t38
<---
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp parser

To enable debug traces for the Media Gateway Control Protocol (MGCP) parser and builder, use the **debug mgcp parser** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp parser [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp parser

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP parser and builder debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.1(3)T	Additional information was displayed for the gateways.
12.1(5)XM, 12.2(2)T	The output was modified to display parameters for the MGCP channel-associated signaling (CAS) PBX and ATM adaptation layer 2 (AAL2) permanent virtual circuit (PVC) features.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp parser** command:

```
Router# debug mgcp parser
```

```
Media Gateway Control Protocol parser debugging for all endpoints is on, trace-level Verbose
Router#
```

```
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(316):[lvl=0]call
mgcp_parse_header
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(320):[lvl=0]out
mgcp_parse_header
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(360):[lvl=1]SUCCESS: - MGCP
 Header parsing was OK
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_string_parse(186):[lvl=0]return code=1.
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_parameter_lines(725):[lvl=1]return
 parse function in mgcp_parm_rules_array[6].
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4762):[lvl=0](in_ptr:
recvonly)
*Sep 10 11:58:51.283:
//-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4780):[lvl=0]tmp_ptr:(recvonly)
*Sep 10 11:58:51.283:
//-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4816):[lvl=0]tmp_ptr:(recvonly)
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4822):[lvl=0]match recvonly
 recvonly
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4830):[lvl=0]case
MODE_RECVONLY
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_conn_mode(4894):[lvl=0]SUCCESS:
Connection Mode parsing is OK
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_string_parse(186):[lvl=0]return code=1.
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_parameter_lines(725):[lvl=1]return
 parse function in mgcp_parm_rules_array[1].
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_call_id(840):[lvl=0]in_ptr: 1
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_call_id(883):[lvl=1]SUCCESS: Call
ID string(1) parsing is OK
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_mandatory_parms(12428):[lvl=0]Entered
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_comp_mp_parms(14923):[lvl=0]Entered
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_val_comp_mp_parms(14928):[lvl=1] -
lcon_opt_ptr could not be obtained
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_parse_packet(378):[lvl=2]SUCCESS: END of
 Parsing
*Sep 10 11:58:51.283:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_validate_version_with_call_agent_serv_type(8322):[lvl=1]req_msg_version:
 5, config_version: 5
*Sep 10 11:58:51.283: //-1/xxxxxxxxxxxx/MGCP/mgcp_validate_net_type(6601):[lvl=1]
lcnw_valid=0, lc_con_valid=0
*Sep 10 11:58:51.287: //-1/xxxxxxxxxxxx/MGCP/mgcp_validate_net_type(6710):[lvl=1]Network
type/conection type valid = 1. connection type = 1 [1->RTP, 2->AAL1_SDT, 4->AAL2]
*Sep 10 11:58:51.287: //-1/xxxxxxxxxxxx/MGCP/mgcp_get_qos(2665):[lvl=1]MGCP msg qos value=0
*Sep 10 11:58:51.287: //-1/xxxxxxxxxxxx/MGCP/mgcp_init_dyn_payload_types(2899):[lvl=1]used
 payload type map = 2F400003
*Sep 10 11:58:51.287: //-1/xxxxxxxxxxxx/MGCP/get_voip_peer_info(7155):[lvl=1]No SDP connection
 info
*Sep 10 11:58:51.287:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_select_codec_only(897):[lvl=1]num
supprt codec=14
*Sep 10 11:58:51.287:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_select_codec_only(1061):[lvl=0]glob
 codec=1 (syn=1)
*Sep 10 11:58:51.287:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_select_codec_only(1063):[lvl=0]supp
 list=
*Sep 10 11:58:51.287:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_select_codec_only(1067):[lvl=0] 1
*Sep 10 11:58:51.287:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_select_codec_only(1067):[lvl=0],2
*Sep 10 11:58:51.287:
//-1/95915C328011/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_select_codec_only(1067):[lvl=0],7
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp src

To enable debug traces for the System Resource Check (SRC) Call Admission Control (CAC) process for Media Gateway Control Protocol (MGCP), use the **debug mgcp src** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp src [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp src

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP SRC debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XB	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp src** command:

```
Router# debug mgcp src
Media Gateway Control Protocol System Resource Check CAC debugging for all endpoints is on,
 trace-level Verbose
Router#
*Sep 10 12:01:14.403:
//-1/EADF209C8013/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_set_call_counter_control(8163):[lvl=1]Outgoing
 call with 1 network leg, flag=TRUE
*Sep 10 12:03:01.051:
//35/EADF209C8013/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_reset_call_direction(8184):[lvl=1]Reseting
 incoming_call flag=FALSE in voice_if
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp state

To enable state traces for Media Gateway Control Protocol (MGCP), use the **debug mgcp state**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp state [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp state

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP state debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XB	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp state**command:

```
Router# debug mgcp state
Media Gateway Control Protocol state transition debugging for all endpoints is on, trace-level
 Verbose
Router#
*Sep 10 12:08:02.755:
//39/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E313E0)
 old state=CALL_IDLE new state=CALL_CONNECTING
*Sep 10 12:08:02.755:
//40/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E311D0)
 old state=CALL_IDLE new state=CALL_CONNECTING
*Sep 10 12:08:02.755:
//39/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E313E0)
 old state=CALL_CONNECTING new state=CALL_CONNECTING
*Sep 10 12:08:02.759:
//40/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E311D0)
 old state=CALL_CONNECTING new state=CALL_CONFERENCING
```

```
*Sep 10 12:08:02.759:
//39/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E313E0)
 old state=CALL_CONNECTING new state=CALL_CONFERENCING
*Sep 10 12:08:02.759:
//40/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E311D0)
 old state=CALL_CONFERENCING new state=CALL_CONFERENCING
*Sep 10 12:08:02.763:
//39/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E313E0)
 old state=CALL_CONFERENCING new state=CALL_ACTIVE
*Sep 10 12:08:02.763:
//40/DE454D0E8015/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_set_call_state(7562):[lvl=2]callp(0x63E311D0)
 old state=CALL_CONFERENCING new state=CALL_ACTIVE
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp tracelevel-default

To set the trace level globally for all Media Gateway Control Protocol (MGCP) debug traces, use the **debug mgcp tracelevel-default** command in privileged EXEC mode. To reset the trace level to the default value, use the **no** form of this command.

debug mgcp tracelevel-default {**critical** | **moderate** | **verbose**}
no debug mgcp tracelevel-default {**critical** | **moderate** | **verbose**}

Syntax Description

critical	Only high priority debug information is displayed.
moderate	Medium and high priority debug information is displayed.
verbose	All debug information is displayed. This is the default value.

Command Default

The default trace level for all MGCP debug commands is verbose.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(4)T	This command was introduced.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

This command sets the trace level globally for all MGCP debug commands and endpoints. You can set the trace level independently for a specific endpoint by using the **debug mgcp endpoint** command. The endpoint-specific trace level takes precedence over the global trace-level set with this command.

Note This command applies only to MGCP debug commands that are issued after the default trace level is set. For example, if you enable several debug commands and then change the default trace level, the new trace level does not apply to any previously enabled MGCP debug commands.

Examples

The following example sets the default trace level to critical for all MGCP debug traces:

```
Router# debug mgcp tracelevel-default critical
Router# debug mgcp events

Media Gateway Control Protocol events debugging for all endpoints is on, trace-level Critical
Router# debug mgcp state

Media Gateway Control Protocol state transition debugging for all endpoints is on, trace-level
 Critical
```

Notice that if the default trace level is then changed, as in the following example, the new trace level applies only to any MGCP debug commands that are issued after the default trace level is changed.

```
Router# debug mgcp tracelevel-default verbose

Router# debug mgcp voipcac

Media Gateway Control Protocol VoIPCAC debugging for all endpoints is on, trace-level Verbose
Router# show debug
MGCP:
  Media Gateway Control Protocol events debugging is on, trace level Critical
  Media Gateway Control Protocol VoIPCAC debugging is on, trace level Verbose
  Media Gateway Control Protocol state transition debugging is on, trace level Critical
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp endpoint	Enables debug traces for a specific MGCP endpoint.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mgcp voipcac

To enable debug traces for the Voice over IP (VoIP) Call Admission Control (CAC) process at the Media Gateway Control Protocol (MGCP) application layer, use the **debug mgcp voipcac** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mgcp voipcac [**tracelevel** {**critical** | **moderate** | **verbose**}]
no debug mgcp voipcac

Syntax Description

tracelevel	(Optional) Sets the priority level for this debug trace. • **critical** --Displays only high priority debug information. • **moderate** --Displays medium and high priority debug information. • **verbose** --Displays all debug information. This is the default level.

Command Default

MGCP VoIP CAC debugging is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(2)XB	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.2(11)T	The command was implemented on the Cisco AS5350, Cisco AS5400, and Cisco AS5850.
12.2(13)T	Support for this command was implemented in Cisco 7200 series images.
12.4(4)T	The **tracelevel** keyword was added.

Usage Guidelines

Trace levels allow you to control the amount of debug information that is displayed in the output from MGCP debug commands. Reducing the amount of output displayed on the console port makes it easier to locate the correct debug information and limits the impact to network performance.

Examples

The following is sample output from the **debug mgcp voipcac**command:

```
Router# debug mgcp voipcac
Media Gateway Control Protocol VoIPCAC debugging for all endpoints is on, trace-level Verbose
Router#
*Sep 10 12:04:47.747:
//-1/6A09713E8014/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_idle_crcx(5251):[lvl=0]Check for
HP and QOS combination
*Sep 10 12:04:47.751:
//-1/6A09713E8014/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/mgcp_idle_crcx(5451):[lvl=0]CAC success
*Sep 10 12:04:47.751:
//-1/6A09713E8014/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/set_up_voip_call_leg(3918):[lvl=0]get
```

```
voice interface
*Sep 10 12:04:47.751:
//-1/6A09713E8014/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/set_up_voip_call_leg(4441):[lvl=0]Initialize
 VoIP CAC record stored in
                     VoIP interface struct
*Sep 10 12:04:47.751:
//38/6A09713E8014/MGCP|aaln/S2/SU1/1|-1|-1/<VOIP>/mgcp_connect_peer_vox_call_leg(1546):[lvl=0]set_up_voip_call_leg
 returns OK
*Sep 10 12:04:47.759:
//37/6A09713E8014/MGCP|aaln/S2/SU1/1|-1|-1/<VOICE>/process_signal_request_list(5608):[lvl=0]Entered
```

Related Commands

Command	Description
debug mgcp all	Enables all debug traces for MGCP.
debug mgcp tracelevel-default	Sets the trace level globally for all MGCP debug traces.
mgcp	Starts the MGCP daemon.
mgcp debug-header	Enables the display of MGCP module-dependent information in the debug header.
show mgcp	Displays MGCP configuration information.
voice call debug	Specifies the format of the debug header.

debug mlrib common

To enable logging of common Multilayer Routing Information Base (MLRIB) debug messages, use the **debug mlrib common** command in privileged EXEC mode. To disable the debugging, use the **no** form of this command.

debug mlrib common {**event** {**all** | **client_api** | **db** | **ha** | **misc** | **notify** | **registration** | **show** | **srw** | **utils**} | **error**}
no debug mlrib common {**event** {**all** | **client_api** | **db** | **ha** | **misc** | **notify** | **registration** | **show** | **srw** | **utils**} | **error**}

Syntax Description

event	Enables logging of event debug messages.
all	Enables logging of all common debug events.
client_api	Enables client API-related debugging.
db	Enables MLRIB database debugging.
ha	Enables MLRIB high availability (HA) debugging.
misc	Enables miscellaneous events debugging.
notify	Enables MLRIB notify debugging.
registration	Enables MLRIB registration-related debugging.
show	Enables debugging of MLRIB **show** commands.
srw	Enables MLRIB Single Reader Writer (SRW) debugging.
utils	Enables MLRIB utilities debugging.
error	Enables error debugging.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.5S	This command was introduced.

Examples

The following example shows how to enable debugging of common MLRIB events:

```
Router# debug mlrib common event all

MLRIB COMMON event all debugging is on
*Oct 28 07:26:54.614: MLRIB_COMMON_REGISTRATION: client state set: ISISL2 OTV Overlay2
moving to REGISTERED state
*Oct 28 07:26:54.614: MLRIB_COMMON_REGISTRATION: client state set: ISISL2 OTV Overlay2
moving to REGISTERED state
```

Related Commands

Command	Description
show OTV	Displays information about OTV.

debug mlrib layer2

To enable logging of Layer 2-specific Multilayer Routing Information Base (MLRIB) debug messages, use the **debug mlrib layer2** command in privileged EXEC mode. To disable the debugging, use the **no** form of this command.

debug mlrib layer2 {**event** {**add** | **all** | **delete** | **flush** | **notify** | **redistribute** | **registration**} | **error**}
no debug mlrib layer2 {**event** {**add** | **all** | **delete** | **flush** | **notify** | **redistribute** | **registration**} | **error**}

Syntax Description

event	Enables logging of Layer 2 event debug messages.
add	Enables logging of Layer 2 add MLRIB debug events.
all	Enables logging of all Layer 2 MLRIB debug events.
delete	Enables logging of Layer 2 delete MLRIB debug events.
flush	Enables logging of Layer 2 flush MLRIB debug events.
notify	Enables logging of Layer 2 notify MLRIB debug events.
redistribute	Enables logging of Layer 2 redistribution MLRIB debug events.
registration	Enables logging of Layer 2 registration MLRIB debug events.
error	Enables logging of Layer 2 error debug messages.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.5S	This command was introduced.

Examples

The following example shows how to enable debugging of Layer 2 MLRIB events:

```
Router# debug mlrib layer2 event all

MLRIB L2 event all debugging is on
*Oct 28 07:25:23.257: MLRIB_L2_FLUSH: u flush req msg: flush notifications sent for pp=0x8,
 topo=10
*Oct 28 07:25:23.257: MLRIB_L2_FLUSH: u flush req msg: complete for pp=0x8, topo=12, client
 ISISL2 OTV Overlay1
*Oct 28 07:25:23.257: MLRIB_L2_FLUSH: u flush req msg: flush notifications sent for pp=0x8,
 topo=12
*Oct 28 07:25:23.257: MLRIB_L2_REDISTRIBUTE: hndl ucast redist refresh msg: Rcvd msg length
 20, redist id = 0x0 walk id 1102745848client = ISISL2 OTV Overlay1
*Oct 28 07:25:23.257: MLRIB_L2_REDISTRIBUTE: hndl ucast redist refresh msg: found filter
for redist id = 0x0
*Oct 28 07:25:23.257: MLRIB_L2_REDISTRIBUTE: redist walk setup: for vpn 0x1 and client
ISISL2 OTV Overlay1
```

Related Commands

Command	Description
show OTV	Displays information about OTV.

debug mls rp

To display various Internetwork Packet Exchange (IPX) Multilayer Switching (MLS) debugging elements, use the **debug mls rp**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mls rp {**error** | **events** | **ipx** | **locator** | **packets** | **all**}
no debug mls rp {**error** | **events** | **ipx** | **locator** | **packets** | **all**}

Syntax Description

error	Displays MLS error messages.
events	Displays a run-time sequence of events for the Multilayer Switching Protocol (MLSP).
ipx	Displays IPX-related events for MLS, including route purging and changes to access lists and flow masks.
locator	Identifies which switch is switching a particular flow of MLS explorer packets.
packets	Displays packet contents (in verbose and hexadecimal formats) for MLSP messages.
all	Displays all MLS debugging events.

Command Default

Debugging is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following shows sample output from the **debug mls rp ipx** command:

```
Router# debug mls rp ipx
IPX MLS debugging is on
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# int vlan 22
Router(config-if)# no ipx access-group out
05:44:37:FCP:flowmask changed to destination
```

Related Commands

Command	Description
debug dss ipx event	Displays debugging messages for route change events that affect IPX MLS.

debug mls rp ip multicast

To display information about Multilayer Switching Protocol (MLSP), use the **debug mls rp ip multicast**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mls rp ip multicast {**all** | **error** | **events** | **packets**}
no debub mls rp ip multicast {**all** | **error** | **events** | **packets**}

Syntax Description

all	Displays all multicast MLSP debugging information, including errors, events, and packets.
error	Displays error messages related to multicast MLSP.
events	Displays the run-time sequence of events for multicast MLSP.
packets	Displays the contents of MLSP packets.

Command Default

Debugging is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Only one of the keywords is required.

Examples

The following example shows output from the **debug mls rp ip multicast**command using the **error** keyword:

```
Router# debug mls rp ip multicast error
mlsm error debugging is on
chtang-7200#
06:06:45: MLSMERR: scb is INACTIVE, free INSTALL_FE
06:06:46: MLSM: --> mlsm_proc_sc_ins_req(10.0.0.1, 224.2.2.3, 10)
```

The following example shows output from the **debug mls rp ip multicast**command using the **event** keyword:

```
Router# debug mls rp ip multicast event
mlsm events debugging is on
Router#
3d23h: MSCP: incoming shortcut flow statistic  from Fa2/0.11
3d23h: MLSM: Flow_stat: (192.1.10.6, 239.255.158.197), byte :537792
packet:8403
3d23h: MLSM: byte delta:7680 packet delta:120, time delta: 10
3d23h: MSCP: incoming shortcut flow statistic  from Fa2/0.11
3d23h: MLSM: Flow_stat: (192.1.10.6, 239.255.158.197), byte :545472
packet:8523
3d23h: MLSM: byte delta:7680 packet delta:120, time delta: 10
3d23h: MSCP: Router transmits keepalive_msg on Fa2/0.11
```

```
3d23h: MSCP: incoming shortcut keepalive ACK  from Fa2/0.11
3d23h: MLSM: Include-list: (192.1.2.1  ->  0.0.0.0)
3d23h: MSCP: incoming shortcut flow statistic  from Fa2/0.11
3d23h: MLSM: Flow_stat: (192.1.10.6, 239.255.158.197), byte :553152
packet:8643
```

The following example shows output from the **debug mls rp ip multicast**command using the **packet** keyword:

```
Router# debug mls rp ip multicast packet
mlsm packets debugging is on
Router#
Router#
Router#
Router#
**23h: MSCP(I): 01 00 0c cc cc cc 00 e0 1e 7c fe 5f 00 30 aa aa
...LLL.`.|~_.0
..23h: MSCP(I): 03 00 00 0c 01 07 01 05 00 28 01 02 0a c7 00 10
.........(...G
 ..23h: MSCP(I): a6 0b b4 ff 00 00 c0 01 0a 06 ef ff 9e c5 00 00
&.4...@...o..E
3d23h: MSCP(I): 00 00 00 09 42 c0 00 00 00 00 00 00 25 0b
....B@......%.
3d23h:
**23h: MSCP(O): 01 00 0c 00 00 00 aa 00 04 00 01 04 00 00 aa aa
......*.......
LL23h: MSCP(O): 03 00 00 0c 00 16 00 00 00 00 01 00 0c cc cc cc
.............L
..23h: MSCP(O): aa 00 04 00 01 04 00 24 aa aa 03 00 00 0c 01 07
*......$**....
..23h: MSCP(O): 01 06 00 1c c0 01 02 01 aa 00 04 00 01 04 00 00
....@...*.....
3d23h: MSCP(O): 00 0b 00 00 00 00 00 00 01 01 0a 62 ...........b
3d23h:
**23h: MSCP(I): 01 00 0c cc cc cc 00 e0 1e 7c fe 5f 00 24 aa aa
...LLL.`.|~_.$
..23h: MSCP(I): 03 00 00 0c 01 07 01 86 00 1c 01 02 0a c7 00 10
.............G
..23h: MSCP(I): a6 0b b4 ff 00 00 00 0b 00 00 c0 01 02 01 00 00
..4.......@...
3d23h: MSCP(I): 00 00
3d23h:
```

Related Commands

Command	Description
debug mdss	Displays information about MDSS.

debug mmoip aaa

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

To display output that relates to authentication, authorization, and accounting (AAA) services with store-and-forward fax, use the **debug mmoip aaa** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mmoip aaa
no debug mmoip aaa

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.
12.3(8)T	This command was replaced by the **debug fax mmoip aaa** command.

Examples

The following output shows how the **debug mmoip aaa** command provides information about AAA for the on-ramp or off-ramp gateways:

```
Router# debug mmoip aaa
5d10h:fax_aaa_begin_authentication:User-Name = mmoip-b.cisco.com
5d10h:fax_aaa_begin_authentication:fax_account_id_origin = GATEWAY_ID
5d10h:fax_aaa_end_authentication_callback:Authentication successful
```

The following output shows how the **debug mmoip aaa** command provides information about AAA for the off-ramp gateway:

```
Router# debug mmoip aaa
5d10h:fax_aaa_start_accounting:User-Name = mmoip-b.cisco.com
5d10h:fax_aaa_start_accounting:Calling-Station-Id = gmercuri@mail-server.cisco.com
5d10h:fax_aaa_start_accounting:Called-Station-Id = fax=571-0839@mmoip-b.cisco.com
5d10h:fax_aaa_start_accounting:fax_account_id_origin = GATEWAY_ID
mmoip-b#ax_aaa_start_accounting:fax_msg_id = <37117AF3.3D98300E@mail-server.cisco.com>
5d10h:fax_aaa_start_accounting:fax_pages = 2
5d10h:fax_aaa_start_accounting:fax_coverpage_flag = TRUE
5d10h:fax_aaa_start_accounting:fax_connect_speed = 14400bps
5d10h:fax_aaa_start_accounting:fax_recipient_count = 1
5d10h:fax_aaa_start_accounting:fax_auth_status = USER SUCCESS
5d10h:fax_aaa_start_accounting:gateway_id = mmoip-b.cisco.com
5d10h:fax_aaa_start_accounting:call_type = Fax Send
```

```
5d10h:fax_aaa_start_accounting:port_used =  slot:0 vfc port:0
5d10h:fax_aaa_do_offramp_accounting tty(6), Stopping accounting
5d10h:fax_aaa_stop_accounting:ftdb->cact->generic.callActiveTransmitBytes = 18038
5d10h:fax_aaa_stop_accounting:ftdb->cact->generic.callActiveTransmitPackets = 14
```

The following output shows how the **debug mmoip aaa** command provides information about AAA for the on-ramp gateway:

```
Router# debug mmoip aaa
5d10h:fax_aaa_start_accounting:User-Name = mmoip-b.cisco.com
5d10h:fax_aaa_start_accounting:Calling-Station-Id = FAX=408@mail-from-hostname.com
5d10h:fax_aaa_start_accounting:Called-Station-Id = FAX=5710839@mail-server.cisco.com
5d10h:fax_aaa_start_accounting:fax_account_id_origin = GATEWAY_ID
5d10h:fax_aaa_start_accounting:fax_msg_id = 00391997233216263@mmoip-b.cisco.com
5d10h:fax_aaa_start_accounting:fax_pages = 2
5d10h:fax_aaa_start_accounting:fax_connect_speed = 14400bps
5d10h:fax_aaa_start_accounting:fax_auth_status = USER SUCCESS
5d10h:fax_aaa_start_accounting:email_server_address = 1.14.116.1
5d10h:fax_aaa_start_accounting:email_server_ack_flag = TRUE
5d10h:fax_aaa_start_accounting:gateway_id = mmoip-b.cisco.com
5d10h:fax_aaa_start_accounting:call_type = Fax Receive
5d10h:fax_aaa_start_accounting:port_used = Cisco Powered Fax System slot:1 port:4
5d10h:fax_aaa_do_onramp_accounting tty(5), Stopping accounting
5d10h:fax_aaa_stop_accounting:endb->cact->generic.callActiveTransmitBytes = 26687
5d10h:fax_aaa_stop_accounting:ftdb->cact->generic.callActiveReceiveBytes = 18558
5d10h:fax_aaa_stop_accounting:ftdb->cact->generic.callActiveReceivePackets = 14
```

debug mmoip send email

To test connectivity between the T.37 on-ramp gateway and the e-mail server by sending a test e-mail to a specified e-mail address, use the **debug mmoip send email** command in privileged EXEC mode.

debug mmoip send email *string*

Syntax Description

string	E-mail address of the sender; for example, mailuser@mail-server.com. There is no default.

Command Default

This command is not enabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(4)T	This command was introduced.
12.2(4)T	This command was introduced 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.

Examples

The **debug mmoip send email** command is used to test connectivity between the on-ramp gateway and the e-mail server. Basically, this **debug** command sends an e-mail message to the recipient specified in the e-mail address string. There is no specific output associated with the **debug mmoip send email** command; to see how the on-ramp gateway and e-mail server interact when processing the test e-mail message, enable the **debug fmail client** command.

The following example tests connectivity between the on-ramp gateway and the e-mail server by sending a test e-mail message to mailuser@mail-server.com:

```
Router#
debug fmail client

Router#
debug mmoip send email mailuser@mail-server.com
01:22:59:faxmail_client_send_test:Sending the test message to
ilya@mail-server.com from testing@mmoip-a.cisco.com...
01:22:59:faxmail_client_send_test:Opening client engine.
01:22:59:faxmail_client_send_test:Sending 59 bytes ...
01:22:59:faxmail_client_send_test:Done sending test email.
```

Related Commands

Command	Description
debug fmail client	Displays e-mail parameters (such as Mail from and Envelope to and Envelope from) and the progress of the SMTP client.

debug mmoip send fax

To send a T.37 off-ramp test fax, use the **debug mmoip send fax** command in privileged EXEC mode.

debug mmoip send fax *string*

Syntax Description

string	E.164 telephone number to be used for sending the test fax. There is no default.

Command Default

This command is not enabled.

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 introduced on the Cisco 7200 series routers.

Examples

The **debug mmoip send fax** command is used to test connectivity between the off-ramp gateway and a recipient fax device. Basically, this **debug** command sends a test fax transmission to the recipient specified in the telephone number string. There is no specific output associated with the **debug mmoip send fax** command.

The following example sends a test fax message to the telephone number 5550839:

```
Router# debug mmoip send fax 5550839
```

The following output shows that the off-ramp gateway is placing a fax call:

```
01:28:18:ftsp_offramp_match_digits:phone number to translate:5550839
01:28:18: destPat(5......), matched(1), prefix() peer_tag(1)
01:28:18:ftsp_offramp_match_digits:target:710839
01:28:18:fap_offcm:tty(4), Got dial message00:00:00.000:AT&F\Q0S7=255
```

Class 2 modem tracing begins, including modem initialization.

```
00:00:00.008:AA
00:00:00.068:TT
00:00:00.128:&F\Q0S7=255
00:00:00.128:
OK

00:00:00.128:E0V1
00:00:00.140:ATE0
OK

00:00:00.140:AT+FCLASS=2
```

```
00:00:00.148:
OK

00:00:00.148:+FDCC=..;+FBOR=
00:00:00.168:AT+FLID
00:00:00.180:
OK

00:00:00.180:ATDTW710839
The following output shows that the fax transmission is complete; in this particular example,
 there was a transmission error, and the modem timed out.
01:28:25:ftsp_setup_for_oc:tty4, callid=0xA
01:28:25:ftsp_setup_for_oc ctl=0, cas grp=-1, snmp_ix=30
01:28:25:ftsp_off_ramp_active_call_init tty4 callid=0xA, snmp_ix=30
01:29:18:fap_offpmt:tty(4), TxPhaseA:modem timeout
01:29:18:%FTSP-6-FAX_DISCONNECT:Transmission er
```

debug mmoip transfer

To send output of the Tag Image File Format (TIFF) writer to a TFTP server, use the **debug mmoip transfer** command in privileged EXEC mode.

debug mmoip transfer *prefix-filename tftp-server-name*

Syntax Description

prefix-filename	Name of the TIFF file. The format for the TIFF filename is "telephone-number.TIFF."
tftp-server-name	TFTP server to which the output from the TIFF writer is sent.

Command Default

Sending output of the TIFF writer to a TFTP server is disabled.

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.

Examples

The **debug mmoip transfer command**sends the content of the fax data received to the TFTP server named by the *tftp-server-name* variable into the file identified by the *prefix-filename* variable. Each page of the fax transmission is a separate file, designated by the letter "p", followed by the page number.

For example, the following command transfers the received fax content to a TFTP server named "keyer". The first page of the transmission goes to the file named "/tftpboot/test/testp1.tiff", the second page goes to the file named "/tftpboot/test/testp2.tiff" and so on.

```
Router# debug mmoip transfer /tftpboot/test/test keyer
```

The named files must exist on the TFTP server and be writable in order for the debug operation to be successful.

debug modem

To observe modem line activity on an access server, use the **debug modem** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem
no debug modem

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Examples

The following is sample output from the **debug modem** command. The output shows when the modem line changes state.

```
Router# debug modem
15:25:51: TTY4: DSR came up
15:25:51: tty4: Modem: IDLE->READY
15:25:51: TTY4: Autoselect started
15:27:51: TTY4: Autoselect failed
15:27:51: TTY4: Line reset
15:27:51: TTY4: Modem: READY->HANGUP
15:27:52: TTY4: dropping DTR, hanging up
15:27:52: tty4: Modem: HANGUP->IDLE
15:27:57: TTY4: restoring DTR
15:27:58: TTY4: DSR came up
```

debug modem csm

To debug the Call Switching Module (CSM), used to connect calls on the modem, use the **debug modem csm**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem csm [{*slot*/*port* | **group** *group-number*}]
no debug modem csm [{*slot*/*port* | **group** *group-number*}]

Syntax Description

slot/port	(Optional) The slot and modem port number.
group *group-number*	(Optional) The modem group.

Command Modes

Privileged EXEC

Usage Guidelines

Use the **debug modem csm** command to troubleshoot call switching problems. With this command, you can trace the complete sequence of switching incoming and outgoing calls.

Examples

The following is sample output from the **debug modem csm** command. In this example, a call enters the modem (incoming) on slot 1, port 0:

```
Router(config)# service timestamps debug uptime
Router(config)# end
Router# debug modem csm
00:04:09: ccpri_ratetoteup bear rate is 10
00:04:09: CSM_MODEM_ALLOCATE: slot 1 and port 0 is allocated.
00:04:09: MODEM_REPORT(0001): DEV_INCALL at slot 1 and port 0
00:04:09: CSM_PROC_IDLE: CSM_EVENT_ISDN_CALL at slot 1, port 0
00:04:11: CSM_RING_INDICATION_PROC: RI is on
00:04:13: CSM_RING_INDICATION_PROC: RI is off
00:04:15: CSM_PROC_IC1_RING: CSM_EVENT_MODEM_OFFHOOK at slot 1, port 0
00:04:15: MODEM_REPORT(0001): DEV_CONNECTED at slot 1 and port 0
00:04:15: CSM_PROC_IC2_WAIT_FOR_CARRIER: CSM_EVENT_ISDN_CONNECTED at slot 1, port 0
```

The following is sample output from the **debug modem csm**command when call is dialed from the modem into the network (outgoing) from slot 1, port 2:

```
Router# debug modem csm
atdt16665202
00:11:21: CSM_PROC_IDLE: CSM_EVENT_MODEM_OFFHOOK at slot 1, port 2
00:11:21: T1_MAIL_FROM_NEAT: DC_READY_RSP: mid = 1, slot = 0, unit = 0
00:11:21: CSM_PROC_OC1_REQUEST_DIGIT: CSM_EVENT_DIGIT_COLLECT_READY at slot 1, port 2
00:11:24: T1_MAIL_FROM_NEAT: DC_FIRST_DIGIT_RSP: mid = 1, slot = 0, unit = 0
00:11:24: CSM_PROC_OC2_COLLECT_1ST_DIGIT: CSM_EVENT_GET_1ST_DIGIT at slot 1, port 2
00:11:27: T1_MAIL_FROM_NEAT: DC_ALL_DIGIT_RSP: mid = 1, slot = 0, unit = 0
00:11:27: CSM_PROC_OC3_COLLECT_ALL_DIGIT: CSM_EVENT_GET_ALL_DIGITS (16665202) at slot 1,
port 2
00:11:27: ccpri_ratetoteup bear rate is 10
00:11:27: MODEM_REPORT(A000): DEV_CALL_PROC at slot 1 and port 2
00:11:27: CSM_PROC_OC4_DIALING: CSM_EVENT_ISDN_BCHAN_ASSIGNED at slot 1, port 2
00:11:31: MODEM_REPORT(A000): DEV_CONNECTED at slot 1 and port 2
00:11:31: CSM_PROC_OC5_WAIT_FOR_CARRIER: CSM_EVENT_ISDN_CONNECTED at slot 1, port 2
CONNECT 19200/REL - MNP
```

The following is sample output from the **debug modem csm** command for an incoming call:

```
Router# debug modem csm
Router#1.19.36.7 2001
Trying 1.19.36.7, 2001 ... Open
atdt111222333444555666
*Apr  7 12:39:42.475: Mica Modem(1/0): Rcvd Dial String(111222333444555666)
*Apr  7 12:39:42.475: CSM_PROC_IDLE: CSM_EVENT_MODEM_OFFHOOK at slot 1, port 0
*Apr  7 12:39:42.479: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_CHANNEL_LOCK at slot 1 and
 port 0
*Apr  7 12:39:42.479: CSM_PROC_OC4_DIALING: CSM_EVENT_DSX0_BCHAN_ASSIGNED at slot 1, port
0
*Apr  7 12:39:42.479: Mica Modem(1/0): Configure(0x1)
*Apr  7 12:39:42.479: Mica Modem(1/0): Configure(0x5)
*Apr  7 12:39:42.479: Mica Modem(1/0): Call Setup
*Apr  7 12:39:42.479: neat msg at slot 0: (1/0): Tx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:42.491: neat msg at slot 0: (0/0): Rx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:42.531: VDEV_ALLOCATE: slot 1 and port 3 is allocated.
*Apr  7 12:39:42.531: CSM_RX_CAS_EVENT_FROM_NEAT:(0004):  EVENT_CALL_DIAL_IN at slot 1 and
 port 3
*Apr  7 12:39:42.531: CSM_PROC_IDLE: CSM_EVENT_DSX0_CALL at slot 1, port 3
*Apr  7 12:39:42.531: Mica Modem(1/3): Configure(0x0)
*Apr  7 12:39:42.531: Mica Modem(1/3): Configure(0x5)
*Apr  7 12:39:42.531: Mica Modem(1/3): Call Setup
*Apr  7 12:39:42.595: Mica Modem(1/0): State Transition to Call Setup
*Apr  7 12:39:42.655: Mica Modem(1/3): State Transition to Call Setup
*Apr  7 12:39:42.655: Mica Modem(1/3): Went offhook
*Apr  7 12:39:42.655: CSM_PROC_IC1_RING: CSM_EVENT_MODEM_OFFHOOK at slot 1, port 3
*Apr  7 12:39:42.671: neat msg at slot 0: (0/0): Tx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:42.691: neat msg at slot 0: (1/0): Rx LOOP_CLOSURE (ABCD=1101)
*Apr 7 12:39:42.731: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_START_TX_TONE at slot 1 and
 port 0
*Apr  7 12:39:42.731: CSM_PROC_OC4_DIALING: CSM_EVENT_DSX0_START_TX_TONE at slot 1, port 0
*Apr  7 12:39:42.731: Mica Modem(1/0): Generate digits:called_party_num= len=1
*Apr  7 12:39:42.835: Mica Modem(1/3): Rcvd Digit detected(#)
*Apr  7 12:39:42.835: CSM_PROC_IC2_COLLECT_ADDR_INFO: CSM_EVENT_KP_DIGIT_COLLECTED (DNIS=,
 ANI=) at slot 1, port 3
*Apr  7 12:39:42.855: neat msg at slot 0: (0/0): Tx LOOP_OPEN (ABCD=0101)
*Apr  7 12:39:42.871: neat msg at slot 0: (1/0): Rx LOOP_OPEN (ABCD=0101)
*Apr  7 12:39:42.899: Mica Modem(1/0): Rcvd Digits Generated
*Apr  7 12:39:42.911: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_END_TX_TONE at slot 1 and
port 0
*Apr  7 12:39:42.911: CSM_PROC_OC4_DIALING: CSM_EVENT_DSX0_END_TX_TONE at slot 1, port 0
*Apr  7 12:39:42.911: Mica Modem(1/0): Generate digits:called_party_num=A len=1
*Apr  7 12:39:43.019: Mica Modem(1/0): Rcvd Digits Generated
*Apr  7 12:39:43.019: CSM_PROC_OC4_DIALING: CSM_EVENT_TONE_GENERATED at slot 1, port 0
*Apr  7 12:39:43.019: Mica Modem(1/3): Rcvd Digit detected(A)
*Apr 7 12:39:43.335: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_START_TX_TONE at slot 1 and
 port 0
*Apr  7 12:39:43.335: CSM_PROC_OC4_DIALING: CSM_EVENT_DSX0_START_TX_TONE at slot 1, port 0
*Apr  7 12:39:43.335: Mica Modem(1/0): Generate digits:called_party_num=111222333444555666
 len=19
*Apr  7 12:39:43.439: Mica Modem(1/3): Rcvd Digit detected(1)
*Apr  7 12:39:43.559: Mica Modem(1/3): Rcvd Digit detected(1)
*Apr  7 12:39:43.619: Mica Modem(1/3): Rcvd Digit detected(1)
*Apr  7 12:39:43.743: Mica Modem(1/3): Rcvd Digit detected(2)
*Apr  7 12:39:43.859: Mica Modem(1/3): Rcvd Digit detected(2)
*Apr  7 12:39:43.919: Mica Modem(1/3): Rcvd Digit detected(2)
*Apr  7 12:39:44.043: Mica Modem(1/3): Rcvd Digit detected(3)
*Apr  7 12:39:44.163: Mica Modem(1/3): Rcvd Digit detected(3)
*Apr  7 12:39:44.223: Mica Modem(1/3): Rcvd Digit detected(3)
*Apr  7 12:39:44.339: Mica Modem(1/3): Rcvd Digit detected(4)
*Apr  7 12:39:44.459: Mica Modem(1/3): Rcvd Digit detected(4)
*Apr  7 12:39:44.523: Mica Modem(1/3): Rcvd Digit detected(4)
*Apr  7 12:39:44.639: Mica Modem(1/3): Rcvd Digit detected(5)
```

```
*Apr  7 12:39:44.763: Mica Modem(1/3): Rcvd Digit detected(5)
*Apr  7 12:39:44.883: Mica Modem(1/3): Rcvd Digit detected(5)
*Apr  7 12:39:44.943: Mica Modem(1/3): Rcvd Digit detected(6)
*Apr  7 12:39:45.063: Mica Modem(1/3): Rcvd Digit detected(6)
*Apr  7 12:39:45.183: Mica Modem(1/3): Rcvd Digit detected(6)
*Apr  7 12:39:45.243: Mica Modem(1/3): Rcvd Digit detected(B)
*Apr  7 12:39:45.243: CSM_PROC_IC2_COLLECT_ADDR_INFO: CSM_EVENT_DNIS_COLLECTED
(DNIS=111222333444555666, ANI=) at slot 1, port 3
*Apr  7 12:39:45.363: Mica Modem(1/0): Rcvd Digits Generated
*Apr  7 12:39:45.891: neat msg at slot 0: (0/0): Tx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:45.907: neat msg at slot 0: (1/0): Rx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:46.115: neat msg at slot 0: (0/0): Tx LOOP_OPEN (ABCD=0101)
*Apr  7 12:39:46.131: neat msg at slot 0: (1/0): Rx LOOP_OPEN (ABCD=0101)
*Apr  7 12:39:46.175: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_START_TX_TONE at slot 1 and
 port 0
*Apr  7 12:39:46.175: CSM_PROC_OC4_DIALING: CSM_EVENT_DSX0_START_TX_TONE at slot 1, port 0
*Apr  7 12:39:46.175: Mica Modem(1/0): Generate digits:called_party_num= len=3
*Apr  7 12:39:46.267: Mica Modem(1/3): Rcvd Digit detected(#)
*Apr  7 12:39:46.387: Mica Modem(1/3): Rcvd Digit detected(A)
*Apr  7 12:39:46.447: Mica Modem(1/3): Rcvd Digit detected(B)
*Apr  7 12:39:46.447: CSM_PROC_IC2_COLLECT_ADDR_INFO: CSM_EVENT_ADDR_INFO_COLLECTED
(DNIS=111222333444555666, ANI=) at slot 1, port 3
*Apr  7 12:39:46.507: Mica Modem(1/0): Rcvd Digits Generated
*Apr  7 12:39:46.507: CSM_PROC_OC4_DIALING: CSM_EVENT_ADDR_INFO_COLLECTED at slot 1, port
0
*Apr  7 12:39:47.127: CSM_RX_CAS_EVENT_FROM_NEAT:(0004):  EVENT_CHANNEL_CONNECTED at slot
1 and port 3
*Apr  7 12:39:47.127: CSM_PROC_IC4_WAIT_FOR_CARRIER: CSM_EVENT_DSX0_CONNECTED at slot 1,
port 3
*Apr  7 12:39:47.127: Mica Modem(1/3): Link Initiate
*Apr  7 12:39:47.131: neat msg at slot 0: (0/0): Tx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:47.147: neat msg at slot 0: (1/0): Rx LOOP_CLOSURE (ABCD=1101)
*Apr  7 12:39:47.191: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_CHANNEL_CONNECTED at slot
1 and port 0
*Apr  7 12:39:47.191: CSM_PROC_OC5_WAIT_FOR_CARRIER: CSM_EVENT_DSX0_CONNECTED at slot 1,
port 0
*Apr  7 12:39:47.191: Mica Modem(1/0): Link Initiate
*Apr  7 12:39:47.227: Mica Modem(1/3): State Transition to Connect
*Apr  7 12:39:47.287: Mica Modem(1/0): State Transition to Connect
*Apr  7 12:39:49.103: Mica Modem(1/0): State Transition to Link
*Apr  7 12:39:52.103: Mica Modem(1/3): State Transition to Link
*Apr  7 12:40:00.927: Mica Modem(1/3): State Transition to Trainup
*Apr  7 12:40:00.991: Mica Modem(1/0): State Transition to Trainup
*Apr  7 12:40:02.615: Mica Modem(1/0): State Transition to EC Negotiating
*Apr  7 12:40:02.615: Mica Modem(1/3): State Transition to EC Negotiating
CONNECT 31200 /V.42/V.42bis
Router>
*Apr  7 12:40:05.983: Mica Modem(1/0): State Transition to Steady State
*Apr  7 12:40:05.983: Mica Modem(1/3): State Transition to Steady State+++
OK
ath
*Apr  7 12:40:09.167: Mica Modem(1/0): State Transition to Steady State Escape
*Apr  7 12:40:10.795: Mica Modem(1/0): State Transition to Terminating
*Apr  7 12:40:10.795: Mica Modem(1/3): State Transition to Terminating
*Apr  7 12:40:11.755: Mica Modem(1/3): State Transition to Idle
*Apr  7 12:40:11.755: Mica Modem(1/3): Went onhook
*Apr  7 12:40:11.755: CSM_PROC_IC5_OC6_CONNECTED: CSM_EVENT_MODEM_ONHOOK at slot 1, port 3
*Apr  7 12:40:11.755: VDEV_DEALLOCATE: slot 1 and port 3 is deallocated
*Apr  7 12:40:11.759: neat msg at slot 0: (0/0): Tx LOOP_OPEN (ABCD=0101)
*Apr 7 12:40:11.767: neat msg at slot 0: (1/0): Rx LOOP_OPEN (ABCD=0101)
*Apr  7 12:40:12.087: neat msg at slot 0: (1/0): Tx LOOP_OPEN (ABCD=0101)
*Apr  7 12:40:12.091: neat msg at slot 0: (0/0): Rx LOOP_OPEN (ABCD=0101)
*Apr  7 12:40:12.111: CSM_RX_CAS_EVENT_FROM_NEAT:(A001):  EVENT_CALL_IDLE at slot 1 and
port 0
```

```
*Apr  7 12:40:12.111: CSM_PROC_IC5_OC6_CONNECTED: CSM_EVENT_DSX0_DISCONNECTED at slot 1,
port 0
*Apr  7 12:40:12.111: Mica Modem(1/0): Link Terminate(0x6)
*Apr  7 12:40:12.779: Mica Modem(1/3): State Transition to Terminating
*Apr  7 12:40:12.839: Mica Modem(1/3): State Transition to Idle
*Apr  7 12:40:13.495: Mica Modem(1/0): State Transition to Idle
*Apr  7 12:40:13.495: Mica Modem(1/0): Went onhook
*Apr  7 12:40:13.495: CSM_PROC_IC6_OC8_DISCONNECTING: CSM_EVENT_MODEM_ONHOOK at slot 1,
port 0
*Apr  7 12:40:13.495: VDEV_DEALLOCATE: slot 1 and port 0 is deallocated
Router#disc
Closing connection to 1.19.36.7 [confirm]
Router#
*Apr  7 12:40:18.783: Mica Modem(1/0): State Transition to Terminating
*Apr  7 12:40:18.843: Mica Modem(1/0): State Transition to Idle
Router#
```

The MICA technologies modem goes through the following internal link states when the call comes in:

- Call Setup
- Off Hook
- Connect
- Link
- Trainup
- EC Negotiation
- Steady State

The following section describes the CSM activity for an incoming call.

When a voice call comes in, CSM is informed of the incoming call. This allocates the modem and sends the Call Setup message to the MICA modem. The Call_Proc message is sent through D channel. The modem sends an offhook message to CSM by sending the state change to Call Setup. The D channel then sends a CONNECT message. When the CONNECT_ACK message is received, the Link initiate message is sent to the MICA modem and it negotiates the connection with the remote modem. In the following debug examples, a modem on slot 1, port 13 is allocated. It goes through its internal states before it is in Steady State and answers the call.

```
Router# debug modem csm
Modem Management Call Switching Module debugging is on
*May 13 15:01:00.609: MODEM_REPORT:dchan_idb=0x60D437F8, call_id=0xE, ces=0x1
   bchan=0x12, event=0x1, cause=0x0
*May 13 15:01:00.609: VDEV_ALLOCATE: slot 1 and port 13 is allocated.
*May 13 15:01:00.609: MODEM_REPORT(000E): DEV_INCALL at slot 1 and port 13
*May 13 15:01:00.609: CSM_PROC_IDLE: CSM_EVENT_ISDN_CALL at slot 1, port 13
*May 13 15:01:00.609: Mica Modem(1/13): Configure(0x0)
*May 13 15:01:00.609: Mica Modem(1/13): Configure(0x0)
*May 13 15:01:00.609: Mica Modem(1/13): Configure(0x6)
*May 13 15:01:00.609: Mica Modem(1/13): Call Setup
*May 13 15:01:00.661: Mica Modem(1/13): State Transition to Call Setup
*May 13 15:01:00.661: Mica Modem(1/13): Went offhook
*May 13 15:01:00.661: CSM_PROC_IC1_RING: CSM_EVENT_MODEM_OFFHOOK at slot 1, port 13
*May 13 15:01:00.661: MODEM_REPORT:dchan_idb=0x60D437F8, call_id=0xE, ces=0x1
   bchan=0x12, event=0x4, cause=0x0
*May 13 15:01:00.661: MODEM_REPORT(000E): DEV_CONNECTED at slot 1 and port 13
```

```
*May 13 15:01:00.665: CSM_PROC_IC3_WAIT_FOR_CARRIER:
CSM_EVENT_ISDN_CONNECTED at slot 1, port 13
*May 13 15:01:00.665: Mica Modem(1/13): Link Initiate
*May 13 15:01:00.693: Mica Modem(1/13): State Transition to Connect
*May 13 15:01:01.109: Mica Modem(1/13): State Transition to Link
*May 13 15:01:09.433: Mica Modem(1/13): State Transition to Trainup
*May 13 15:01:11.541: Mica Modem(1/13): State Transition to EC Negotiating
*May 13 15:01:12.501: Mica Modem(1/13): State Transition to Steady State
```

The following section describes the status of CSM when a call is connected.

The **show modem csm x/y** command is similar to AS5200 access server. For an active incoming analog call, the modem_status and csm_status should be VDEV_STATUS_ACTIVE_CALL and CSM_IC4_CONNECTED, respectively.

```
Router# show modem csm 1/13
MODEM_INFO: slot 1, port 13, unit 0, modem_mask=0x0000, modem_port_offset=0
tty_hwidb=0x60D0BCE0, modem_tty=0x60B6FE7C, oobp_info=0x00000000,
modem_pool=0x60ADC998
modem_status(0x0002):VDEV_STATUS_ACTIVE_CALL.
csm_state(0x0204)=CSM_IC4_CONNECTED, csm_event_proc=0x600C6968, current
call thru PRI line
invalid_event_count=0, wdt_timeout_count=0
wdt_timestamp_started is not activated
wait_for_dialing:False, wait_for_bchan:False
pri_chnl=TDM_PRI_STREAM(s0, u0, c18), modem_chnl=TDM_MODEM_STREAM(s1, c13)
dchan_idb_start_index=0, dchan_idb_index=0, call_id=0x000E, bchan_num=18
csm_event=CSM_EVENT_ISDN_CONNECTED, cause=0x0000
ring_indicator=0, oh_state=0, oh_int_enable=0, modem_reset_reg=0
ring_no_answer=0, ic_failure=0, ic_complete=1
dial_failure=0, oc_failure=0, oc_complete=0
oc_busy=0, oc_no_dial_tone=0, oc_dial_timeout=0
remote_link_disc=0, stat_busyout=0, stat_modem_reset=0
oobp_failure=0
call_duration_started=1d02h, call_duration_ended=00:00:00,
total_call_duration=00:00:00
The calling party phone number = 4085552400
The called party phone number  = 4085551400
total_free_rbs_timeslot = 0, total_busy_rbs_timeslot = 0,
total_dynamic_busy_rbs_timeslot = 0, total_static_busy_rbs_timeslot = 0,
min_free_modem_threshold = 6
```

The following section describes the CSM activity for an outgoing call.

For MICA modems, the dial tone is not required to initiate an outbound call. Unlike in the AS5200, the digit collection step is not required. The dialed digit string is sent to the CSM in the outgoing request to the CSM. CSM signals the D channel to generate an outbound voice call, and the B channel assigned is connected to the modem and the CSM.

The modem is ordered to connect to the remote side with a CONNECT message, and by sending a link initiate message, the modem starts to train.

```
Router# debug modem csm
Modem Management Call Switching Module debugging is on
Router# debug isdn q931
ISDN Q931 packets debugging is on
*May 15 12:48:42.377: Mica Modem(1/0): Rcvd Dial String(5552400)
*May 15 12:48:42.377: CSM_PROC_IDLE: CSM_EVENT_MODEM_OFFHOOK at slot 1, port 0
*May 15 12:48:42.377: CSM_PROC_OC3_COLLECT_ALL_DIGIT:
CSM_EVENT_GET_ALL_DIGITS at slot 1, port 0
*May 15 12:48:42.377: CSM_PROC_OC3_COLLECT_ALL_DIGIT: called party num:
(5552400) at slot 1, port 0
```

```
*May 15 12:48:42.381:  process_pri_call making a voice_call.
*May 15 12:48:42.381: ISDN Se0:23: TX ->  SETUP pd = 8  callref = 0x0011
*May 15 12:48:42.381:         Bearer Capability i = 0x8090A2
*May 15 12:48:42.381:         Channel ID i = 0xE1808397
*May 15 12:48:42.381:         Called Party Number i = 0xA1, '5552400'
*May 15 12:48:42.429: ISDN Se0:23: RX <-  CALL_PROC pd = 8  callref = 0x8011
*May 15 12:48:42.429:         Channel ID i = 0xA98397
*May 15 12:48:42.429: MODEM_REPORT:dchan_idb=0x60D437F8, call_id=0xA011, ces=0x1
   bchan=0x16, event=0x3, cause=0x0
*May 15 12:48:42.429: MODEM_REPORT(A011): DEV_CALL_PROC at slot 1 and port 0
*May 15 12:48:42.429: CSM_PROC_OC4_DIALING: CSM_EVENT_ISDN_BCHAN_ASSIGNED
at slot 1, port 0
*May 15 12:48:42.429: Mica Modem(1/0): Configure(0x1)
*May 15 12:48:42.429: Mica Modem(1/0): Configure(0x0)
*May 15 12:48:42.429: Mica Modem(1/0): Configure(0x6)
*May 15 12:48:42.429: Mica Modem(1/0): Call Setup
*May 15 12:48:42.489: Mica Modem(1/0): State Transition to Call Setup
*May 15 12:48:42.589: ISDN Se0:23: RX <-  ALERTING pd = 8  callref = 0x8011
*May 15 12:48:43.337: ISDN Se0:23: RX <-  CONNECT pd = 8  callref = 0x8011
*May 15 12:48:43.341: MODEM_REPORT:dchan_idb=0x60D437F8, call_id=0xA011, ces=0x1
   bchan=0x16, event=0x4, cause=0x0
*May 15 12:48:43.341: MODEM_REPORT(A011): DEV_CONNECTED at slot 1 and port 0
*May 15 12:48:43.341: CSM_PROC_OC5_WAIT_FOR_CARRIER:
CSM_EVENT_ISDN_CONNECTED at slot 1, port 0
*May 15 12:48:43.341: Mica Modem(1/0): Link Initiate
*May 15 12:48:43.341: ISDN Se0:23: TX ->  CONNECT_ACK pd = 8  callref = 0x0011
*May 15 12:48:43.385: Mica Modem(1/0): State Transition to Connect
*May 15 12:48:43.849: Mica Modem(1/0): State Transition to Link
*May 15 12:48:52.665: Mica Modem(1/0): State Transition to Trainup
*May 15 12:48:54.661: Mica Modem(1/0): State Transition to EC Negotiating
*May 15 12:48:54.917: Mica Modem(1/0): State Transition to Steady State
```

Related Commands

Command	Description
debug modem oob	Creates modem startup messages between the network management software and the modem on the specified OOB port.
debug modem trace	Performs a call trace on the specified modem, which allows you to determine why calls are terminated.

debug modem dsip

To display output for modem control messages that are received or sent to the router, use the **debugmodemdsip** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem dsip {*tty-range* | **group** | *shelf/ slot/ port*}
no debug modem dsip {*tty-range* | **group** | *shelf/ slot/ port*}

Syntax Description

tty-range	Modem tty number or range. You can specify a single TTY line number or a range from 0 through the number of modems you have in your Cisco AS5800 access server. Be sure to include a dash (-) between the range values you specify.
group	Modem group information.
shelf/slot/port	Location of the modem by shelf/slot/port numbers for internal modems.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.3(2)AA	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The**debugmodemdsip**command displays each Distributed System Interconnect Protocol (DSIP) message that relates to a modem and is sent from or received at the router shelf. This command can be applied to a single modem or a group of modems.

Examples

The following examples show a display of the available **debugmodem** command options and **debugmodemdsip** command options:

```
Router# debug modem ?
  dsip         Modem DSIP activity
  maintenance  Modem maintenance activity
  oob          Modem out of band activity
  trace        Call Trace Upload
  traffic      Modem data traffic
  <cr>

Router# debug modem dsip ?
  <0-935>  First Modem TTY Number
  group    Modem group information
  x/y/z    Shelf/Slot/Port for Internal Modems
  <cr>
```

The following example indicates that a Real Time Server (RTS) status message was received from the router shelf, and an ACK message was sent back:

```
Router# debug modem dsip
00:11:02: RSMODEM_SEND-1/2/06: MODEM_RING_INDICATION_MSG cci1 si0 ms0 mm65535,0 dc0
00:11:02: RSMODEM_sRCV-1/2/06:l12,MODEM_CALL_ACK_MSG:
```

```
00:11:02: RSMODEM_SEND-1/2/06: MODEM_CALL_ACCEPT_MSG
00:11:11: RSMODEM_sRCV-2:l0,MODEM_POLL_MSG: 0 16 0 7 0 146 0 36 21
00:11:18: RSMODEM_sRCV-1/2/06:l12,MODEM_SET_DCD_STATE_MSG: 1
00:11:19: RSMODEM_SEND-1/2/06: MODEM_RTS_STATUS_MSG 1
00:11:19: RSMODEM_dRCV-2:l1258607996,MODEM_RTS_STATUS_MSG: 0 6 0 23 0 0 0 0 0
00:11:23: RSMODEM_sRCV-2:l0,MODEM_POLL_MSG: 0 16 0 7 0 146 0 150 21
00:12:31: RSMODEM_sRCV-1/2/06:l12,MODEM_SET_DCD_STATE_MSG: 0
00:12:31: RSMODEM_SEND-1/2/06: MODEM_CALL_HANGUP_MSG
00:12:31: RSMODEM_sRCV-1/2/06:l12,MODEM_ONHOOK_MSG:
00:12:32: RSMODEM_SEND-1/2/06: MODEM_RTS_STATUS_MSG 1
00:12:32: RSMODEM_SEND-1/2/06: MODEM_SET_DTR_STATE_MSG 0
00:12:32: RSMODEM_dRCV-2:l1258659676,MODEM_RTS_STATUS_MSG: 0 6 0 16 0 0 0 0 0
00:12:32: RSMODEM_SEND-1/2/06: MODEM_RTS_STATUS_MSG 1
00:12:32: RSMODEM_dRCV-2:l1258600700,MODEM_RTS_STATUS_MSG: 0 6 0 13 0 0 0 0 0
00:12:33: RSMODEM_SEND-1/2/06: MODEM_SET_DTR_STATE_MSG 0
00:12:33: RSMODEM_SEND-1/2/06: MODEM_RTS_STATUS_MSG 1
00:12:33: RSMODEM_dRCV-2:l1258662108,MODEM_RTS_STATUS_MSG: 0 6 0 16 0 0 0 0 0
00:12:35: RSMODEM_sRCV-2:l0,MODEM_POLL_MSG: 0 16 0 7 0 146 1 34 22
00:12:38: RSMODEM_SEND-1/2/06: MODEM_SET_DTR_STATE_MSG 1
00:12:47: RSMODEM_sRCV-2:l0,MODEM_POLL_MSG: 0 16 0 7 0 146 0 12 22
```

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

Table 1: debug modem dsip Field Descriptions

Field	Description
RSMODEM_SEND-1/2/06	Router shelf modem shelf sends a MODEM_RING_INDICATION_MSG message.
RSMODEM_sRCV-1/2/06	Router shelf modem received a MODEM_CALL_ACK_MSG message.
MODEM_CALL_ACCEPT_MSG	Router shelf accepts the call.
MODEM_CALL_HANGUP_MSG	Router shelf sends a hangup message.
MODEM_RTS_STATUS_MSG	Request to send message status.

Related Commands

Command	Description
debug dsip	Displays output for DSIP used between the router shelf and the dial shelf.
debug modem traffic	Displays output for framed, unframed, and asynchronous data transmission received from the modem cards.

debug modem oob

To debug the out-of-band port used to poll modem events on the modem, use the **debug modem oob**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem oob [{*slot*/*modem-port* | **group** *group-number*}]
no debug modem oob [{*slot*/*modem-port* | **group** *group-number*}]

Syntax Description

slot/modem-port	(Optional) The slot and modem port number.
group *group-number*	(Optional) The modem group.

Command Modes

Privileged EXEC

Usage Guidelines

The message types and sequence numbers that appear in the debugging output are initiated by the Modem Out-of-Band Protocol and used by service personnel for debugging purposes.

Caution Entering the **debug modem oob** command without specifying a slot and modem number debugs *all* out-of-band ports, which generates a substantial amount of information.

Examples

The following is sample output from the **debug modem oob**command. This example debugs the out-of-band port on modem 2/0, which creates modem startup messages between the network management software and the modem.

```
Router# debug modem oob 2/0
MODEM(2/0): One message sent --Message type:3, Sequence number:0
MODEM(2/0): Modem DC session data reply
MODEM(2/0): One message sent --Message type:83, Sequence number:1
MODEM(2/0): DC session event =
MODEM(2/0): One message sent --Message type:82, Sequence number:2
MODEM(2/0): No status changes since last polled
MODEM(2/0): One message sent --Message type:3, Sequence number:3
MODEM(2/0): Modem DC session data reply
MODEM(2/0): One message sent --Message type:83, Sequence number:4
```

debug modem relay errors

To view modem relay network errors, use the **debug modem relay errors** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **errors**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **errors**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers and the Cisco AS5300 universal access server.

Usage Guidelines

In a stable modem relay network, the **debug modem relay errors** command produces little output.

Examples

The following is sample output from the **debug modem relay errors** command. The output shows the sequence number of the packet, time stamp, direction, layer, and payload bytes, followed by each byte of the payload in hexadecimal.

```
Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 28 tm 11944 OUT ERR, pb=12, payload: 00 06
 00 00 00 00 00 07 00 00 01 DE
*Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 29 tm 11944 OUT ERR, pb=12, payload: 00
06 00 00 00 00 00 04 00 00 00 BE
*Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 30 tm 11944 OUT ERR, pb=12, payload: 00
06 00 00 00 00 00 05 FF FF FF FD
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay events	Displays events that may cause failure of the modem relay network.

debug modem relay events

To view the events that may cause failure of the modem relay network, use the **debug modem relay events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **events**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **events**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers and the Cisco AS5300 universal access server.

Usage Guidelines

In a stable modem relay network, the **debug modem relay events**command produces little output.

Examples

The following is sample output from the **debug modem relay events** command. The output shows the sequence number of the packet, time stamp, direction, layer, and payload bytes, followed by each byte of the payload in hexadecimal.

```
Router# debug modem relay events
Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 28 tm 11944 OUT EVNT, pb=12, payload: 00
06 00 00 00 00 00 07 00 00 01 DE
*Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 29 tm 11944 OUT EVNT, pb=12, payload: 00
06 00 00 00 00 00 04 00 00 00 BE
*Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 30 tm 11944 OUT EVNT, pb=12, payload: 00
06 00 00 00 00 00 05 FF FF FF FD
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay errors	Displays modem relay network errors.

debug modem relay packetizer

To view events occurring in the modem relay packetizer module, use the **debug modem relay packetizer** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **packetizer**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **packetizer**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers, and the Cisco AS5300 universal access server.

Usage Guidelines

Disable console logging and use buffered logging before using the **debug modem relay packetizer**command. Using the **debug modem relay packetizer**command generates a large volume of debugs, which can affect router performance.

Examples

The following is sample output from the **debug modem relay packetizer** command. The output shows the sequence number of the packet, time stamp, direction, layer, and payload bytes, followed by each byte of the payload in hexadecimal.

```
Router# debug modem relay packetizer
```

```
Jan 11 05:33:33.715:ModemRelay pkt[0:D:11]. sqn 8 tm 47610 IN PKTZR, pb=7, payload: 82 38
00 18 03 01 87
*Jan 11 05:33:33.727:ModemRelay pkt[0:D:11]. sqn 9 tm 47616 OUT PKTZR, pb=7, payload: 82
20 00 18 03 01 47
*Jan 11 05:33:35.719:ModemRelay pkt[0:D:11]. sqn 10 tm 49614 IN PKTZR, pb=7, payload: 82
39 00 18 03 01 87
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay errors	Displays modem relay network errors.

debug modem relay physical

To view modem relay physical layer packets, use the **debug modem relay physical** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **physical**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **physical**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers, and the Cisco AS5300 universal access server.

Usage Guidelines

Disable console logging and use buffered logging before using the **debug modem relay physical**command. Using the **debug modem relay physical**command generates a large volume of debugs, which can affect router performance.

Examples

The following is sample output from the **debug modem relay physical** command. The output shows the sequence number of the packet, time stamp, direction, layer, and payload bytes, followed by each byte of the payload in hexadecimal.

```
Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 28 tm 11944 OUT PHYS, pb=12, payload: 00
06 00 00 00 00 00 07 00 00 01 DE
*Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 29 tm 11944 OUT PHYS, pb=12, payload: 00
06 00 00 00 00 00 04 00 00 00 BE
*Jan 11 05:35:09.119:ModemRelay pkt[0:D:11]. sqn 30 tm 11944 OUT PHYS, pb=12, payload: 00
06 00 00 00 00 00 05 FF FF FF FD
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay errors	Displays modem relay network errors.

debug modem relay sprt

To view modem relay Simple Packet Relay Transport (SPRT) protocol packets, use the **debug modem relay sprt** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **sprt**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **sprt**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers, and the Cisco AS5300 universal access server.
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.

Usage Guidelines

Disable console logging and use buffered logging before using the **debug modem relay sprt** command. Using the **debug modem relay sprt** command generates a large volume of debugs, which can affect router performance.

Examples

The following is sample output from the **debug modem relay sprt** command. The output shows the sequence number of the packet, time stamp, direction, layer, and payload bytes, followed by each byte of the payload in hexadecimal.

```
Jan 11 05:37:16.151:ModemRelay pkt[0:D:11]. sqn 34 tm 7910 OUT SPRT, pb=4, payload: 02 00
03 71
*Jan 11 05:37:16.295:ModemRelay pkt[0:D:11]. sqn 35 tm 8048 IN SPRT, pb=13, payload: 02 00
 01 F1 F7 7E FD F5 90 F3 3E 90 55
*Jan 11 05:37:16.303:ModemRelay pkt[0:D:11]. sqn 36 tm 8060 IN SPRT, pb=6, payload: 02 00
01 41 04 00
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay errors	Displays modem relay network errors.

debug modem relay udp

To view events occurring in the User Datagram Protocol (UDP) stack, use the **debug modem relay udp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **udp**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **udp**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers, and the Cisco AS5300 universal access server.
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.

Usage Guidelines

Disable console logging and use buffered logging before using the **debug modem relay udp** command. Using the **debug modem relay udp** command generates a large volume of debugs, which can affect router performance.

Examples

The following is sample output from the **debug modem relay udp** command. The output shows three UDP packets related to modem relay. In the sample output, OUT or IN represent packet direction, and UDP indicates the specific layer that reported the packet.

```
Jan 1 03:39:29.407:ModemRelay pkt[0:D (4)]. sqn 61 tm 3060 OUT UDP, pb=6, payload: 80 00
00 00 00 00
*Jan 1 03:39:29.471:ModemRelay pkt[0:D (4)]. sqn 62 tm 3120 IN UDP, pb=6, payload: 40 00
00 00 00 00
*Jan 1 03:39:29.471:ModemRelay pkt[0:D (4)]. sqn 63 tm 3120 IN UDP, pb=6, payload: 80 00
00 00 00 00
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay errors	Displays modem relay network errors.

debug modem relay v14

To observe events occurring in the V.14 layer, use the **debug modem relay v14** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **v14**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **v14**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

No debugging output is displayed.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(2)T	This command was introduced .

Usage Guidelines

Use the **debug modem relay v14**command to debug V.14 layer modem relay calls. Using this command generates a large volume of debugs, which can affect router performance; disable console logging and use buffered logging before using the **debug modem relay v14**command. In most instances you will use this command only at the request of Cisco Technical Assistance Center (TAC).

Examples

The following is sample output from the **debug modem relay v14** command. The output shows the sequence number of the packet time stamp, direction, layer, and payload bytes, followed by each byte of the payload in hexadecimal.

```
Router# debug modem relay v14
*Aug 10 22:51:37.496: ModemRelay pkt[1/1:1]. sqn 15649 tm 48766 OUT V14, pb=18, payload:
08 BC 4C 51 CE 1A 69 ED D6 65 62 8C 7F D3 9A 82 5A 7A
*Aug 10 22:51:38.216: ModemRelay pkt[1/1:1]. sqn 15650 tm 48778 IN V14, pb=22, payload:
9A 9C 7F 57 2D D7 4C 98 E8 EC FC 73 69 F2 FF A3 E8 B0 A4 58 BB AE
*Aug 10 22:51:38.216: ModemRelay pkt[1/1:1]. sqn 15651 tm 48790 OUT V14, pb=18, payload:
64 F9 73 D3 AB 11 61 ED 1E 5D 51 8D B1 9F CA 49 BF F4
*Aug 10 22:51:38.216: ModemRelay pkt[1/1:1]. sqn 15652 tm 48796 IN V14, pb=21, payload:
C1 77 90 12 F8 37 E8 7A 64 8D 0E 61 58 7E E4 E8 87 E0 B4 83 C7 A4 60 7A 64 8B 09 B9 80 2E
E5 2E 94 65 79 C2 A8 E9 6F D9 6C 3B
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.

Command	Description
debug modem relay errors	Displays modem relay network errors.

debug modem relay v42

To view events occurring in the V.42 layer, use the **debug modem relay v42** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem relay [**call-identifier** *call-setup-time call-index*] **v42**
no debug modem relay [**call-identifier** *call-setup-time call-index*] **v42**

Syntax Description

call-identifier	(Optional) Identifies a particular call.
call-setup-time	(Optional) Value of the system UpTime when the call associated with this entry was started. Valid values are 0 through 4294967295.
call-index	(Optional) Dial peer identification number used to distinguish between calls with the same setup time. Valid values are 0 through 10.

Command Default

This command is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(11)T	This command was introduced for the Cisco 2600, Cisco 3620, Cisco 3640, Cisco 3660, and Cisco 7200 series routers, and the Cisco AS5300 universal access server.
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.

Usage Guidelines

Disable console logging and use buffered logging before using the **debug modem relay v42** command. Using the **debug modem relay v42** command generates a large volume of debugs, which can affect router performance.

Examples

The following is sample output from the **debug modem relay v42** command. The output shows the sequence number of the packet, timestamp, direction, layer, and payload-bytes, followed by each byte of the payload in hexadecimal.

```
Jan 11 05:42:08.715:ModemRelay pkt[0:D:13]. sqn 3 tm 10104 OUT V42, pb=43, payload: 03 AF
82 80 00 13 03 03 8A 89 00 05 02 03 E0 06 02 03 E0 07 01 08 08 01 08 F0 00 0F 00 03 56 34
32 01 01 03 02 02 04 00 03 01 20
*Jan 11 05:42:08.847:ModemRelay pkt[0:D:13]. sqn 4 tm 10236 IN V42, pb=2, payload: 03 7F
```

Related Commands

Command	Description
debug hpi all	Displays gateway DSP modem relay termination codes.
debug modem relay errors	Displays modem relay network errors.

debug modem trace

To debug a call trace on the modem to determine why calls are terminated, use the **debug modem trace**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem trace [{**normal** | **abnormal** | **all**}] [{*slot/modem-port* | **group** *group-number*}]
no debug modem trace [{**normal** | **abnormal** | **all**}] [{*slot/modem-port* | **group** *group-number*}]

Syntax Description

normal	(Optional) Uploads the call trace to the syslog server on normal call termination (for example, a local user hangup or a remote user hangup).
abnormal	(Optional) Uploads the call trace to the syslog server on abnormal call termination (for example, any call termination other than normal termination, such as a lost carrier or a watchdog timeout).
all	(Optional) Uploads the call trace on all call terminations including normal and abnormal call termination.
slot/modem-port	(Optional) The slot and modem port number.
group *group-number*	(Optional) The modem group.

Command Modes

Privileged EXEC

Usage Guidelines

The **debug modem trace** command applies only to manageable modems. For additional information, use the **show modem**command.

Examples

The following is sample output from the **debug modem trace abnormal** command:

```
Router# debug modem trace abnormal 1/14
Modem 1/14 Abnormal End of Connection Trace. Caller 123-4567
    Start-up Response: AS5200 Modem, Firmware 1.0
    Control Reply: 0x7C01
    DC session response: brasil firmware 1.0
    RS232 event:
    DSR=On, DCD=On, RI=Off, TST=Off
    changes: RTS=No change, DTR=No change, CTS=No change
    changes: DSR=No change, DCD=No change, RI=No change, TST=No change
    Modem State event: Connected
    Connection event: Speed = 19200, Modulation = VFC
    Direction = Originate, Protocol = reliable/LAPM, Compression = V42bis
    DTR event: DTR On
    Modem Activity event: Data Active
    Modem Analog signal event: TX = -10, RX = -24, Signal to noise = -32
    End connection event: Duration = 10:34-11:43,
    Number of xmit char =    67, Number of rcvd char = 88, Reason: Watchdog Time-out.
```

Related Commands

Command	Description
debug modem csm	Debugs the CSM used to connect calls on the modem.

Command	Description
debug modem oob	Creates modem startup messages between the network management software and the modem on the specified OOB port.

debug modem traffic

To display output for framed, unframed, and asynchronous data sent received from the modem cards, use the **debug modem traffic** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug modem traffic
no debug modem traffic

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.3(2)AA	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The **debug modem traffic** command displays output for framed, unframed, and asynchronous data sent or received by the modem cards.

Examples

The following example displays information about unframed or framed data sent to or received from the modem cards:

```
Router# debug modem traffic
MODEM-RAW-TX:modem = 6/5/00, length = 1, data = 0x61, 0xFF, 0x7D, 0x23
MODEM-RAW-RX:modem = 6/5/00, length = 1, data = 0x61, 0x0, 0x0, 0x0
```

The information indicates unframed asynchronous data transmission and reception involving the modem on shelf 6, slot 5, port 00.

The following example displays framed asynchronous data transmission and reception involving the modem on shelf 6, slot 5, port 00:

```
Router# debug modem traffic
MODEM-FRAMED-TX:modem = 6/5/00, length = 8, data = 0xFF, 0x3, 0x82
MODEM-FRAMED-RX:modem = 6/5/00, length = 14, data = 0xFF, 0x3, 0x80
```

Related Commands

Command	Description
debug modem dsip	Displays output for modem control messages that are received or sent to the router.

debug mpls adjacency

To display changes to label switching entries in the adjacency database, use the **debugmplsadjacency**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls adjacency
no debug mpls adjacency

Usage Guidelines

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 command syntax.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use the **debugmplsadjacency** command to monitor when entries are updated in or added to the adjacency database.

Examples

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

```
Router# debug mpls adjacency
TAG ADJ: add 10.10.0.1, Ethernet0/0/0
TAG ADJ: update 10.10.0.1, Ethernet0/0/0
```

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

Table 2: debug mpls adjacency Field Description

Field	Description
add	Adding an entry to the database.
update	Updating the MAC address for an existing entry.
10.10.0.1	Address of neighbor TSR.
Ethernet0/0/0	Connecting interface.

debug mpls atm-cos

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

To display ATM label virtual circuit (VC) bind or request activity that is based on the configuration of a Quality of Service (QoS) map, use the **debug mpls atm-cos** command in privileged EXEC mode. To disable this feature, use the **no** form of this command.

debug mpls atm-cos [{**bind** | **request**}]
no debug mpls atm-cos [{**bind** | **request**}]

Syntax Description

bind	(Optional) Specifies debug information about bind responses for a VC path.
request	(Optional) Specifies debug information about bind requests for a VC path.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC (#)

Command History

Release	**Modification**
12.0(5)T	This command was introduced.
12.0(10)ST	This command was modified to reflect MPLS IETF syntax and terminology.
12.2(2)T	This command was incorporated into Cisco IOS Release 12.2(2)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(20)T	This command was removed.

Examples

The following command sequence demonstrates how to obtain sample output from the **debug mpls atm-cos** command.

First, display the Multiprotocol Label Switching (MPLS) forwarding table to see which prefixes are associated with a single label VC (LVC), as shown below:

```
Router# show mpls forwarding
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
26     28          10.17.17.17/32    0          PO6/0      point2point
27     Pop tag     10.11.11.11/32    1560       PO6/0      point2point
28     27          10.16.16.16/32    0          PO6/0      point2point
29     30          10.92.0.0/8       0          PO6/0      point2point
30     Pop tag     10.95.0.0/8       2600       PO6/0      point2point
31     2/34        10.10.10.10/32    0          AT2/0.1    point2point
32     Pop tag     10.14.14.14/32    0          Fa5/0      10.91.0.1
33     Pop tag     10.90.0.0/8       0          Fa5/0      10.91.0.1
```

```
34      Pop tag     10.96.0.0/8       0          Fa5/0      10.91.0.1
        2/36        10.96.0.0/8       0          AT2/0.1    point2point
35      35          10.93.0.0/8       0          PO6/0      point2point
36      36          10.12.12.12/32    0          PO6/0      point2point
37      37          10.15.15.15/32    0          PO6/0      point2point
38      37          10.18.18.18/32    0          Fa5/0      10.91.0.1
39      39          10.97.0.0/8       540        PO6/0      point2point
40      40          10.98.0.0/8       0          PO6/0      point2point
```

Second, enable debugging of request and bind events, as shown in the command sequence below:

```
Router# debug mpls atm-cos ?
  bind     Bind response for VC path
  request  Requests for VC binds path
Router# debug mpls atm-cos request
ATM TAGCOS VC requests debugging is on
Router# debug mpls atm-cos bind
ATM TAGCOS Bind response debugging is on
```

Third, configure an MPLS ATM subinterface for multi-VC mode. The corresponding request and bind events are displayed, as shown below:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface a2/0.1
Router(config-subif)# mpls atm multi-vc
Router(config-subif)# end
Router#
19:59:14:%SYS-5-CONFIG_I:Configured from console by console
Router#
19:59:24:TAGCOS-REQ:vc request 10.10.10.10/32, available
19:59:24:TAGCOS-REQ:vc request 10.10.10.10/32, standard
19:59:24:TAGCOS-REQ:vc request 10.10.10.10/32, premium
19:59:24:TAGCOS-REQ:vc request 10.10.10.10/32, control
19:59:24:TAGCOS-REQ:vc request 10.96.0.0/8, available
19:59:24:TAGCOS-REQ:vc request 10.96.0.0/8, standard
19:59:24:TAGCOS-REQ:vc request 10.96.0.0/8, premium
19:59:24:TAGCOS-REQ:vc request 10.96.0.0/8, control
TAGCOS-REQ/TCATM:10.11.11.11/32,len=4352,band=1099528405504,class=0x700
TAGCOS-REQ/TCATM:10.12.12.12/32,len=4352,band=2199040033280,class=0x700
TAGCOS-REQ/TCATM:10.13.13.13/32,len=4352,band=3298551661056,class=0x700
TAGCOS-REQ/TCATM:10.14.14.14/32,len=4352,band=4398063288832,class=0x700
TAGCOS-REQ/TCATM:10.15.15.15/32,len=4352,band=5497574916608,class=0x700
TAGCOS-REQ/TCATM:10.16.16.16/32,len=4352,band=6597086544384,class=0x700
TAGCOS-REQ/TCATM:10.17.17.17/32,len=4352,band=7696598172160,class=0x700
TAGCOS-REQ/TCATM:10.18.18.18/32,len=4352,band=8796109799936,class=0x700
TAGCOS-REQ/TCATM:10.90.0.0/8,len=768,band=3940649674539009,class=0x2
TAGCOS-REQ/TCATM:10.91.0.0/8,len=768,band=3940649674604545,class=0x2
TAGCOS-REQ/TCATM:10.92.0.0/8,len=768,band=3940649674670081,class=0x2
TAGCOS-REQ/TCATM:10.93.0.0/8,len=768,band=3940649674735617,class=0x2
TAGCOS-REQ/TCATM:10.94.0.0/8,len=768,band=3940649674801153,class=0x2
TAGCOS-REQ/TCATM:10.95.0.0/8,len=768,band=3940649674866689,class=0x2
TAGCOS-REQ/TCATM:10.97.0.0/8,len=768,band=3940649674932225,class=0x2
TAGCOS-REQ/TCATM:10.98.0.0/8,len=768,band=3940649674997761,class=0x2
TAGCOS-BIND:binding_ok 10.10.10.10/32,VCD=41 - control 41,41,41,41
TAGCOS-BIND:binding_ok 10.10.10.10/32, Inform TFIB pidx=0, in_tag=31, idx=0x80000000
TAGCOS-BIND:binding_ok 10.96.0.0/8,VCD=42 - control 42,42,42,42
TAGCOS-BIND:binding_ok 10.96.0.0/8, Inform TFIB pidx=1, in_tag=34, idx=0x80000001
TAGCOS-BIND:binding_ok 10.10.10.10/32,VCD=43 - premium 43,43,43,41
TAGCOS-BIND:binding_ok 10.96.0.0/8,VCD=44 - premium 44,44,44,42
TAGCOS-BIND:binding_ok 10.10.10.10/32,VCD=45 - standard 45,45,43,41
TAGCOS-BIND:binding_ok 10.96.0.0/8,VCD=46 - standard 46,46,44,42
```

```
TAGCOS-BIND:binding_ok 10.10.10.10/32,VCD=47 - available 47,45,43,41
TAGCOS-BIND:binding_ok 10.96.0.0/8,VCD=48 - available 48,46,44,42
```

debug mpls atm-ldp api

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

To display information about the virtual channel identifier (VCI) allocation of label virtual circuits (LVCs), label-free requests, and cross-connect requests, use the **debugmplsatm-ldpapi**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls atm-ldp api
no debug mpls atm-ldp api

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 modified.
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.
12.4(20)T	This command was removed.

Usage Guidelines

Use the **debugmplsatm-ldpapi** command in conjunction with the**debugmplsatm-ldproutes**and **debugmplsatm-ldpstates**command to display more complete information about an LVC.

Examples

The following shows sample output from the **debugmplsatm-ldpapi** command:

```
Router# debug mpls atm-ldp api
Tailend Router Free label Req 167.50.0.0 on ATM0/0.2 VPI/VCI 1/674
 TAGATM_API: received label free request
             interface: ATM0/0.2  dir: in  vpi: 1  vci: 674
 TAGATM_API: completed label free
             interface: ATM0/0.2  vpi: 1  vci: 674
             result: TAGATM_OK
```

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

Table 3: debug mpls atm-ldp api Field Descriptions

Field	Description
TAGATM_API	Subsystem that displays the message.
interface	Interface used by the driver to allocate or free VPI/VCI resources.
dir	Direction of the VC: • In--Input or receive VC • Out--Output VC
vpi	Virtual path identifier.
vci	Virtual channel identifier.
result	The return error code from the driver API.

Related Commands

Command	Description
debug mpls atm-ldp states	Displays information about LVC state transitions as they occur.

debug mpls atm-ldp failure

To display failure information about the LC-ATM, use the **debug mpls atm-ldp failure** command in privileged EXEC mode. To disable this feature, use the no form of the command.

debug mpls atm-ldp failure
no debug mpls atm-ldp failure

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(8)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS 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.
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.

Usage Guidelines

Use the **debug mpls atm-ldp failure** command to display failure information about the LC-ATM. This command is useful for determining failure cases. This command displays only failure information, unlike the **debug mpls atm-ldp api** command, which displays all API events.

Examples

This section shows sample output from the **debug mpls atm-ldp failure** command.

The following failure message displays during a race condition where the LC-ATM attempts to allocate label virtual circuits (LVCs) on an interface where MPLS has been disabled:

```
Router# debug mpls atm-ldp failure
TAGATM_API_FAILURE: allocate_tag_req on ATM1/0/0 tagsw not enabled
```

The following failure message displays when the LC-ATM fails to deallocate the output leg LVC of a cross connect:

```
Router# debug mpls atm-ldp failure
TAGATM_API_FAILURE: connDeAllocateHalfLeg returned false interface: ATM1/0/0
vpi: 1  vci: 48
```

The following failure message displays when a cross connect cannot be installed on the switching fabric. The result code is also provided.

```
Router# debug mpls atm-ldp failure
TAGATM_API_FAILURE: setup_xconn_req InstallSvcXconn failed result
```

The following message displays when attempts to establish a cross connect fail. The result describes the reason for the failure.

```
Router# debug mpls atm-ldp failure
TCATM-4-XCONNECT_FAILED: 10.254.13.237/32 for ATM0/1/2 ATM1/0/0
TAGATM_API: x-conn setup request completed
             input interface: ATM0/1/2  vpi: 1  vci: 48
            output interface: ATM1/0/0  vpi: 2  vci: 2038
            result = TAGATM_FAIL
Xconnect setup response for 10.254.13.215: failure, 8
```

The following message displays when attempts to remove a cross connect fail. The result describes why the cross connect cannot be removed.

```
Router# debug mpls atm-ldp failure
TCATM-4-XCONNECT_REMOVE_FAILED: Remove XConnect API failed for ATM1/0/12 1/894
-> ATM1/0/13 1/528
TAGATM_API: x-conn remove request completed
             input interface: ATM1/0/12  vpi: 1  vci: 894
            output interface: ATM1/0/13  vpi: 1  vci: 528
            result = TAGATM_FAIL
```

Related Commands

Command	Description
debug mpls atm-ldp api	Displays all driver API events.

debug mpls atm-ldp routes

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

To display information about the state of the routes for which virtual circuit identifier (VCI) requests are being made, use the **debugmplsatm-ldproutes**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls atm-ldp routes
no debug mpls atm-ldp routes

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC (#)

Command History

Command	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 modified.
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.
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.

Command	Modification
12.4(20)T	This command was removed.

Usage Guidelines

When there are many routes and system activities (that is, shutting down interfaces, learning new routes, and so forth), the **debugmplsatm-ldproutes** command displays extensive information that might interfere with system timing. Most commonly, this interference affects normal label distribution protocol (LDP) operation. To avoid this problem, you can increase the LDP hold time by means of the **mplsldpholdtime** command.

Examples

The following shows sample output from the**debugmplsatm-ldproutes** command:

```
Router# debug mpls atm-ldp routes
CleanupRoutes,not deleting route of idb ATM0/0.2,rdbIndex 0
tcatmFindRouteTags,153.7.0.0/16,idb=ATM0/0.2,nh=134.111.102.98,index=0
AddNewRoute,153.7.0.0/16,idb=ATM0/0.2
CleanupRoutes,153.7.0.0/16
CleanupRoutes,not deleting route of idb ATM0/0.2,rdbIndex 0
tcatmFindRouteTags,153.8.0.0/16,idb=ATM0/0.2,nh=134.111.102.98,index=0
AddNewRoute,153.8.0.0/16,idb=ATM0/0.2
CleanupRoutes,153.8.0.0/16
CleanupRoutes,not deleting route of idb ATM0/0.2,rdbIndex 0
tcatmFindRouteTags,153.9.0.0/16,idb=ATM0/0.2,nh=134.111.102.98,index=0
AddNewRoute,153.9.0.0/16,idb=ATM0/0.2
CleanupRoutes,153.9.0.0/16
CleanupRoutes,not deleting route of idb ATM0/0.2,rdbIndex 0
tcatmFindRouteTags,153.10.0.0/16,idb=ATM0/0.2,nh=134.111.102.98,index=0
AddNewRoute,153.10.0.0/16,idb=ATM0/0.2
CleanupRoutes,153.10.0.0/16
CleanupRoutes,not deleting route of idb ATM0/0.2,rdbIndex 0
tcatmFindRouteTags,153.11.0.0/16,idb=ATM0/0.2,nh=134.111.102.98,index=0
AddNewRoute,153.11.0.0/16,idb=ATM0/0.2
CleanupRoutes,153.11.0.0/16
```

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

Table 4: debug mpls atm-ldp routes Field Descriptions

Field	Description
CleanupRoutes	Cleans up the routing table after a route has been deleted.
not deleting route of idb ATM0/0.2	The route cleanup event has not removed the specified route.
rdbIndex	Index identifying the route.
tcatmFindRouteTags	Request a VC for the route.
idb	The internal descriptor for an interface.
nh	Next hop for the route.
index	Identifier for the route.
AddNewRoute	Action of adding routes for a prefix or address.

Related Commands

Command	Description
mpls ldp holdtime	Changes the time an LDP session is maintained in the absence of LDP messages from the session peer.

debug mpls atm-ldp states

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

To display information about label virtual circuit (LVC) state transitions as they occur, use the **debugmplsatm-ldpstates**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls atm-ldp states
no debug mpls atm-ldp states

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 modified.
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.
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.

Release	Modification
12.4(20)T	This command was removed.

Usage Guidelines

When there are many routes and system activities (such as shutting down interfaces, learning new routes, and so forth), the **debugmplsatm-ldpstates** command outputs extensive information that might interfere with system timing. Most commonly, this interference affects normal label distribution protocol (LDP) operation. To avoid this problem, you should increase the LDP hold time by means of the **mplsldpholdtime** command.

Examples

The following shows sample output from the**debugmplsatm-ldpstates**command:

```
Router# debug mpls atm-ldp states
Transit Output 166.35.0.0 VPI/VCI 1/67 Active -> XmitRelease NoPath
Transit Input 166.35.0.0 VPI/VCI 1/466 Active -> ApiWaitParentLoss ParentLoss
Transit Input 166.35.0.0 VPI/VCI 1/466 ApiWaitParentLoss -> ParentWait ApiSuccess
Transit Input 166.35.0.0 VPI/VCI 1/466 ParentWait -> XmitWithdraw NoPath
Transit Input 166.35.0.0 VPI/VCI 1/466 XmitWithdraw -> XmitWithdraw Transmit
Transit Input 166.35.0.0 VPI/VCI 1/466 XmitWithdraw -> NonExistent Release
Transit Input 166.35.0.0 VPI/VCI 1/466 NonExistent -> NonExistent ApiSuccess
```

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

Table 5: debug mpls atm-ldp states Field Descriptions

Field	Description
Transit Output	Output side of an LVC.
VPI/VCI	VC value.
Transit Input	Input side of an LVC.

Related Commands

Command	Description
mpls ldp holdtime	Changes the time an LDP session is maintained in the absence of LDP messages from the session peer.

debug mpls checkpoint label-binding

To display the events for the checkpoint label bindings of Multiprotocol Label Switching (MPLS) applications running on the router, use the debug mpls checkpoint label-binding command in privileged EXEC mode. To disable the display of these events, use the **no** form of this command.

debug mpls checkpoint label-binding
no debug mpls checkpoint label-binding

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 router.
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

Use this command with caution. The command displays the events for every label binding.

Examples

The following example shows the output when you issue the command on the standby Route Processor:

```
Router# debug mpls checkpoint label-binding
MPLS Label Binding Checkpoint debugging is on
3d17h: mpls_lbl_bind_chkpt: client ID 13 up, total client 1
3d17h: mpls_lbl_bind_chkpt: msg rx for 1D, vers 0, type 1
action 56, len 0, state 4, peer 13
3d17h: mpls_lbl_bind_chkpt: post msg type 1
3d17h: mpls_lbl_bind_chkpt: msg rx for 1D, vers 0, type 1
action 56, len 0, state 4, peer 13
3d17h: mpls_lbl_bind_chkpt: post msg type 1
3d17h: mpls_lbl_bind_chkpt: msg rx for 1D, vers 0, type 1
action 56, len 0, state 4, peer 13
3d17h: mpls_lbl_bind_chkpt: post msg type 1
3d17h: mpls_lbl_bind_chkpt: appl_id 13, KEY 000C800018888200
3d17h: mpls_chkpt_db: AVL insert successful, Key 000C800018888200 action Add, label 19
3d17h: mpls_lbl_bind_chkpt: appl_id 13, KEY 000C800013200080
3d17h: mpls_chkpt_db: AVL insert successful, Key 000C800013200080 action Add, label 20
3d17h: mpls_lbl_bind_chkpt: appl_id 13, KEY 000C8000138383838200
3d17h: mpls_chkpt_db: AVL insert successful, Key 000C8000138383838200 action Add, label 21
3d17h: Stby RP OR CF peer not ready, don't send msg
3d17h: mpls_lbl_bind_chkpt: client ID 13 down, total client 0
3d17h: mpls_lbl_bind_chkpt: msg rx for 1D, vers 0, type 1
action 56, len 2, state 4, peer 13
3d17h: mpls_lbl_bind_chkpt: post msg type 1
3d17h: mpls_lbl_bind_chkpt: appl_id 13, KEY  action NSF unconfig, appl id 13
```

Related Commands

Command	Description
debug ip bgp vpnv4 checkpoint	Display the events for the VRF checkpointing system between the active and standby Route Processors.

debug mpls events

To display information about significant Multiprotocol Label Switching (MPLS) events, use the **debug mpls events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls events
no debug mpls events

Syntax Description

This command has no keywords or arguments.

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

Usage Guidelines

Use this command to monitor significant MPLS events.

Examples

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

```
Router# debug mpls events

MPLS events debugging is on
TAGSW: Unbound IP address, 155.0.0.55, from Router ID
TAGSW: Bound IP address, 199.44.44.55, to Router ID
```

debug mpls infra label-broker api

To display Multiprotocol Label Switching (MPLS) label-broker API error messages, use the **debug mpls infra label-broker api** command in privileged EXEC mode. To disable the display of the messages, use the **no** form of this command.

debug mpls infra label-broker api [{**ipv4** | **ipv6** | [{**default** | **vrf** *vrf-name* }] | **prefix-list** { *prefix-name*}}]
no debug mpls infra label-broker api [{**ipv4** | **ipv6** | [{**default** | **vrf** *vrf-name* }] | **prefix-list** { *prefix-name*}}]

Syntax Description

ipv4	(Optional) Displays track labels for IPv4 prefixes.
ipv6	(Optional) Displays track labels for IPv6 prefixes.
default	(Optional) Displays the default routing/forwarding table.
vrf *vrf-name*	(Optional) Displays debugging information for the specified Virtual Private Network (VPN) routing and forwarding (VRF) instance. You can find VRF names using the **show ip vrf** command.
prefix-list	(Optional) Displays debugging information for the specified prefix list.
prefix-name	The name of the prefix list. You can find prefix list names using the **show ip prefix-list** command.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.8S	This command was introduced.

Usage Guidelines

To enable the **debug mpls infra label-broker api** command, the user must first enter global configuration mode and then enter the **service internal** command, followed by the **end** command.

Example

The following shows how to enable the **debug mpls infra label-broker api** command:

```
Device> enable
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# service internal
Device(config)# end
00:01:05: %SYS-5-CONFIG_I: Configured from console by console
Device# debug mpls infra label-broker api
MFI Label Broker:
  api debugging is on for all prefixes and labels
```

Related Commands

Command	Description
debug mpls infra label-broker api key	Displays information about the MFI label broker and track labels for key database entries.
debug mpls infra lfd label-block	Displays information about label-block debugging.
debug mpls infra lfd label-broker key	Displays information about keyed label debugging for all key entries.
service internal	Enables **infra** commands to be configured.
show ip prefix-list	Displays information about a prefix list or prefix list entries.
show ip vrf	Displays the set of defined VPN VRF instances and associated interfaces.
show xconnect	Displays information about xconnect attachment circuits and pseudowires.

debug mpls infra label-broker api key

To display Multiprotocol Label Switching (MPLS) application programming interface (API) key error messages, use the **debug mpls infra label-broker api key** command in privileged EXEC mode. To disable the display of the messages, use the **no** form of this command.

debug mpls infra label-broker api key [{ **vpnv4** | **vpnv6** | [{**rd** *ip-address*}]}] | [{ **per-vrf** [{**vrf** *vrf-name* | **default** | {**ipv4** | **ipv6**}}]}]
no debug mpls infra label-broker api key [{ **vpnv4** | **vpnv6** | [{**rd** *ip-address*}]}] | [{ **per-vrf** [{**vrf** *vrf-name* | **default** | {**ipv4** | **ipv6**}}]}]

Syntax Description

vpnv4	(Optional) Displays Virtual Private Network version 4 (VPNv4) events.
vpnv 6	(Optional) Displays Virtual Private Network version 6 (VPNv6) events.
rd	(Optional) Specifies a route distinguisher (RD) for a VPN routing and forwarding (VRF) instance.
ip-address	IPv4 or IPv6 address and mask.
per-vrf	(Optional) Specifies per-prefix label mode.
vrf *vrf-name*	(Optional) Displays debugging information for the specified Virtual Private Network (VPN) routing and forwarding (VRF) instance. You can find VRF names using the **show ip vrf** command.
ipv4	(Optional) Displays track labels for IPv4 prefixes.
ipv6	(Optional) Displays track labels for IPv6 prefixes.
default	(Optional) Displays the default routing/forwarding table.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.8S	This command was introduced.

Usage Guidelines

To enable the **debug mpls infra label-broker api key** command, the user must first enter global configuration mode, and then enter the **service internal** command, followed by the **end** command.

Example

The following shows how to enable the **debug mpls infra label-broker api key** command:

```
Device> enable
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# service internal
Device(config)# end
00:01:05: %SYS-5-CONFIG_I: Configured from console by console
Device# debug mpls infra label-broker api key
```

```
MFI Label Broker:
  api debugging is on for all IPv4 tables for IPv4 prefix list prefix-list
  api debugging is on for all IPv6 tables
  api debugging is on for all MPLS tables
  api debugging is on for all key entries
```

Related Commands

Command	Description
debug mpls infra label-broker api	Displays information about the MFI label broker and the API for all prefixes and labels.
debug mpls infra lfd label-block	Displays information about label-block debugging.
debug mpls infra lfd label-broker key	Displays information about keyed label debugging for all key entries.
service internal	Enables **infra** commands to be configured.
show ip prefix-list	Displays information about a prefix list or prefix list entries.
show ip vrf	Displays the set of defined VPN VRF instances and associated interfaces.
show xconnect	Displays information about xconnect attachment circuits and pseudowires.

debug mpls infra lfd label-block

To display information about label-block debugging, use the **debug mpls infra lfd label-block** command in privileged EXEC mode. To disable the display of the messages, use the **no** form of this command.

debug mpls infra lfd label-block [{**broker**}]
no debug mpls infra lfd label-block

Syntax Description

broker	(Optional) Displays debug messages for label-block broker events.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.8S	This command was introduced.

Usage Guidelines

To enable the **debug mpls infra lfd label-block** command, the user must first enter global configuration mode, and then enter the **service internal** command, followed by the **end** command.

Example

The following shows how to enable the **debug mpls infra lfd label-block** command:

```
Device> enable
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# service internal
Device(config)# end
00:01:05: %SYS-5-CONFIG_I: Configured from console by console
Device# debug mpls infra lfd label-block
  label block debugging is on
```

Related Commands

Command	Description
debug mpls infra label-broker api	Displays information about the MFI label broker and the API for all prefixes and labels.
debug mpls infra label-broker api key	Displays information about the MFI label broker and track labels for key database entries.
debug mpls infra lfd label-broker key	Displays information about keyed label debugging for all key entries.
service internal	Enables **infra** commands to be configured.
show ip prefix-list	Displays information about a prefix list or prefix list entries

Command	Description
show ip vrf	Displays the set of defined VPN VRF instances and associated interfaces.
show xconnect	Displays information about xconnect attachment circuits and pseudowires.

debug mpls infra lfd label-broker key

To display information about keyed label debugging for all key entries, use the **debug mpls infra lfd label-broker key** command in privileged EXEC mode. To disable the display of the messages, use the **no** form of this command.

debug mpls infra lfd label-broker key [{**per-vrf** | [{**default** | **vrf** | {*vrf-name* | {**ipv4** | **ipv6**}}}] | | [{**vpnv4** | **vpnv6** | [{**rd** | {*ASN***:***nn* | *ip-address***:***nn* | {*ipv4-address-maskipv6-address-prefix*}}}]}]}]
no debug mpls infra lfd label-broker key [{**per-vrf** | [{**default** | **vrf** | {*vrf-name* | {**ipv4** | **ipv6**}}}] | | [{**vpnv4** | **vpnv6** | [{**rd** | {*ASN***:***nn* | *ip-address***:***nn* | {*ipv4-address-maskipv6-address-prefix*}}}]}]}]

Syntax Description

per-vrf	(Optional) Specifies per-prefix label mode.
default	(Optional) Displays the default routing/forwarding table.
vrf	Displays debugging information for the specified Virtual Private Network (VPN) routing and forwarding (VRF) instance.
vrf-name	The name of the VRF instance. You can find VRF names using the **show ip vrf** command.
ipv4	Displays track labels for IPv4 prefixes.
ipv6	Displays track labels for IPv6 prefixes.
vpnv4	(Optional) Displays Virtual Private Network version 4 (VPNv4) events.
vpnv6	(Optional) Displays Virtual Private Network version 6 (VPNv6) events.
rd	(Optional) Specifies a route distinguisher (RD) for a VRF instance.
*asn***:***nn*	IP address and network number.
*ip-address***:***nn*	Autonomous system number (ASN) and network number.
ipv4-address-mask	IPv4 address and subnet mask of the remote peer.
ipv6-address-prefix	IPv6 address and prefix length of the remote peer.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.8S	This command was introduced.

Usage Guidelines

To enable the **debug mpls infra lfd label-broker key** command, the user must first enter global configuration mode, and then enter the **service internal** command, followed by the **end** command.

Example

The following shows how to enable the **debug mpls infra lfd label-broker key** command:

```
Device> enable
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# service internal
Device(config)# end
00:01:05: %SYS-5-CONFIG_I: Configured from console by console
Device# debug mpls infra lfd label-broker key
  keyed label debugging is on for all key entries
```

Related Commands

Command	Description
debug mpls infra label-broker api	Displays information about the MFI label broker and the API for all prefixes and labels.
debug mpls infra label-broker api key	Displays information about the MFI label broker and track labels for key database entries.
debug mpls infra lfd label-block	Displays information about label-block debugging.
service internal	Enables **infra** commands to be configured.
show ip prefix-list	Displays information about a prefix list or prefix-list entries.
show ip vrf	Displays the set of defined VPN VRF instances and associated interfaces.
show xconnect	Displays information about xconnect attachment circuits and pseudowires.

debug mpls ip iprm

To display debugging information for the Multiprotocol Label Switching (MPLS) IP Rewrite Manager (IPRM), use the debug mpls ip iprm command in privileged EXEC mode. To disable the display of this information, use the **no** form of this command.

debug mpls ip iprm
no debug mpls ip iprm

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(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Usage Guidelines

This command displays all output related to IPRM.

Examples

The command in the following examples display all IPRM debugging for the global routing table.

Cisco 7500 Series Example

```
Router# debug mpls ip iprm
IPRM debugging is on for global routing table
  iprm: prefix deleted: 10.0.0.44/32(glbl)
  iprm: delete mfi rewrite: 10.0.0.44/32(glbl)
  .
  .
  .
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38002
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois
  iprm: announce prefix local labels: lcatm; trans #80; 10.0.0.44/32(glbl); 0 labels; flags
 0x0
  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm: omit rewrite create: 10.0.0.44/32(glbl)
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38000
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois
  iprm: announce prefix local labels: lcatm; trans #81; 10.0.0.44/32(glbl); 0 labels; flags
 0x0
```

```
  iprm: get path labels: 10.0.0.44/32(glbl); nh 10.0.0.55(glbl), Et4/0/1; trans #81; recurs
 tree change
  iprm: ldm get path labels: 10.0.0.44/32(glbl), ldp; flags 0x8000
  iprm: announce prefix local labels: ldp; trans #81; 10.0.0.44/32(glbl); 1 label; flags
0x0
  iprm:    lab 21, ltbl 0
  iprm: announce path labels: ldp; trans #81; 10.0.0.44/32(glbl); 0 labels; flags 0x0
  iprm:    path: nh 10.0.0.55(glbl), Et4/0/1
  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm:    lcl lab 21, ltbl 0, ldp
  iprm:    path lab -, nh 10.0.0.55(glbl), Et4/0/1; ldp
  iprm: create mfi rewrite 10.0.0.44/32(glbl) passed: 2 fpis/1 mois
  iprm:    fpi[0] IV4, owner IPRM; 10.0.0.44/32; glbl
  iprm:    fpi[1] LBL, owner LDP; 21, ltbl 0
  iprm:    moi[0] PKT, flags 0x8; lab label-no-label; nh 10.0.0.55; nh if Et4/0/1 (nsf)
```

Cisco 10000 Series Example

```
Router# debug mpls ip iprm
IPRM debugging is on for global routing table
  iprm: prefix deleted: 10.0.0.44/32(glbl)
  iprm: delete mfi rewrite: 10.0.0.44/32(glbl)
  .
  .
  .
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38002
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois

  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm: omit rewrite create: 10.0.0.44/32(glbl)
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38000
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois

  iprm: get path labels: 10.0.0.44/32(glbl); nh 10.0.0.55(glbl), GigabitEthernet4/0/0; trans
 #81; recurs tree change
  iprm: ldm get path labels: 10.0.0.44/32(glbl), ldp; flags 0x8000
  iprm: announce prefix local labels: ldp; trans #81; 10.0.0.44/32(glbl); 1 label; flags
0x0
  iprm:    lab 21, ltbl 0
  iprm: announce path labels: ldp; trans #81; 10.0.0.44/32(glbl); 0 labels; flags 0x0
  iprm:    path: nh 10.0.0.55(glbl), GigabitEthernet4/0/0

  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm:    lcl lab 21, ltbl 0, ldp
  iprm:    path lab -, nh 10.0.0.55(glbl), GigabitEthernet4/0/0; ldp
  iprm: create mfi rewrite 10.0.0.44/32(glbl) passed: 2 fpis/1 mois
  iprm:    fpi[0] IV4, owner IPRM; 10.0.0.44/32; glbl
  iprm:    fpi[1] LBL, owner LDP; 21, ltbl 0
  iprm:    moi[0] PKT, flags 0x8; lab label-no-label; nh 10.0.0.55; nh if GigabitEthernet4/0/0
 (nsf)
```

The table below describes the significant fields shown in the display. The field descriptions also apply to the output of following debug commands:

- **debug mpls ip iprm cef**
- **debug mpls ip iprm events**
- **debug mpls ip iprm ldm**
- **debug mpls ip iprm mfi**

Table 6: debug mpls ip iprm Field Descriptions

Field	Description
discover prefix labels	The prefix labels that the IP LDM discovered.
announce prefix local labels announce path labels	IP LDMs pass prefix incoming (local) and outgoing (path) labels to IPRM by announcing the labels.
mfi rewrite	The information required by MPLS Forwarding Infrastructure (MFI) to create forwarding data structures for an MPLS forwarding equivalence class (FEC). For IP over MPLS a prefix is an MPLS FEC. An MFI rewrite includes a set of forwarding path identifier (FPI) and MPLS output information (MOI) elements.
fpi	Forwarding path identifier, which is required to locate MPLS forwarding information for a FEC. IP over MPLS deals with several types of FPIs, including IPv4 (IV4), IPv6 (IV6), and label (LBL) FPIs. **Note** The Cisco 10000 series router does not support IPv6.
moi	MPLS output information. For IP over MPLS, there is a MOI for each prefix path. The MOI includes the next hop (nh), outgoing interface (nh if), and outgoing label. IP over MPLS handles several types of MOIs, including packet (PKT) and ATM (ATM).
get/create/update MFI rewrite	The process IPRM uses to read (get) or update (create/update) an MFI rewrite.
recurs tree change	Recursion tree change. Cisco Express Forwarding notifies IPRM when the recursion tree (see below) for a prefix changes. IPRM responds by performing label discovery (see above).
recursion tree	A prefix known to Cisco Express Forwarding can have one or more paths (routes). Each is either a terminal path with a next hop and an outgoing interface or a recursive path with a next hop and no outgoing interface. The next hop for a recursive path typically matches a prefix known to Cisco Express Forwarding. That prefix also has one or more paths. The IP recursion tree for prefix P is a tree rooted at P's Cisco Express Forwarding entry with one of more path descendants. Terminal paths are leaf nodes in P's recursion tree and recursive paths are nonleaf nodes, each of which points to the Cisco Express Forwarding entry for its next hop.
glbl	The global (default) routing table.
ctxt	Context. Information used by IPRM when it performs label discovery.
flags	Information passed between IPRM and other components.
trans #	Transaction number used to identify an ongoing label discovery.
ltbl	Label table.
nsf	Nonstop forwarding.

Related Commands

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

debug mpls ip iprm cef

To display debugging information for interactions between Cisco Express Forwarding and the Multiprotocol Label Switching (MPLS) IP Rewrite Manager (IPRM), use the debug mpls ip iprm cef command in privileged EXEC mode. To disable the display of these events, use the **no** form of this command.

debug mpls ip iprm cef [{**table** {**all***table-id*} | **vrf** *vrf-name* | **acl** *acl-name* | **prefix-list** *prefix-list-name*}]
no debug mpls ip iprm cef

Syntax Description

table	(Optional) Displays the debugging information for one or more routing tables.
all	Displays debugging information for all routing tables.
table-id	The ID of the routing table for which you want to display debugging information. Table 0 is the default or global routing table.
vrf	(Optional) Displays debugging information for the VPN routing and forwarding (VRF) instance you specify.
vrf-name	The name of the VRF instance. You can find VRF names with the show ip vrf command.
acl	(Optional) Displays debugging information for the access control list (ACL) you specify.
acl-name	The name of the ACL. You can find ACL names with the show ip access-list command.
prefix-list	(Optional) Displays debugging information for the prefix list you specify.
prefix-list-name	The name of the prefix list. You can find prefix list names with the show ip prefix-list command.

Command Default

Debugging is not enabled. If you do not supply an optional keyword, all the debugging events are displayed.

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(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Usage Guidelines

This command limits the debug output to the IPRM interactions with Cisco Express Forwarding.

Examples

In the following example, IPRM events related to Cisco Express Forwarding are displayed.

Cisco 7500 Series Example

```
Router# debug mpls ip iprm cef
IPRM CEF interaction debugging is on for global routing table
iprm: prefix deleted: 10.0.0.44/32(glbl)
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38002
  iprm: announce prefix local labels: lcatm; trans #94; 10.0.0.44/32(glbl); 0 labels; flags
 0x0
  .
  .
  .
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38000
  iprm: announce prefix local labels: lcatm; trans #97; 10.0.0.44/32(glbl); 0 labels; flags
 0x0
  iprm: get path labels: 10.0.0.44/32(glbl); nh 10.0.0.55(glbl), Et4/0/1; trans #97; recurs
 tree change
  iprm: announce prefix local labels: ldp; trans #97; 10.0.0.44/32(glbl); 1 label; flags
0x0
  iprm:    lab 21, ltbl 0
  iprm: announce path labels: ldp; trans #97; 10.0.0.44/32(glbl); 0 labels; flags 0x0
  iprm:    path: nh 10.0.0.55(glbl), Et4/0/1
```

Cisco 10000 Series Example

```
Router# debug mpls ip iprm cef
IPRM CEF interaction debugging is on for global routing table
iprm: prefix deleted: 10.0.0.44/32(glbl)
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38002
  .
  .
  .
  iprm: discover prefix labels: 10.0.0.44/32(glbl); recurs tree change; ctxt 0x38000

  iprm: get path labels: 10.0.0.44/32(glbl); nh 10.0.0.55(glbl), GigabitEthernet4/0/0; trans
 #97; recurs tree change
  iprm: announce prefix local labels: ldp; trans #97; 10.0.0.44/32(glbl); 1 label; flags
0x0
  iprm:    lab 21, ltbl 0
  iprm: announce path labels: ldp; trans #97; 10.0.0.44/32(glbl); 0 labels; flags 0x0
  iprm:    path: nh 10.0.0.55(glbl), GigabitEthernet4/0/0
```

See the field descriptions for the **debug mpls ip iprm** command for an explanation of the fields displayed in the output.

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 ip iprm events

To display events related to the Multiprotocol Label Switching (MPLS) IP Rewrite Manager (IPRM), use the debug mpls ip iprm events command in privileged EXEC mode. To disable the display of these events, use the **no** form of this command.

debug mpls ip iprm events
no debug mpls ip iprm events

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(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Examples

See the command page for **debug mpls ip iprm** for sample command output and an explanation of the fields displayed in the output.

Related Commands

Command	Description
debug mpls ip iprm cef	Displays debugging information for interactions between Cisco Express Forwarding and the IPRM.
debug mpls ip iprm ldm	Displays debugging information for interactions between the LDMs and the MPLS IPRM.
debug mpls ip iprm mfi	Displays debugging information for interactions between the MFI and the MPLS IPRM.

debug mpls ip iprm ldm

To display debugging information for interactions between the IP Label Distribution Modules (LDMs) and the Multiprotocol Label Switching (MPLS) IP Rewrite Manager (IPRM), use the debug mpls ip iprm ldm command in privileged EXEC mode. To disable the display of this information, use the **no** form of this command.

debug mpls ip iprm ldm [{**bgp** | **lcatm** | **ldp** | **vpnv4** | **6pe** | **table** {**all** | **table-id**} | **vrf** **vrf-name** | **acl** **acl-name** | **prefix-list** **prefix-list-name**}]
no debug mpls ip iprm ldm

Cisco 10000 Series Routers
debug mpls ip iprm ldm [{**bgp** | **ldp** | **vpnv4** | **table** {**all** | **table-id**} | **vrf** **vrf-name** | **acl** **acl-name** | **prefix-list** **prefix-list-name**}]
no debug mpls ip iprm ldm

Syntax Description

bgp	(Optional) Displays Border Gateway Protocol (BGP) events.
lcatm	(Optional) Displays Label Controlled ATM (LC-ATM) events. **Note** This keyword applies to Cisco 7000 series routers only.
ldp	(Optional) Displays Label Distribution Protocol (LDP) events.
vpnv4	(Optional) Displays Virtual Private Network (VPNv4) events.
6pe	(Optional) Displays IPv6 over MPLS events. **Note** This keyword applies to Cisco 7000 series routers only.
table	(Optional) Displays debugging information for one or more routing tables.
all	(Optional) Displays debugging information for all routing tables.
table-id	(Optional) Specifies the routing table for which you want to display debugging information. Table 0 is the default or global routing table.
vrf	(Optional) Displays debugging information for the VPN routing and forwarding (VRF) instance you specify.
vrf-name	(Optional) The name of the VRF instance. You can find VRF names with the show ip vrf command.
acl	(Optional) Displays debugging information for the access control list (ACL) you specify.
acl-name	(Optional) The name of the ACL. You can find ACL names with the show ip access-list command.
prefix-list	(Optional) Displays debugging information for the prefix list you specify.
prefix-list-name	(Optional) The name of the prefix list. You can find prefix list names with the show ip prefix-list command.

Command Default

Debugging is not enabled. If you do not supply an optional keyword, all the debugging events are displayed.

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(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Examples

See the **debug mpls ip iprm** command page for sample output and an explanation of the fields displayed in the output.

Related Commands

Command	Description
debug mpls ip iprm cef	Displays debugging information for interactions between Cisco Express Forwarding and the IPRM.
debug mpls ip iprm events	Displays debugging information about events related to the MPLS IPRM.
debug mpls ip iprm mfi	Displays debugging information for interactions between the MFI and the MPLS IPRM.

debug mpls ip iprm mfi

To display debugging information for interactions between the Multiprotocol Label Switching (MPLS) Forwarding Infrastructure (MFI) and the MPLS IP Rewrite Manager (IPRM), use the debug mpls ip iprm mfi command in privileged EXEC mode. To disable the display of this information, use the **no** form of this command.

debug mpls ip iprm mfi [{**table** {**all***table-id*} | **vrf** *vrf-name* | **acl** *acl-name* | **prefix-list** *prefix-list-name*}]
no debug mpls ip iprm mfi

Syntax Description

table	(Optional) Displays debugging information for one or more routing tables.
all	(Optional) Displays debugging information for all routing tables.
table-id	(Optional) Displays debugging information for the routing table you specify. Table 0 is the default or global routing table.
vrf	(Optional) Displays debugging information for the VPN Routing and Forwarding (VRF) instance you specify.
vrf-name	(Optional) The name of the VRF instance. You can find VRF names with the show ip vrf command.
acl	(Optional) Displays debugging information for the access control list (ACL) you specify.
acl-name	(Optional) The name of the ACL. You can find ACL names with the show ip access-list command.
prefix-list	(Optional) Displays debugging information for the prefix list you specify.
prefix-list-name	(Optional) The name of the prefix list. You can find prefix list names with the show ip prefix-list command.

Command Default

Debugging is not enabled. If you enable debugging but do not supply an optional keyword, all the debugging events are displayed.

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(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Examples

The command in the following example displays MFI events.

Cisco 7500 Series Example

```
Router# debug mpls ip iprm mfi
IPRM MFI interaction debugging is on for global routing table
iprm: delete mfi rewrite: 10.0.0.44/32(glbl)
 .
 .
 .
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois
  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm: omit rewrite create: 10.0.0.44/32(glbl)
 .
 .
 .
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois
  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm:    lcl lab 21, ltbl 0, ldp
  iprm:    path lab -, nh 10.0.0.55(glbl), Et4/0/1; ldp
  iprm: create mfi rewrite 10.0.0.44/32(glbl) passed: 2 fpis/1 mois
  iprm:    fpi[0] IV4, owner IPRM; 10.0.0.44/32; glbl
  iprm:    fpi[1] LBL, owner LDP; 21, ltbl 0
  iprm:    moi[0] PKT, flags 0x8; lab label-no-label; nh 10.0.0.55; nh if Et4/0/1 (nsf)
```

Cisco 10000 Series Example

```
Router# debug mpls ip iprm mfi
IPRM MFI interaction debugging is on for global routing table
iprm: delete mfi rewrite: 10.0.0.44/32(glbl)
 .
 .
 .
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois
  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm: omit rewrite create: 10.0.0.44/32(glbl)
 .
 .
 .
  iprm: get mfi rewrite 10.0.0.44/32(glbl) obtained: 0 fpis/0 mois
  iprm: update mfi rewrite: 10.0.0.44/32(glbl); prefix label info
  iprm:    lcl lab 21, ltbl 0, ldp
  iprm:    path lab -, nh 10.0.0.55(glbl), GigabitEthernet4/0/0; ldp
  iprm: create mfi rewrite 10.0.0.44/32(glbl) passed: 2 fpis/1 mois
  iprm:    fpi[0] IV4, owner IPRM; 10.0.0.44/32; glbl
  iprm:    fpi[1] LBL, owner LDP; 21, ltbl 0
  iprm:    moi[0] PKT, flags 0x8; lab label-no-label; nh 10.0.0.55; nh if 
GigabitEthernet4/0/0 (nsf)
```

See the **debug mpls ip iprm** command page for an explanation of the fields displayed in the output.

Related Commands

Command	Description
debug mpls ip iprm cef	Displays debugging information for interactions between Cisco Express Forwarding and the MPLS IPRM .

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 l2transport checkpoint

To enable the display of Any Transport over MPLS (AToM) events when AToM is configured for nonstop forwarding/stateful switchover (NSF/SSO) and Graceful Restart, use the debug mpls l2transport checkpoint command in privileged EXEC mode. To disable the display of these messages, use the no form of this command.

debug mpls l2transport checkpoint
no debug mpls l2transport checkpoint

Syntax Description

This command has no arguments or keywords.

Command Default

Debugging of the AToM NSF/SSO and Graceful Restart feature is disabled.

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.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.

Usage Guidelines

Use debug commands with care. They use a significant amount of CPU time and can affect system performance.

Examples

In the following example, the output shows that NSF/SSO and Graceful Restart synchronize the data between the active and backup Route Processors after an AToM virtual circuit (VC) is created. (Both the **debug mpls l2transport checkpoint** and the **debug acircuit checkpoint** commands are enabled in this example.)

The debug mpls l2transport checkpoint command is enabled on the active RP:

```
Router# debug mpls l2transport checkpoint
Router# debug acircuit checkpoint
Router# show debug
AToM HA:
  AToM checkpointing events and errors debugging is on
AC HA:
  Attachment Circuit Checkpoint debugging is on
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface Fa5/1/1.2
Router(config-subif)# xconnect 10.55.55.2 1002 pw-class mpls
AToM HA [10.55.55.2, 1002]: Build provision msg, SSM sw/seg 8192/8194 [0x2000/0x2002] PW
id 9216 [0x2400] local label 21
AC HA: Dynamic Sync. Event:4 Sw:8192[2000] Se:16385[4001]
AToM HA: CF sync send complete
AC HA CF: Sync send complete. Code:0
```

On the standby Route Processor, the following messages indicate that it receives checkpointing data:

```
AC HA [10.55.55.2, 1002]: Add to WaitQ. Flags:1
AToM HA [105.55.55.2, 1002]: Received 32-byte provision version 1 CF message
AC HA CF: ClientId:89, Entity:0 Length:40
AToM HA [10.55.55.2, 1002]: Process chkpt msg provision [1], ver 1
AToM HA [10.55.55.2, 1002]: Reserved SSM sw/seg 8192/8194 [0x2000/0x2002] PW id 9216 [0x2400]
AC HA: Process Msg:35586. Ptr:44CBFD90. Val:0
AC HA: Sync. Event:4 CktType:4 Sw:8192[2000] Se:16385[4001]
AC HA [10.55.55.2, 1002]: Remove from WaitQ. Flags:1[OK][OK]
```

During a switchover from the active to the backup Route Processor, the debug messages look similar to the following:

```
%HA-5-MODE: Operating mode is hsa, configured mode is sso.
AC HA RF: CId:83, Seq:710, Sta:RF_STATUS_OPER_REDUNDANCY_MODE_CHANGE, Opr:5, St:STANDBY
HOT, PSt:ACTIVE
AToM HA: CID 84, Seq 715, Status RF_STATUS_OPER_REDUNDANCY_MODE_CHANGE, Op 5, State STANDBY
 HOT, Peer ACTIVE
AC HA RF: CId:83, Seq:710, Sta:RF_STATUS_PEER_PRESENCE, Opr:0, St:STANDBY HOT, PSt:ACTIVE
AToM HA: CID 84, Seq 715, Status RF_STATUS_PEER_PRESENCE, Op 0, State STANDBY HOT, Peer
ACTIVE
AC HA RF: CId:83, Seq:710, Sta:RF_STATUS_PEER_COMM, Opr:0, St:STANDBY HOT, PSt:DISABLED
AToM HA: CID 84, Seq 715, Status RF_STATUS_PEER_COMM, Op 0, State STANDBY HOT, Peer DISABLED
%HA-2-CUTOVER_NOTICE: Cutover initiated. Cease all console activity until system restarts.
%HA-2-CUTOVER_NOTICE: Do not add/remove RSPs or line cards until switchover completes.
%HA-2-CUTOVER_NOTICE: Deinitializing subsystems...
%OIR-6-REMCARD: Card removed from slot 4, interfaces disabled
%OIR-6-REMCARD: Card removed from slot 5, interfaces disabled
%OIR-6-REMCARD: Card removed from slot 9, interfaces disabled
%HA-2-CUTOVER_NOTICE: Reinitializing subsystems...
%HA-2-CUTOVER_NOTICE: System preparing to restart...
%HA-5-NOTICE: Resuming initialization...
AC HA RF: CId:83, Seq:710, Sta:RF_STATUS_REDUNDANCY_MODE_CHANGE, Opr:7, St:STANDBY HOT,
PSt:DISABLED
.
.
.
%LDP-5-GR: LDP restarting gracefully.  Preserving forwarding state for 250 seconds.
AC HA RF: CId:83, Seq:710, Sta:RF_PROG_ACTIVE, Opr:0, St:ACTIVE, PSt:DISABLED
AToM HA: CID 84, Seq 715, Event RF_PROG_ACTIVE, Op 0, State ACTIVE, Peer DISABLED
AC HA: Process Msg:35588. Ptr:0. Val:0
AC HA: Switchover: Standby->Active
AC HA RF: Reconciling
```

Related Commands

Command	Description
debug acircuit checkpoint	Enables the display of AToM attachment circuit events when AToM is configured for NSF/SSO and Graceful Restart.

debug mpls l2transport fast-reroute

To enable the display of Fast Reroute debugging information, use the debug mpls l2transport fast-reroute command in privileged EXEC mode. To stop the display of these messages, use the no form of this command.

debug mpls l2transport fast-reroute commanddebug mpls l2transport fast-reroute
no debug mpls l2transport fast-reroute

Syntax Description

This command has no arguments or keywords.

Command Default

Debugging of the fast reroute feature is not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(25)S	This command was introduced.
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.

Usage Guidelines

This command does not display output on platforms where AToM Fast Reroute is implemented in the forwarding code. This command does not display output for the Cisco 7500 (both RP and VIP) series routers, 7200 series routers, and Cisco 12000 series route processor. The command does display output on Cisco 10720 Internet router line cards and Cisco 12000 series line cards.

Examples

In the following example, the primary link is disabled, which causes the backup tunnel (Tu1) to become the primary path.

```
Router# execute-on slot 3 debug mpls l2transport fast-reroute
========= Line Card (Slot 3) =========
AToM fast reroute debugging is on
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Processing TFIB FRR event for 1.4.0.1
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Finished processing TFIB FRR event for 1.4.0.1
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Processing TFIB FRR event for Tunnel41
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Finished processing TFIB FRR event for Tunnel41
Sep 16 17:58:58.342: %LINK-3-UPDOWN: Interface POS0/0, changed state to down
Sep 16 17:58:58.342: %OSPF-5-ADJCHG: Process 1, Nbr 1.0.0.1 on POS0/0 from FULL to DOWN,
Neighbor Down: Interface down or detached
Sep 16 17:58:59.342: %LINEPROTO-5-UPDOWN: Line protocol on Interface POS0/0, changed state
 to down
```

Related Commands

Command	Description
show mpls traffic-eng fast-reroute database	Displays the contents of the Fast Reroute database.

debug mpls l2transport ipc

To display the interprocessor communication (IPC) messages exchanged between distributed platforms, such as the Cisco 12000 series router and the Cisco 7500 series routers, use the **debug mpls l2transport ipc** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls l2transport ipc
no debug mpls l2transport ipc

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

You can issue this command either from the line card or the route processor to log Any Transport over MPLS (AToM) updates to or from line cards. This command applies only to platforms that support distributed mode.

Examples

The following is sample output from the **debug mpls l2transport ipc** command:

```
Router# debug mpls l2transport ipc
AToM ipc debugging is on
*May 27 23:56:04.699 UTC: AToM SMGR: Repopulating line card 255
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1101]: Sending Imposition update to slot
 255
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1101]: Imposition being done on ingress
interface
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1101]: Sending disposition update to slot
 255
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1101]: Distributing disposition info to
all linecards
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 701]: Sending Imposition update to slot
255
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 701]: Imposition being done on ingress
interface
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 701]: Sending disposition update to slot
 255
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 701]: Distributing disposition info to
all linecards
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1201]: Sending Imposition update to slot
 255
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1201]: Imposition being done on ingress
interface
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1201]: Sending disposition update to slot
 255
```

```
*May 27 23:56:04.699 UTC: AToM SMGR [17.17.17.17, 1201]: Distributing disposition info to
all linecards
```

debug mpls l2transport packet

To display information about the status of Any Transport over MPLS (AToM) switched packets, use the **debug mpls l2transport packet** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls l2transport packet {**data** | **error**}
no debug mpls l2transport packet {**data** | **error**}

Syntax Description

data	Displays (in hex) the AToM switched packets for imposition and disposition. This can help validate that packets are flowing between the customer edge (CE) routers. Also, you can display the packets to check the format of the data or the data itself.
error	Displays AToM switching errors, such as the reason that packets cannot be switched. This can help identify why data is not being transported.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use this command sparingly because the command output can be overwhelming.

For platforms that support distributed switching, the command displays output only for packets switched by the central route processor module. Packets switched autonomously by the linecards are not displayed. For example, packets switched by Versatile Interface Processors (VIPs) on the Cisco 7500 router are not displayed.

Examples

The following is sample output from the **debug mpls l2transport packet** commands for a PPP over MPLS configuration:

```
Router# debug mpls l2transport packet data
AToM packet data debugging is on
Router# debug mpls l2transport packet error
AToM packet errors debugging is on
Router# show debug

AToM:
  AToM packet data debugging is on
  AToM packet errors debugging is on
*Mar 24 23:29:30.495: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:30.495: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:30.495: 0F 00 88 47 00 01 10 FF 00 01 51 02 00 00 00 00
*Mar 24 23:29:30.495: 00 FD C0 01 01 01 C0 4B 41 73 F4 00 01 00 02 CC
*Mar 24 23:29:30.495: 66 51 88 B4 CE 73 39 00 00 40 00 88 03 02 00 70
```

```
*Mar 24 23:29:30.495: 23 30 00 04 3C 61 83 C0 00 06 00 06 94 CC A7 23
*Mar 24 23:29:30.495: 49 84 D8 33 17 8C F2 60 00 11 9E 80 00 50 08 08
*Mar 24 23:29:30.495: 86 69 39 98 CD E2 02 49 B8 E9 9D 0D C6 53 A1 DC
*Mar 24 23:29:30.495: DE 72 35 88 09 E7 0C 60 61 3A 1A 4D C6 71 01 4C
*Mar 24 23:29:30.495: F2 73 CC 06 DC 38 6F 33 66 83 09 C8 CA 20 05 12
*Mar 24 23:29:30.495: 49 E5 31 00 A0 E8 6D 14 88 06 E3 21 80 C3 31 E4
*Mar 24 23:29:30.495: 28 21 E4 21 69 28 A6 2D 26 8A 45 82 02 B6 FC 39
*Mar 24 23:29:30.499: D8 60 A3 62 B1 60 A5 80
*Mar 24 23:29:31.835: ATOM-L2 Switching Disposition Packet data:
*Mar 24 23:29:31.835: FF 03 00 FD C0 04 8A 57 FF FF FF FF FF FF FF FF
*Mar 24 23:29:31.835: FF FF FB 14 B0 00
*Mar 24 23:29:49.423: ATOM-L2 Switching Disposition Packet data:
*Mar 24 23:29:49.423: FF 03 C0 21 01 11 00 0F 03 05 C2 23 05 05 06 5F
*Mar 24 23:29:49.423: 23 35 D4
*Mar 24 23:29:49.435: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:49.435: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:49.435: 0F 00 88 47 00 01 10 FF 00 01 61 02 00 15 00 00
*Mar 24 23:29:49.435: C0 21 01 2F 00 0F 03 05 C2 23 05 05 06 5F CC 5F
*Mar 24 23:29:49.435: E5
*Mar 24 23:29:49.435: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:49.435: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:49.435: 0F 00 88 47 00 01 10 FF 00 01 61 02 00 15 00 00
*Mar 24 23:29:49.435: C0 21 02 11 00 0F 03 05 C2 23 05 05 06 5F 23 35
*Mar 24 23:29:49.435: D4
*Mar 24 23:29:49.443: ATOM-L2 Switching Disposition Packet data:
*Mar 24 23:29:49.443: FF 03 C0 21 02 2F 00 0F 03 05 C2 23 05 05 06 5F
*Mar 24 23:29:49.443: CC 5F E5
*Mar 24 23:29:49.447: ATOM-L2 Switching Disposition Packet data:
*Mar 24 23:29:49.447: FF 03 C2 23 01 D0 00 1C 10 45 59 13 1A 92 FD 93
*Mar 24 23:29:49.447: 01 A2 CF B6 FB 3A 04 46 93 63 65 32 2D 67 73 72
*Mar 24 23:29:49.451: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:49.451: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:49.451: 0F 00 88 47 00 01 10 FF 00 01 61 02 00 22 00 00
*Mar 24 23:29:49.451: C2 23 01 F5 00 1C 10 F1 98 35 3F 79 F2 1A 15 10
*Mar 24 23:29:49.451: B4 C0 73 D7 B1 9F 2A 63 65 31 2D 67 73 72
*Mar 24 23:29:49.455: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:49.455: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:49.455: 0F 00 88 47 00 01 10 FF 00 01 61 02 00 22 00 00
*Mar 24 23:29:49.455: C2 23 02 D0 00 1C 10 56 4A 32 5B 99 55 D5 CF 44
*Mar 24 23:29:49.455: FC D3 D9 3F CC 8C A8 63 65 31 2D 67 73 72
*Mar 24 23:29:49.463: ATOM-L2 Switching Disposition Packet data:
*Mar 24 23:29:49.463: FF 03 C2 23 02 F5 00 1C 10 45 84 E4 E5 DD C0 5F
*Mar 24 23:29:49.463: FD 2F 37 63 9A 3D 03 7B B9 63 65 32 2D 67 73 72
*Mar 24 23:29:49.463: ATOM-L2 Switching Disposition Packet data:
*Mar 24 23:29:49.463: FF 03 C2 23 03 D0 00 04
*Mar 24 23:29:49.471: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:49.471: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:49.471: 0F 00 88 47 00 01 10 FF 00 01 61 02 00 0A 00 00
*Mar 24 23:29:49.471: C2 23 03 F5 00 04
*Mar 24 23:29:49.471: ATOM-PPP Switching: check features failed.
*Mar 24 23:29:49.471: ATOM-PPP Switching (Fast) Imposition Packet data: experimental bits
are 0
*Mar 24 23:29:49.471: 0F 00 88 47 00 01 10 FF 00 01 61 02 00 10 00 00
*Mar 24 23:29:49.471: 80 21 01 0B 00 0A 03 06 78 01 01 78
*Mar 24 23:29:49.475: ATOM-PPP Switching: check features failed.
```

debug mpls l2transport signaling

To display information about the Any Transport over MPLS (AToM) signaling protocol, use the **debug mpls l2transport signaling** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls l2transport signaling {**event** | **message**}
no debug mpls l2transport signaling {**event** | **message**}

Syntax Description

event	Displays AToM signaling events.
message	Displays AToM signaling status messages.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.
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 mpls l2transport signaling** command:

```
Router# debug mpls l2transport signaling event
AToM LDP event debugging is on
Router# debug mpls l2transport signaling message
AToM LDP message debugging is on
Router# show debugging
AToM:
  AToM LDP event debugging is on
  AToM LDP message debugging is on
*Mar 24 23:10:55.611: AToM LDP [9.9.9.9]: Allocate LDP instance
*Mar 24 23:10:55.611: AToM LDP [9.9.9.9]: Opening session, 1 clients
*Mar 24 23:10:56.063: %SYS-5-CONFIG_I: Configured from console by console
*Mar 24 23:10:56.583: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed
state to up
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Session is up
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Peer address change, add 1.1.1.100
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Peer address change, add 46.1.1.6
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Peer address change, add 9.9.9.9
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Peer address change, add 57.1.1.6
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Sending label mapping msg
vc type 7, cbit 1, vc id 50, group id 6, vc label 21, status 0, mtu 1500
*Mar 24 23:11:00.539: AToM LDP [9.9.9.9]: Received label mapping msg, id 113
vc type 7, cbit 1, vc id 50, group id 6, vc label 21, status 0, mtu 1500
```

debug mpls l2transport static-oam

To enable the display of messages related to static pseudowire operations administrative and management (OAM), use the **debug mpls l2transport static-oam** command in privileged EXEC mode. To disable the display of these messages, use the **no** form of this command.

debug mpls l2transport static-oam [{**elog** | **error** | **event** | **fsm**}]
no debug mpls l2transport static-oam

Syntax Description

elog	Displays logging messages for static pseudowire OAM.
error	Displays error messages for static pseudowire OAM.
event	Displays event messages for static pseudowire OAM.
fsm	Displays finite state machine (FSM) messages for static pseudowire OAM.

Command Default

Static sseudowire messages are not displayed.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.1(1)SA	This command was introduced.
15.1(3)S	This command was integrated.

Examples

The following example enables the display of error messages for static pseudowire OAM:

```
Router# debug mpls l2transport static-oam error
```

Related Commands

Command	Description
show mpls l2transport static-oam	Displays the status of static pseudowires.

debug mpls l2transport vc

To display information about the status of the Any Transport over MPLS (AToM) virtual circuits (VCs), use the**debug mpls l2transport vc** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls l2transport vc {**event** | **fsm** | **ldp** | **sss** | **status** {**event** | **fsm**}}
no debug mpls l2transport vc {**event** | **fsm** | **ldp** | **sss** | **status** {**event** | **fsm**}}

Syntax Description

event	Displays AToM event messages about the VCs.
fsm	Displays debug information related to the finite state machine (FSM).
ldp	Displays debug information related to the Label Distribution Protocol (LDP).
sss	Displays debug information related to the subscriber service switch (SSS).
status	Displays debug information related to the status of the VCs.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.
12.0(25)S	This command was integrated into Cisco IOS Release 12.0(25)S.
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.
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.
12.2(33)SRC	The command was updated to include the **ldp**, **sss**, and **status** keywords as part of the MPLS Pseudowire Status Signaling feature.
12.2(50)SY	This command was integrated into Cisco IOS Release 12.2(50)SY.

Usage Guidelines

You can issue this command from the line card or the route processor.

Examples

The following is sample output from the **debug mpls l2transport vc** command:

```
Router# debug mpls l2transport vc event
```

```
AToM vc event debugging is on
Router# debug mpls l2transport vc fsm
AToM vc fsm debugging is on
Router# show debugging

AToM:
  AToM vc event debugging is on
  AToM vc fsm debugging is on
*Mar 24 23:17:24.371: AToM MGR [10.9.9.9, 50]: Event provision, state changed from idle to
 provisioned
*Mar 24 23:17:24.371: AToM MGR [10.9.9.9, 50]: Provision vc
*Mar 24 23:17:24.371: AToM SMGR [10.9.9.9, 50]: Requesting VC create, vc_handle 61A09930
*Mar 24 23:17:24.371: AToM MGR [10.9.9.9, 50]: Event local up, state changed from provisioned
 to local standby
*Mar 24 23:17:24.371: AToM MGR [10.9.9.9, 50]: Update local vc label binding
*Mar 24 23:17:24.371: AToM SMGR [10.9.9.9, 50]: sucessfully processed create request
*Mar 24 23:17:24.875: %SYS-5-CONFIG_I: Configured from console by console
*Mar 24 23:17:25.131: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed
state to up
*Mar 24 23:17:28.567: AToM MGR [10.9.9.9, 50]: Event ldp up, state changed from local standby
 to local ready
*Mar 24 23:17:28.567: AToM MGR [10.9.9.9, 50]: Advertise local vc label binding
*Mar 24 23:17:28.567: AToM MGR [10.9.9.9, 50]: Event remote up, state changed from local
ready to establishing
*Mar 24 23:17:28.567: AToM MGR [10.9.9.9, 50]: Remote end up
*Mar 24 23:17:28.567: AToM MGR [10.9.9.9, 50]: Event remote validated, state changed from
establishing to established
*Mar 24 23:17:28.567: AToM MGR [10.9.9.9, 50]: Validate vc, activating data plane
*Mar 24 23:17:28.567: AToM SMGR [10.9.9.9, 50]: Processing imposition update, vc_handle
61A09930, update_action 3, remote_vc_label 21
*Mar 24 23:17:28.567: AToM SMGR [10.9.9.9, 50]: Imposition Programmed, Output Interface:
PO5/0
*Mar 24 23:17:28.567: AToM SMGR [10.9.9.9, 50]: Processing disposition update, vc_handle
61A09930, update_action 3, local_vc_label 22
*Mar 24 23:17:28.571: AToM SMGR: Processing TFIB event for 10.9.9.9
*Mar 24 23:17:28.571: AToM SMGR [10.9.9.9, 50]: Imposition Programmed, Output Interface:
PO5/0
```

The following is sample output of MPLS Pseudowire Status Signaling messages from the **debug mpls l2transport vc status event and debug mpls l2transport vc status fsm**commands:

```
Router# debug mpls l2transport vc status event
Router# debug mpls l2transport vc status fsm
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: Receive SSS STATUS(UP)
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: AC status UP
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Evt local up, LndRru->LnuRru
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Evt local ready, LnuRru->LruRru
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Act send label(UP)
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: Send label(UP)
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: Local AC : UP
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: Dataplane: no fault
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: Overall  : no fault
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: Remote label is ready
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Evt remote ready in LruRru
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Evt remote up in LruRru
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Evt dataplane clear fault in LruRru
*Feb 26 14:03:42.543: AToM MGR [10.9.9.9, 100]: S:Evt dataplane clear fault in LruRru
*Feb 26 14:03:42.551: AToM MGR [10.9.9.9, 100]: S:Evt dataplane clear fault in LruRru
```

The status codes in the messages, such as S: and LruRru, indicate the staus of the local and remote routers. The following list translates the status codes:

L--local router

R--remote router

r or n--ready (r) or not ready (n)

u or d-- up (u) or down (d) status

The output also includes the following values:

D--Dataplane

S--Local shutdown

debug mpls l2transport vc subscriber

To enable debugging for Any Transport over MPLS (AToM) virtual circuit (VC) subscriber sessions, use the **debug mpls l2transport vc subscriber**command in privileged EXEC mode. To disable debugging for AToM VC subscribers, use the **no** form of this command.

debug l2transport vc subscriber {**error** | **event**}

Syntax Description

error	Specifies debugging for AToM VC subscriber session errors.
event	Specifies debugging for AToM VC subscriber session events.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.1(2)S	This command was introduced.

Usage Guidelines

The **debug l2transport vc subscriber** command displays the debugs for flows involving the creation of AToM VCs as a result of the detection of First Sign of Life (FSOL) events.

Examples

The following is sample output from the **debug l2transport vc subscriber** command:

```
Router# debug mpls l2transport vc subscriber error
AToM LDP subscriber error debugging is on
Router# debug mpls l2transport vc subscriber event
AToM LDP subscriber event debugging is on
Router# show debugging
AToM:
  AToM vc event debugging is on
  AToM LDP subscriber event debugging is on
  AToM LDP subscriber error debugging is on
Router# show logging
Syslog logging: enabled (0 messages dropped, 41 messages rate-limited, 0 flushes, 0 overruns,
 xml disabled, filtering disabled)
No Active Message Discriminator.
No Inactive Message Discriminator.
    Console logging: level debugging, 498 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 229 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled
    Trap logging: level informational, 123 message lines logged
Log Buffer (10000000 bytes):
*Apr 15 20:58:34.048: AToM LDP: Receive label adv
*Apr 15 20:58:34.048: AToM[33.33.33.34, 12346]: Received LDP label msg, signal peer ID
0.0.0.0
*Apr 15 20:58:34.048: AToM[33.33.33.34, 12346]: label details: type 5, label 20
*Apr 15 20:58:34.048: AToM LDP Sub::Found subscriber author group atom_test1, for peer ID
33.33.33.34, VC ID 12346
```

```
*Apr 15 20:58:34.048: AToM LDP Sub::33.33.33.34 created pre-auth key
peer-ip:33.33.33.34:vc-id:12346
*Apr 15 20:58:34.049: AToM LDP Sub::SDB find string/context not found
31:peer-ip:33.33.33.34:vc-id:12346
*Apr 15 20:58:34.049: AToM LDP Sub::creating FSOL context - string
31:peer-ip:33.33.33.34:vc-id:12346, service 0
*Apr 15 20:58:34.049: AToM LDP Sub::SDB add string success 31:peer-ip:33.33.33.34:vc-id:12346,
 0x43EFE12
*Apr 15 20:58:34.049: AToM LDP Sub::Init notify 31:peer-ip:33.33.33.34:vc-id:12346
*Apr 15 20:58:34.049: AToM LDP Sub::SDB get FSOL handle success 0x87E7000, 0x43EFE12
*Apr 15 20:58:34.049: AToM LDP Sub::SDB sanity check success 0x87E7000, 0x83928940
*Apr 15 20:58:34.049: AToM LDP Sub::SDB found string/context
31:peer-ip:33.33.33.34:vc-id:12346, 0x43EFE12
*Apr 15 20:58:34.049: AToM LDP Sub::find/create fsol found fsol after add :
*Apr 15 20:58:34.049: AToM LDP Sub::71237138(0x43EFE12) 31:peer-ip:33.33.33.34:vc-id:12346
*Apr 15 20:58:34.049: AToM LDP Sub::find/create fsol context success
*Apr 15 20:58:34.050: AToM LDP Sub::Preauth request success, handle 0x43EFE12, AAA ID 0x14,
 policy handle 0x5100000C
*Apr 15 20:58:34.050: AToM[33.33.33.34, 12346]: Succeeded to make pre-author request for
ATOM LDP FSOL
*Apr 15 20:58:34.050: AToM[33.33.33.34]: status notification failed: no such vc
*Apr 15 20:58:34.075: AToM LDP Sub::handle 0x43EFE12, AAA attribute 1054
*Apr 15 20:58:34.075: AToM LDP Sub::AAA attribute 1080 not handled
*Apr 15 20:58:34.075: AToM LDP Sub::AAA attribute 968 not handled
*Apr 15 20:58:34.075: AToM LDP Sub::handle 0x43EFE12, AAA attribute 1075
*Apr 15 20:58:34.075: AToM LDP Sub::handle 0x43EFE12, AAA attribute 1056
*Apr 15 20:58:34.075: AToM LDP Sub::handle 0x43EFE12, attribute 1056 val 1
*Apr 15 20:58:34.075: AToM LDP Sub::handle 0x43EFE12, protocol 1
*Apr 15 20:58:34.075: AToM LDP Sub::added 0x8152A00 to 0x370000C7 successfully
*Apr 15 20:58:34.075: AToM LDP Sub::handle 0x43EFE12, Pre-author parser returning 0
*Apr 15 20:58:34.098: AToM LDP Sub::handle 0x43EFE12, AAA attribute 1054
*Apr 15 20:58:34.098: AToM LDP Sub::AAA attribute 1080 not handled
*Apr 15 20:58:34.099: AToM LDP Sub::AAA attribute 968 not handled
*Apr 15 20:58:34.099: AToM LDP Sub::handle 0x43EFE12, AAA attribute 1075
*Apr 15 20:58:34.099: AToM LDP Sub::handle 0x43EFE12, AAA attribute 1056
*Apr 15 20:58:34.099: AToM LDP Sub::handle 0x43EFE12, attribute 1056 val 1
*Apr 15 20:58:34.099: AToM LDP Sub::handle 0x43EFE12, protocol 1
*Apr 15 20:58:34.099: AToM LDP Sub::added 0x81529A0 to 0xB50000D0 successfully
*Apr 15 20:58:34.099: AToM LDP Sub::handle 0x43EFE12, Pre-author parser returning 0
*Apr 15 20:58:34.099: AToM LDP Sub::Preauth callback for client 0x43EFE12, AAA 0x14
*Apr 15 20:58:34.100: AToM LDP Sub::SDB get FSOL handle success 0x87E7000, 0x43EFE12
*Apr 15 20:58:34.100: AToM LDP Sub::SDB sanity check success 0x87E7000, 0x83928940
*Apr 15 20:58:34.100: AToM LDP Sub::atom preauth callback 0. processing info 108
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14, processing info type
108, val 0x81529A0, list 0xB50000D0
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14, attribute 1054 12346 len
 5, VC ID 12346
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14 attribute 1075 len 555819298
 33.33.33.34
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14, attribute 1056 val 1
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14, method 3, protocol 4
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14 connect notifyfor VPWS
service
*Apr 15 20:58:34.100: AToM LDP Sub::handle 0x43EFE12, AAA ID 0x14, added member to provision
 VPWS service
*Apr 15 20:58:34.100: AToM LDP Sub::downloaded attribute parsing success, handle 0x43EFE12,
 AAA ID 0x14
*Apr 15 20:58:34.124: AToM LDP Sub::
atom_ldp_subscriber_parse_preauth1char = p:31
*Apr 15 20:58:34.124: AToM LDP Sub::
atom_ldp_subscriber_parse_preauth2char = 3:23
*Apr 15 20:58:34.124: AToM LDP Sub::
atom_ldp_subscriber_parse_preauth3char = v:11:11
*Apr 15 20:58:34.124: AToM LDP Sub::
```

```
atom_ldp_subscriber_parse_preauth4char = 33.33.33.34
*Apr 15 20:58:34.124: AToM LDP Sub::0.0.0.0: parsed 33.33.33.34
*Apr 15 20:58:34.124: AToM LDP Sub::
atom_ldp_subscriber_parse_preauth5char = v:11
*Apr 15 20:58:34.124: AToM LDP Sub::
atom_ldp_subscriber_parse_preauth6char = 1:5
*Apr 15 20:58:34.124: AToM LDP Sub::
atom_ldp_subscriber_parse_preauth7char = 12346
*Apr 15 20:58:34.124: AToM LDP Sub::33.33.33.34: parsed VC ID 12346
*Apr 15 20:58:34.124: AToM LDP Sub::Found subscriber author group atom_test1, for peer ID
33.33.33.34, VC ID 12346
*Apr 15 20:58:34.124: AToM LDP Sub::SDB get FSOL handle success 0x87E7000, 0x43EFE12
*Apr 15 20:58:34.124: AToM LDP Sub::SDB sanity check success 0x87E7000, 0x83928940
*Apr 15 20:58:34.124: AToM LDP Sub::SDB found string/context
31:peer-ip:33.33.33.34:vc-id:12346, 0x43EFE12
*Apr 15 20:58:34.125: AToM LDP Sub::found context from earlier trigger, 1, 0, ignoring this
 request
*Apr 15 20:58:34.125: AToM LDP Sub::Found existing FSOL context ID
31:peer-ip:33.33.33.34:vc-id:12346, re-use 1, 0
*Apr 15 20:58:34.156: AToM[33.33.33.34, 12346]: Provisioned
*Apr 15 20:58:34.156: AToM[33.33.33.34, 12346]: Evt provision, idle -> provisioned
*Apr 15 20:58:34.156: AToM[33.33.33.34, 12346]: . Provision vc
*Apr 15 20:58:34.156: AToM LDP[33.33.33.34, 12346]: LDP OPEN request
*Apr 15 20:58:34.156: AToM LDP[33.33.33.34, 12346]: Signaling peer-id of VC changed to
33.33.33.34
*Apr 15 20:58:34.156: AToM[33.33.33.34, 12346]: Evt remote ready, provisioned -> remote
ready
*Apr 15 20:58:34.156: AToM[33.33.33.34, 12346]: . Receive remote vc label binding, instance
 3
*Apr 15 20:58:34.160: AToM[33.33.33.34, 12346]: Receive SSS CONNECT
*Apr 15 20:58:34.160: AToM[33.33.33.34, 12346]: . Update AIE peer 9F00000F our 43000010
*Apr 15 20:58:34.161: AToM[33.33.33.34, 12346]: ... Evt local ready, remote ready ->
establishing
*Apr 15 20:58:34.161: AToM[33.33.33.34, 12346]: ..... Alloc local binding
*Apr 15 20:58:34.161: AToM[33.33.33.34, 12346]: ...... autosense disabled [init]
*Apr 15 20:58:34.161: AToM[33.33.33.34, 12346]: ...... autosense enabled
*Apr 15 20:58:34.161: AToM[33.33.33.34, 12346]: ...... Grouping on (value 1)
*Apr 15 20:58:34.161: AToM[33.33.33.34, 12346]: ...... Grouping ignored, set to 0
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: ...... MTU set to 1500
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: .... Local end available
*Apr 15 20:58:34.162: AToM LDP[33.33.33.34, 12346]: Send label(DOWN)
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: .... Validate remote binding
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: ..... Evt remote validated in establishing
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: ...... Validate vc, activating data plane
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: ....... Update peer with our circuit type
Eth
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: ....... Check if can activate dataplane
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: .. Check if can activate dataplane
*Apr 15 20:58:34.162: AToM[33.33.33.34, 12346]: . Send SSS CONNECTED
*Apr 15 20:58:34.163: AToM[33.33.33.34, 12346]: ... No PW Status SP info to report to SSS
peer
*Apr 15 20:58:34.163: AToM LDP[33.33.33.34, 12346]: LDP open
*Apr 15 20:58:34.163: AToM LDP[33.33.33.34, 12346]: Signaling peer-id of VC changed to
33.33.33.34
*Apr 15 20:58:34.163: AToM LDP[33.33.33.34, 12346]: LDP UP
*Apr 15 20:58:34.163: AToM[33.33.33.34, 12346]: Evt ldp up in establishing
*Apr 15 20:58:34.163: AToM[33.33.33.34, 12346]: . Take no action
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: Receive SSS FSP STATUS
*Apr 15 20:58:34.169: AToM LDP[33.33.33.34, 12346]: Send notify(UP)
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: ... Evt local ready in establishing
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: .... Take no action
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: .. Check if can activate dataplane
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: ... Attempt to activate dataplane on Active
 RP, VC in establishing state
```

```
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: ... Evt dataplane activate, establishing
-> activating
*Apr 15 20:58:34.169: AToM[33.33.33.34, 12346]: .... Activating data plane
*Apr 15 20:58:34.170: AToM[33.33.33.34, 12346]: .... Activate dataplane
*Apr 15 20:58:34.170: AToM[33.33.33.34, 12346]: .... Need to setup the dataplane
*Apr 15 20:58:34.170: AToM[33.33.33.34, 12346]: .... Setup dataplane
*Apr 15 20:58:34.170: AToM[33.33.33.34, 12346]: ...... Same peer; get switch hdl 4100
*Apr 15 20:58:34.170: AToM[33.33.33.34, 12346]: ..... Set segment count to 1
*Apr 15 20:58:34.170: AToM[33.33.33.34, 12346]: ..... Provision SSM with 4100/8203 (sw/seg)
*Apr 15 20:58:34.173: AToM[33.33.33.34, 12346]: Receive SSM dataplane up notification
*Apr 15 20:58:34.174: AToM[33.33.33.34, 12346]: Receive SSM dataplane up notification
*Apr 15 20:58:34.174: AToM[33.33.33.34, 12346]: Evt dataplane up, activating -> established
*Apr 15 20:58:34.174: AToM[33.33.33.34, 12346]: . Dataplane activated
*Apr 15 20:58:34.174: AToM[33.33.33.34, 12346]: SYSLOG: VC is UP
*Apr 15 20:58:34.174: AToM[33.33.33.34, 12346]: Evt dataplane up in established
*Apr 15 20:58:34.174: AToM[33.33.33.34, 12346]: . Take no action
*Apr 15 20:58:42.222: AToM[33.33.33.34, 12346]: Label 23 freed
```

Related Commands

Command	Description
show debugging	Displays information about the types of debugging that are enabled for your router.
show logging	Displays the state of system logging (syslog) and the contents of the standard system logging buffer.
show mpls l2transport vc	Displays information about the status of the AToM VCs.

debug mpls l2transport vc vccv

To enable Any Transport over MPLS (AToM) Virtual Circuit Connection Verification (VCCV) debugging, use the **debug mpls l2transport vc vccv**command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug mpls l2transport vc vccv [**bfd**] **event**
no debug mpls l2transport vc vccv [**bfd**] **event**

Syntax Description

bfd	(Optional) Displays event messages when Bidirectional Forwarding Detection (BFD) sessions are created, when BFD sends dataplane fault notifications to Layer 2 VPN (L2VPN), and when L2VPN sends the attachment circuit (AC) signaling status to BFD.
event	Displays AToM event messages about the VCCV.

Command Modes

Privileged EXEC (#)

Command History

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

Usage Guidelines

Use this command to enable AToM VCCV events and AToM VCCV BFD events debugging.

Examples

The following examples show how to enable MPLS L2transport VC VCCV and VCCV BFD event debugging:

```
Router# debug mpls l2transport vc vccv bfd event
AToM VCCV BFD events debugging is on
Router# debug mpls l2transport vc vccv event
AToM VCCV events debugging is on
Router# show debugging
AToM VCCV BFD events debugging is on
AToM VCCV events debugging is on
```

Related Commands

Command	Description
show mpls l2transport vc	Displays information about the status of the AToM VCs.

debug mpls ldp advertisements

To display information about the advertisement of labels and interface addresses to label distribution protocol (LDP) peers, use the **debugmplsldpadvertisements**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls ldp advertisements [**peer-acl** *acl*] [**prefix-acl** *acl*]
no debug mpls ldp advertisements [**peer-acl** *acl*] [**prefix-acl** *acl*]

Syntax Description

peer-acl *acl*	(Optional) Limits the displayed advertisements to those for LDP peers permitted by the access control list (*acl*).
prefix-acl *acl*	(Optional) Limits the displayed advertisements to those for prefixes permitted by the access control list (*acl*).

Command Default

Displays information about advertisements to all LDP peers for all prefixes.

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

Use this command to monitor the label and address advertisements to LDP peers.

Use the **peer-acl** or **prefix-acl** options separately or together to limit the information display to specific LDP peers and/or specific prefixes.

Note This command monitors advertisement of non-LC-ATM labels (generic labels) only. Use the **debugmplsatm-ldp** command to monitor LC-ATM activity.

Examples

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

```
Router# debug mpls ldp advertisements
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 130.77.0.33
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 133.0.0.33
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 34.0.0.33
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 103.0.0.33
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 35.0.0.33
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 38.0.0.33
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 34.0.0.0/8, label 3 (#2)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 203.0.7.7/32, label 24 (#4)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 35.0.0.0/8, label 3 (#8)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 103.0.0.0/8, label 3 (#10)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 138.1.0.0/16, label 26 (#14)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 155.0.0.55/32, label 27 (#16)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 38.0.0.0/8, label 3 (#18)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 212.10.1.0/24, label 30 (#24)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 59.0.0.0/8, label 32 (#28)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 144.0.0.44/32, label 33 (#30)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 106.0.0.0/8, label 34 (#32)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 133.0.0.33/32, label 3 (#34)
tagcon: peer 144.0.0.44:0 (pp 0x60E105BC): advertise 45.0.0.0/8, label 39 (#36)
```

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

Table 7: debug mpls ldp advertisements Field Descriptions

Field	Description
tagcon:	Identifies the source of the message as the label control subsystem.
peer a.b.c.d:e	LDP identifier of the peer to which the advertisement was targeted.
(pp 0xnnnnnnnn)	Identifier for the data structure used to represent the peer at the label distribution level. Useful for correlating debug output.
advertise X	Identifies what was advertised to the peer--either an interface address (“a.b.c.d”) or label binding (“a.b.c.d/m, label t (#n)”).
(#n)	For a label binding advertisement, the sequence number of the label information base (LIB) modification that made it necessary to advertise the label.

Related Commands

Command	Description
debug mpls ldp bindings	Displays information about changes to the LIB used to keep track of label bindings learned from LDP peers through LDP downstream label distribution.

Command	Description
show mpls ip binding	Displays specified information about label bindings learned by LDP.
show mpls ldp neighbor	Displays the status of LDP sessions.

debug mpls ldp backoff

To display information about the label distribution protocol (LDP) backoff mechanism parameters, use the **debugmplsldpbackoff**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls ldp backoff
no debug mpls ldp backoff

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.
12.1(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.
12.0(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.

Usage Guidelines

Use this command to monitor backoff parameters configured for LDP sessions.

Examples

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

```
Router# debug mpls ldp backoff

LDP session establishment backoff debugging is on
Router#
Jan 6 22:31:13.012: ldp: Backoff peer ok: 12.12.12.12:0; backing off; threshold/count 8/6
Jan 6 22:31:13.824: ldp: Backoff peer ok: 12.12.12.12:1; backing off; threshold/count 8/6
Jan 6 22:31:17.848: ldp: Backoff peer ok: 12.12.12.12:0; backing off; threshold/count 8/6
Jan 6 22:31:18.220: ldp: Backoff peer ok: 12.12.12.12:1; backing off; threshold/count 8/6
Jan 6 22:31:21.908: ldp: Backoff peer ok: 12.12.12.12:0; backing off; threshold/count 8/6
Jan 6 22:31:22.980: ldp: Backoff peer ok: 12.12.12.12:1; backing off; threshold/count 8/6
Jan 6 22:31:25.724: ldp: Backoff peer ok: 12.12.12.12:0; backing off; threshold/count 8/7
Jan 6 22:31:26.944: ldp: Backoff peer ok: 12.12.12.12:1; backing off; threshold/count 8/7
Jan 6 22:31:30.140: ldp: Backoff peer ok: 12.12.12.12:0; backing off; threshold/count 8/7
Jan 6 22:31:31.932: ldp: Backoff peer ok: 12.12.12.12:1; backing off; threshold/count 8/7
Jan 6 22:31:35.028: ldp: Backoff peer ok: 12.12.12.12:0; backing off; threshold/count 8/7
Jan 6 22:31:35.788: ldp: Backoff peer ok: 12.12.12.12:1; backing off; threshold/count 8/7
Jan 6 22:31:39.332: ldp: Update backoff rec: 12.12.12.12:0, threshold = 8, tbl ents 2
Jan 6 22:31:39.640: ldp: Update backoff rec: 12.12.12.12:1, threshold = 8, tbl ents 2
```

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

Table 8: debug mpls ldp backoff Field Descriptions

Field	Description
ldp	Identifies the Label Distribution Protocol.
Backoff peer ok: a.b.c.d:n	Identifies the LDP peer for which a session is being delayed because of a failure to establish a session due to incompatible configuration.
backing off;	Indicates that a session setup attempt failed and the LSR is delaying its next attempt (that is, is backing off).
threshold/count x/y	Identifies a set threshold (x) and a count (y) that represents the time that has passed since the last attempt to set up a session with the peer. The count is incremented every 15 seconds until it reaches the threshold. When the count equals the threshold, a fresh attempt is made to set up an LDP session with the peer.
Update backoff rec	Indicates that the backoff period is over and that it is time for another attempt to set up an LDP session.
threshold = x	Indicates the backoff time of x*15 seconds, for the next LDP session attempt with the peer.
tbl ents 2	Indicates unsuccessful attempts to set up an LDP session with two different LDP peers. In this example, attempts to set up sessions with LDP peers 12.12.12.12:0 and 12.12.12.12:1 are failing.

Related Commands

Command	Description
mpls ldp backoff	Configures session setup delay parameters for the LDP backoff mechanism.
show mpls ldp backoff	Displays information about the configured session setup backoff parameters and any potential LDP peers with which session setup attempts are being throttled.

debug mpls ldp bindings

To display information about addresses and label bindings learned from Label Distribution Protocol (LDP) peers by means of LDP downstream unsolicited label distribution, use the **debugmplsldpbindings**command in privileged EXEC mode. To disable this feature, use the no form of this command.

debug mpls ldp bindings [**filter**] [**peer-acl** *acl*] [**prefix-acl** *acl*]
no debug mpls ldp bindings [**filter**] [**peer-acl** *acl*] [**prefix-acl** *acl*]

Syntax Description

filter	(Optional) Display information about LDP local label allocation filtering.
peer-acl *acl*	(Optional) Limits the displayed binding information to that learned from LDP peers permitted by the access control list (*acl*).
prefix-acl *acl*	(Optional) Limits the displayed binding information to that learned for prefixes permitted by the access control list (*acl*).

Command Default

Displays information about all bindings learned from all LDP peers.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
11.1CT	This command was introduced.
12.0(10)ST	This command was modified to correspond to MPLS Internet Engineering Task Force (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.

Release	Modification
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)SRC	The **filter** keyword was added and the output of the command was updated to display information about LDP local label allocation filtering.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines

Use this command to monitor label bindings and label switch router (LSR) addresses learned from LDP peers.

Note This command monitors non-LC-ATM labels (generic labels) only. Use the debug mpls atm-ldp states command to monitor LC-ATM activity.

Examples

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

```
Router# debug mpls ldp bindings
tagcon:tibent(10.34.0.0/8):created; find route tags request
tagcon:tibent(10.34.0.0/8):label 3 (#2) assigned
tagcon:tibent(10.0.7.7/32):created; find route tags request
tagcon:tibent(10.0.7.7/32):label 24 (#4) assigned
tagcon:tibent(10.0.0.44/32):created; find route tags request
tagcon:tibent(10.0.0.44/32):label 33 (#30) assigned
tagcon:tibent(10.106.0.0/8):created; find route tags request
tagcon:tibent(10.106.0.0/8):label 34 (#32) assigned
tagcon:tibent(10.0.0.33/32):created; find route tags request
tagcon:tibent(10.0.0.33/32):label 3 (#34) assigned
tagcon:tibent(10.45.0.0/8):created; find route tags request
tagcon:tibent(10.45.0.0/8):label 39 (#36) assigned
tagcon:Assign peer id; 10.0.0.44:0:id 0
tagcon:10.0.0.44:0:10.0.0.44 added to addr<->ldp ident map
tagcon:10.0.0.44:0:10.34.0.44 added to addr<->ldp ident map
tagcon:10.0.0.44:0:10.45.0.44 added to addr<->ldp ident map
tagcon:tibent(10.0.0.44/32):rem label 3 from 10.0.0.44:0 added
tagcon:tibent(10.34.0.0/8):label 3 from 10.0.0.44:0 added
tagcon:tibent(10.45.0.0/8):label 3 from 10.0.0.44:0 added
tagcon:tibent(10.107.0.0/8):created; remote label learned
tagcon:tibent(10.107.0.0/8):label 55 from 10.0.0.44:0 added
tagcon:tibent(10.0.7.7/32):label 209 from 10.0.0.44:0 added
tagcon:tibent(10.0.0.33/32):label 207 from 10.0.0.44:0 added
```

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

Table 9: debug mpls ldp bindings Field Descriptions

Field	Description
tagcon:	Identifies the source of the message as the label control subsystem.
tibent(network/mask)	Destination that has a label binding change.
created; reason	An LIB entry has been created for the specified destination for the indicated reason.

Field	Description
rem label ...	Describes a change to the label bindings for the specified destination. The change is for a label binding learned from the specified LDP peer.
lcl label ...	Describes a change to a locally assigned (incoming) label for the specified destination.
(#n)	Sequence number of the modification to the LIB corresponding to the local label change.
a.b.c.d:n: e.f.g.h added to addr<->ldp ident map	The address e.f.g.h has been added to the set of addresses associated with LDP identifier a.b.c.d:n.

The following is output from the **debugmplsldpbindings**command when LDP local label allocation filtering is configured:

```
Router# debug mpls ldp
%SYS-5-CONFIG_I: Configured from console by console
Router# debug mpls ldp bindings filter
LDP Local Label Filtering changes debugging is on
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# mpls ldp label
Router(config-ldp-lbl)# allocate global host-routes
Router(config-ldp-lbl)#
 LDP LLAF: Enqueued work item to walk tib for all tables
 LDP LLAF: Withdraw local label for 10.10.7.0
 LDP LLAF: Withdraw local label for 10.10.8.0
 LDP LLAF: Withdraw local label for 10.10.9.0
Router(config-ldp-lbl)#
 LDP LLAF: announce zero local and path labels: 10.10.7.0
 LDP LLAF: announce zero local and path labels: 10.10.8.0
 LDP LLAF: announce zero local and path labels: 10.10.9.0
Router(config-ldp-lbl)#
Router(config-ldp-lbl)# no allocate global host-routes

Router(config-ldp-lbl)#
 LDP LLAF: Enqueued work item to walk tib for all tables
 tib: get path labels: 10.1.1.1/32, tableid: 0, Et1/0, nh 10.10.7.2
 LDP LLAF: 10.1.1.1 accepted, absence of filtering config
 tagcon: announce labels for: 10.1.1.1/32; nh 10.10.7.2, Et1/0, inlabel 17, outlabel imp-null
 (from 10.1.1.1:0), get path labels
 tib: get path labels: 10.2.2.2/32, tableid: 0, Et2/0, nh 10.10.8.2
 LDP LLAF: 10.2.2.2 accepted, absence of filtering config
 tagcon: announce labels for: 10.2.2.2/32; nh 10.10.8.2, Et2/0, inlabel 16, outlabel imp-null
 (from 10.2.2.2:0), get path labels
 tib: get path labels: 10.10.7.0/24, tableid: 0, Et1/0, nh 0.0.0.0
 LDP LLAF: 10.10.7.0 accepted, absence of filtering config
 tagcon: tibent(10.10.7.0/24): label 1 (#20) assigned
 tagcon: announce labels for: 10.10.7.0/24; nh 0.0.0.0, Et1/0, inlabel imp-null, outlabel
unknown (from 0.0.0.0:0), get path labels
 tib: get path labels: 10.10.8.0/24, tableid: 0, Et2/0, nh 0.0.0.0
 LDP LLAF: 10.10.8.0 accepted, absence of filtering config
 tagcon: tibent(10.10.8.0/24): label 1 (#21) assigned
 tagcon: announce labels for: 10.10.8.0/24; nh 0.0.0.0, Et2/0, inlabel imp-null, outlabel
unknown (from 0.0.0.0:0), get path labels
 tib: get path labels: 10.10.9.0/24, tableid: 0, Et1/0, nh 10.10.7.2
 LDP LLAF: 10.10.9.0 accepted, absence of filtering config
 tagcon: tibent(10.10.9.0/24): label 22 (#22) assigned
 tagcon: announce labels for: 10.10.9.0/24; nh 10.10.7.2, Et1/0, inlabel 22, outlabel
imp-null (from 10.1.1.1:0), get path labels
.
.
.
Router(config-ldp-lbl)# no mpls ldp label
Router(config-ldp-lbl)# end
Router# no debug mpls ldp bindings filter
```

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

Table 10: debug mpls ldp bindings Field Descriptions with LDP Local Label Allocation Filtering

Field	Description
LDP LLAF	Indicates that the messages apply to LDP local label allocation filtering.
Withdraw local label for 10.10.7.0	Prefix 10.10.7.0 is not in the global routing table. LDP withdraws the label and does not assign a local labels.
announce zero local and path labels: 10.10.7.0	LDP does not announce local and path label for prefix 10.10.7.0.
tagcon: announce labels for:	The label control subsystem announces the next hop (nh) and labels for the named prefix.
tib: get path labels:	LDP LIB searches for the routing and forwarding path for the named prefix.
LDP LLAF: 10.1.1.1 accepted;	LDP accepts the prefix. The prefix was found in the global table (or accepted by the prefix list, if a prefix list was named as a filter).
tibent(network/mask)	Destination that has a label binding change.

Related Commands

Command	Description
debug mpls atm-ldp states	Displays information about label virtual circuit (lvc) state transitions as they occur.
show mpls ldp bindings	Displays the contents of the LIB.