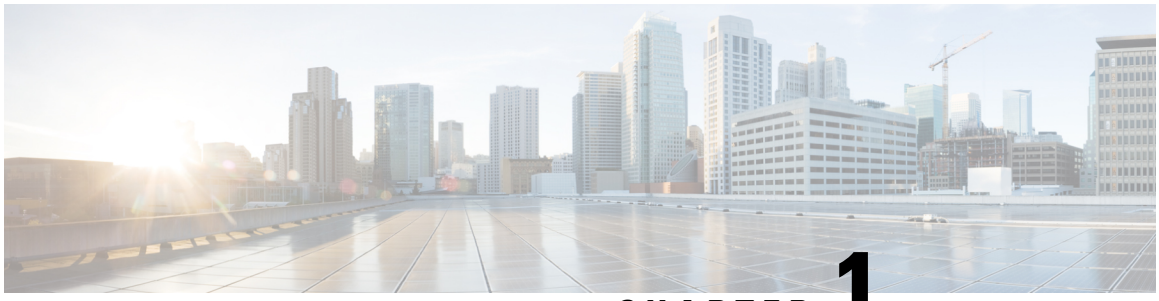
Cisco cBR Converged Broadband Routers Layer 2 and Layer 3 VPN Configuration Guide for Cisco IOS XE Amsterdam 17.3.x

First Published: 2020-09-30

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

L2VPN Support over Cable

The Layer 2 VPN (L2VPN) Support over Cable feature on the Cisco CMTS provides point-to-point Transparent LAN Service (TLS) in support of the Business Services over DOCSIS (BSOD) Cable Labs specification.

The L2VPN Support over Cable feature supports the following:

- The feature uses an Ethernet trunking interface to transport traffic for multiple L2VPNtunnels in support of different cable modems (CMs) and service flows (SFs) based on IEEE 802.1qVLAN IDs. For the legacy TLS service, only the primary upstream or downstream SFs are used. With the new L2VPNSupport over Cable feature, both primary and secondary SFs can be used.
- The TLS feature uses CLI to provision the service. The L2VPN Support over Cable feature uses the CM configuration file to provision the service, and a single CLI to identify the default Ethernet Network System Interface (NSI).
- Downstream traffic is forwarded on a per-CM basis and upstream traffic is forwarded on a per-SF basis. For L2VPN Support over Cable feature, upstream traffic for the same L2VPN can use multiple upstream service flows and downstream traffic can use different downstream service flows.

Finding Feature Information

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 1: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Prerequisites for L2VPN Support over Cable

- You should use crypto-supported images.
- Cable modems must be configured to support BPI+.

Restrictions for L2VPN Support over Cable

The L2VPN Support over Cable feature has the following general restrictions:

- DOCSIS 1.0 CMs are not supported.
- Load balancing and Dynamic Channel Change (DCC) are not supported for CMs that are enabled for L2VPN support.
- DSx messages (Dynamic Service Add [DSA], Dynamic Service Change [DSC], and Dynamic Service Delete [DSD]) are supported for L2VPN-provisioned CMs. However, DSx with L2VPN type, length, values (TLVs) are not supported.
- Multipoint L2VPN is not supported, and any Simple Network Management Protocol (SNMP) MIBs for multipoint L2VPN are not supported.
- eSAFE (embedded Service/Application Functional Entities) DHCP snooping is not supported (L2VPN subtype 43.5.3)
- Maximum of 1024 L2VPNs are supported on a single MAC domain.
- Maximum of eight upstream SFs are supported per L2VPN service.
- Maximum of eight downstream classifiers are supported per L2VPN service.
- eSAFE exclusion is supported for only one eSAFE host. If the REG-REQ message for a compliant CM specifies multiple eSAFE hosts, then the eMTA (ifIndex 16) is selected as the eSAFE host to be excluded by the Cisco CMTS router. If the eMTA is not included as part of the capability of the CM, then the first eSAFE host in the capability is selected for exclusion.
- Maximum length of the Cable Modem Interface Mask (CMIM) is 4 bytes.
- Areas of the Business Services over DOCSIS (BSOD) Layer 2 Virtual Private Networks specification that are not supported are:
    - Vendor-specific L2VPN encodings for the replacement of the required VPN ID and NSI Encapsulation subtype are not supported.
    - Mapping of egress user priority to an NSI port transmission traffic class as specified by IEEE 802.1s is not supported.
    - Forwarding with non-zero default user priority values with vendor-specific configuration is not supported.
    - Accepting multiple Downstream Classifier L2VPN Encoding with the same VPN ID to clasify packets to different service flows is not supported.
    - Assigning multiple SAIDs to the same L2VPN on the same CM is not supported. The primary SAID is used for encrypting all downstream traffic.
    - Assigning of the same group-level L2VPN SAID to different CMs on the same MAC domain attached to the same L2VPN identifier is not supported.
    - Implementing the DOCSIS Spanning Tree Protocol (DSTP) and transmission of DSTP BPDUs on all NSI and RF interfaces configured for L2VPN operation is not supported.
    - Implementing a DSTP SAID specifically for DSTP forwarding to the customer premises equipment (CPE) ports of all L2VPN CMs is not supported.

- dot1q L2VPN is not supported over a port-channel with load-balancing vlan configured.

VPN ID Restrictions

- A maximum of four VPN IDs are supported for each CM.
- A maximum of one VPN ID can be associated with each SF in a CM; although multiple SFs in a CM can belong to the same L2VPN.
- A maximum of 4093 unique VPN IDs are supported per Cisco CMTS router.
- The maximum length of a VPN ID is 16 bytes.
- All L2VPN encodings must contain a VPN ID, except for upstream classifier encodings.

Information About L2VPN Support over Cable

L2VPN Support Over Cable provides the following benefits and functions on a Cisco CMTS router:

- Supports point-to-point L2VPN forwarding mode.
- Supports up to four VPN IDs per CM.
- Supports multiple upstream SFs per CM, with one or more SFs belonging to the same VPN ID.
- Supports a single Ethernet NSI that serves as a trunking port for one or more L2VPN tunnels on the Cisco CMTS router.
- Supports BPI+ encryption using primary SAID of the CM.
- Supports L2VPN encodings in the CM configuration file and CM registration (REG-REQ with L2VPN encoding).
- Supports upstream L2VPN tunnel in support of per-CM and per-SF forwarding.
- Supports synchronization and recovery of the L2VPN database and upstream and downstream SFs during SUP NSF/SSO and N+1 line card redundancy switchovers.
- Supports QoS in upstream and downstream.
- Supports stacked IEEE 802.1q tags.
- Supports exclusion of traffic from the L2VPN tunnel for a single Embedded Service/Application Functional Entity (eSAFE) host.
- Supports Layer 2 classifier via CMIM and IEEE 802.1p priority bits.
- Supports detection of provisioning errors, such as duplicate VLAN IDs across CMs or existing VLAN IDs in use, and moves a CM offline with a corresponding error message.
- Supports coexistence of L2VPN and non-L2VPN traffic on the same RF MAC domain, with non-L2VPN traffic isolated from other tunnel traffic.
- Supports voice calls from L2VPN-provisioned CMs. However, voice calls are not part of the L2VPN.
- Supports BSOD VLAN Redundancy feature, which allows users to configure a backup WAN interface in addition to the primary WAN interface. When the primary WAN interface is down, the L2VPN traffic flows through the backup WAN interface.
- Supports manual switchover for VLAN Redundancy feature, which allows users to manually switch active uplink port from the current port to another port when both the uplink ports are up.
- Supports 2000 bytes layer 2 MTU.

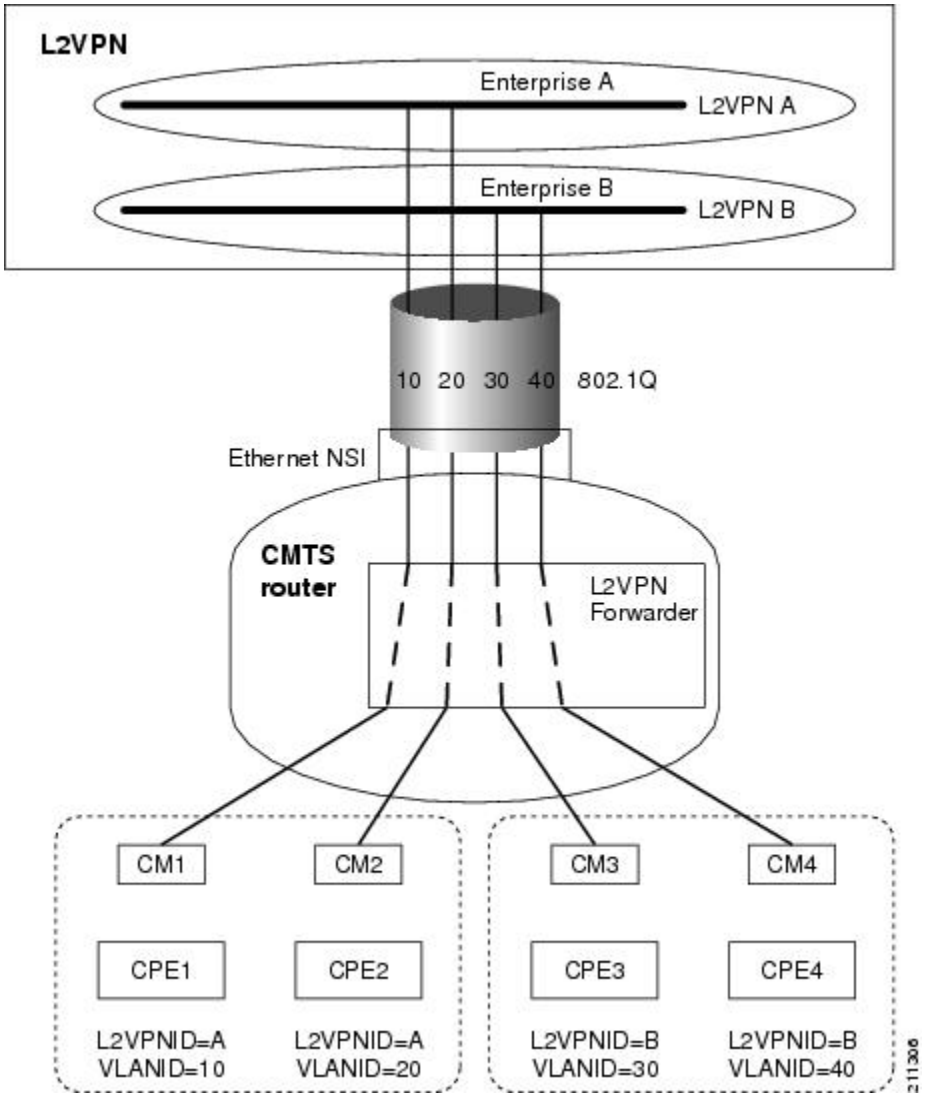
Point-to-Point L2VPN Forwarding Mode

The Cisco CMTS routers supports the point-to-point L2VPN forwarding mode described in the BSOD specification. Each attachment circuit (either SF or CM) on the Cisco CMTS router has a NSI encapsulation value, and is configured with an IEEE 802.1q VLAN ID.

The L2VPN forwarder on the Cisco CMTS router forwards both upstream and downstream traffic between the NSI port on the router and an attachment circuit without using MAC address learning for the forwarding decision. A L2VPN bridge on the backbone network of the cable operator performs the MAC-address learning to bridge packets between VLAN IDs.

The image below shows an example of a point-to-point L2VPN network using IEEE 802.1q NSI encapsulation. In this example, four CMs are associated with four different VLAN IDs: 10, 20, 30, and 40. The L2VPN encoding of the CM includes the logical L2VPN ID (in this case, A or B) with an NSI encapsulation subtype for IEEE 802.1q with the associated VLAN ID.

Figure 1: Point-to-Point L2VPN Network Diagram



The logical L2VPN IDs allow creation of separate broadcast domains for certain VLAN IDs. In the diagram, traffic for VLANs 10 and 20 from CM1 and CM2 can be sent to the network of Enterprise A, and traffic for VLAN's 30 and 40 from CM3 and CM4 can be sent to the network of Enterprise B.

L2VPN Encodings in the CM Configuration File

The CM configuration file contains a set of L2VPN encodings that control how the Cisco CMTS processes L2VPN forwarding of upstream and downstream CPE packets. As per the BSOD specification, the L2VPN encoding is encapsulated using a General Extension Information (GEI) encoding, which uses the type code 43 and subtype of 5 (43.5) with the reserved Vendor ID of 0xFFFFFF.

L2VPN defines the following types of encodings:

- Per-CM L2VPN encodings—An encoding that appears at the top level of the CM configuration file.
- Per-SF L2VPN Encoding—An encoding that appears as a subtype of the Upstream Service Flow Encoding (type 24).
- Upstream Classifier L2VPN Encoding—An encoding that appears in an Upstream Packet Classification Configuration Setting (type 22).
- Downstream Classifier L2VPN Encoding—An encoding that appears in a Downstream Packet Classification Configuration Setting (type 23).

The simplest CM configuration file has a single per-SF L2VPN Encoding within the primary upstream SF definition and a single per-CM L2VPN Encoding with a NSI Encapsulation subtype for that L2VPN.

Note When BSOD (CM configuration file) is used for L2VPN configuration, and QoS policy-map settings are applied to Cisco CMTS WAN interfaces, the packets do not match the QoS policy-map. When CLI mode is used for L2VPN configuration, and QoS policy-map settings are applied to Cisco CMTS WAN interfaces, the packets will match the QoS policy-map first.

Note Cisco CMTS supports BSOD VLAN redundancy feature with support for two Ethernet Network Side Interface (NSI) configuration and a backup WAN interface. When the active NSI WAN interface is down, the L2VPN traffic flows through the backup WAN interface.

Supported L2VPN Encodings

This section describes the supported L2VPN encodings in the CM configuration file that are supported by the Cisco CMTS routers.

- The Cisco CMTS routers support the following CM capabilities:
  - L2VPN capability (5.17)
  - eSAFE host capability (5.18)
  - Downstream Unencrypted Traffic (DUT) filtering (5.19)
- The Cisco CMTS routers support the following top-level encodings:

- VPN identifier (43.5.1)
- CMIM (43.5.4)—When provided, applies to all upstream SFs associated with an L2VPN tunnel; Supports only one eSAFE host.
- NSI encapsulation (43.5.2) with format code 2 for IEEE 802.1q (43.5.2.2)
- DUT filtering encoding

- The Cisco CMTS routers support the following per-SF encodings:
  - VPN identifier (43.5.1)
  - Ingress user priority (43.5.8)
- The Cisco CMTS routers support the following downstream classifier encodings:
  - VPN identifier (43.5.1)
  - CMIM (43.5.4) and (22/23.13)
  - User priority range (43.5.9)

For more information about the CM configuration file and L2VPN encodings, see the "Business Services over DOCSIS (BSOD) Layer 2 Virtual Private Networks" specification.

For information about how to use the configuration file generator on the Cisco CMTS, see the “DOCSIS Internal Configuration File Generator for the Cisco CMTS” document.

Voice-Call Support on L2VPN CM

Voice calls are supported on L2VPN CMs. This feature enables the Cisco CMTS routers to support dynamic service flows on L2VPN-provisioned cable modems to permit voice calls from a non-L2VPN CPE.

To provide voice-call support on a L2VPN CM, you have to configure correct classifiers and create two static service flows (primary and secondary) using the cable modem configuration file. If the eMTA is L2VPN-capable with the embedded CPE configured as an eSAFE host, then only one service flow is required. When correct CMIM bits are configured, the Cisco CMTS does not send packets from the eSAFE host to the L2VPN.

Though the L2VPN can be configured on the primary or secondary service flow, it cannot coexist with eMTAs on the same service flow. The eMTAs should always use a different service flow from that of L2VPN. The classifiers to direct the traffic should also be based on the service flows the L2VPN and eMTAs are using. When the above configuration is in place, the dynamic service flows are created automatically whenever voice calls are initiated.

How to Configure L2VPN Support over Cable

This section contains the following procedures:

Configuring the Ethernet Network System Interface

To configure the L2VPN Support over Cable feature, you need to specify an Ethernet NSI to operate as the trunking interface for the L2VPN traffic. You must configure the NSI using a command on the Cisco CMTS router. It is not configurable through the CM configuration file.

Before you begin

The following interface types can be configured as an NSI for L2VPN Support over Cable:

- Cisco cBR Series Converged Broadband Router—GigabitEthernet and TenGigabitEthernet

Note The Cisco CMTS routers only support the configuration of a single L2VPN NSI per CMTS.

>

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**cable l2-vpn-service xconnect nsi dot1q interface***ethernet-intf*[**backup-interface ethernet-intf**] **Example:** `Router(config)# cable l2-vpn-service xconnect nsi dot1q interface Te4/1/0 backup-interface Te4/1/4`	Configures WAN interface for DOT1Q L2VPN . (Optional) Backup-interface - If backup-interface is configured it means that BSoD VLAN redundancy feature is enabled.

Preparing the DOCSIS Configuration File for L2VPN Support

To support L2VPN, the DOCSIS configuration file must be configured with the appropriate encodings. For information about the supported encodings by the Cisco CMTS routers, see the L2VPN Encodings in the CM Configuration File, on page 6.

Manual Switchover Command Line Interface

For BSoD VLAN Redundancy feature, users can manually switch active uplink ports from the active port to another port when both the uplink ports are up through the command line interface. To manually switchover, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **cable l2-vpn dot1q-nsi-redundancy force-switchover from** *active-nsi-interface*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted
Step 2	**cable l2-vpn dot1q-nsi-redundancy force-switchover from** *active-nsi-interface* **Example:** `Router# cable l2-vpn dot1q-nsi-redundancy force-switchover from Te4/0/1`	Switches the active uplink port from the current active port to the specified port.

To display the dot1q L2VPN uplink redundancy information, use the **show cable l2-vpn dot1q-nsi-redundancy** as shown in the following example:

```
Router# show cable l2-vpn dot1q-nsi-redundancy
Primary-NSI Backup-NSI Active-NSI Elapsed-after-SW
Te4/1/0       Te4/0/4    Te4/1/0    31m9s
Te4/1/2       Te4/0/5    Te4/1/2    59s
```

Verifying L2VPN Support over Cable

To verify L2VPN information on the Cisco CMTS router, use the **show cable l2-vpn xconnect dot1q-vc-map** command.

SUMMARY STEPS

1. To display VLAN information for all cable modems, use the **show cable l2-vpn xconnect dot1q-vc-map** command as shown in the following example:
2. To display VLAN information for a particular L2VPN ID or customer, use the **show cable l2-vpn xconnect dot1q-vc-map customer** form of the command as shown in the following example:
3. To display information for a particular L2VPN ID on a specific cable modem, use the **show cable l2-vpn xconnect dot1q-vc-map vpn** form of the command along with specification of the cable modem MAC address, as shown in the following example:
4. To display detailed information for a particular L2VPN ID on a specific cable modem, use the **show cable l2-vpn xconnect dot1q-vc-map vpn verbose** form of the command along with specification of the cable modem MAC address, as shown in the following example:

5. To display detailed information and the current redundancy information for a particular cable modem, use the the **show cable l2-vpn xconnect dot1q-vc-map verbose** form of the command along with specification of the cable modem MAC address, as shown in the following example:
6. To display the dot1q L2VPN uplink redundancy information, use the **show cable l2-vpn dot1q-nsi-redundancy** as shown in the following example:

DETAILED STEPS

Step 1 To display VLAN information for all cable modems, use the **show cable l2-vpn xconnect dot1q-vc-map** command as shown in the following example:

Example:

```
Router# show cable l2-vpn xconnect dot1q-vc-map
MAC Address    Ethernet Interface      VLAN ID   Cable Intf  SID  Customer Name/VPN ID
0014.f8c1.fd66 GigabitEthernet4/0/0        68    Cable6/0/0  3    0234560001
```

Step 2 To display VLAN information for a particular L2VPN ID or customer, use the **show cable l2-vpn xconnect dot1q-vc-map customer** form of the command as shown in the following example:

Example:

```
Router# show cable l2-vpn xconnect dot1q-vc-map customer 0234560001

MAC Address    Ethernet Interface      VLAN ID   Cable Intf  SID  Customer Name/VPNID
0014.f8c1.fd66 GigabitEthernet4/0/0        68    Cable6/0/0  3    0234560001
```

Step 3 To display information for a particular L2VPN ID on a specific cable modem, use the **show cable l2-vpn xconnect dot1q-vc-map vpn** form of the command along with specification of the cable modem MAC address, as shown in the following example:

Example:

```
Router# show cable l2-vpn xconnect dot1q-vc-map 0014.f8c1.fd66 vpn 0234560001

MAC Address    Ethernet Interface      VLAN ID   Cable Intf  SID  Customer Name/VPNID
0014.f8c1.fd66 GigabitEthernet4/0/0        68    Cable6/0/0  3    0234560001
```

Step 4 To display detailed information for a particular L2VPN ID on a specific cable modem, use the **show cable l2-vpn xconnect dot1q-vc-map vpn verbose** form of the command along with specification of the cable modem MAC address, as shown in the following example:

Example:

```
Router# show cable l2-vpn xconnect dot1q-vc-map 0014.f8c1.fd66 vpn 0234560001 verbose
MAC Address                  : 0014.f8c1.fd66
Prim Sid                     : 3
Cable Interface              : Cable6/0/0
VPN ID                       : 0234560001
L2VPN SAID                   : 12294
Upstream SFID                : 23
Downstream CFRID[SFID]       : 2[24]
CMIM                         : 0x60
Ethernet Interface           : GigabitEthernet4/0/0
DOT1Q VLAN ID                : 68
Total US pkts                : 1372
Total US bytes               : 500226
```

```
Total US pkt Discards              : 0
Total US byte Discards             : 0
Total DS pkts                      : 1248
Total DS bytes                     : 415584
Total DS pkt Discards              : 0
Total DS byte Discards             : 0
```

Step 5 To display detailed information and the current redundancy information for a particular cable modem, use the the **show cable l2-vpn xconnect dot1q-vc-map verbose** form of the command along with specification of the cable modem MAC address, as shown in the following example:

Example:

```
Router# show cable l2-vpn xconnect dot1q-vc-map 0014.f8c1.fd66 verbose
MAC Address                        : 5039.5589.4302
Prim Sid                           : 45
Cable Interface                    : Cable6/0/2
L2VPNs provisioned                 : 1
DUT Control/CMIM                   : Disable/0x8000FFFF

VPN ID                             : 000234560001
L2VPN SAID                         : 45
Upstream SFID Summary              : 77
Upstream SFID [77   ]              : SID 45
Downstream CFRID[SFID] Summary     : Primary SF
CMIM                               : 0x60
Primary Ethernet Interface         : GigabitEthernet4/0/0
Backup  Ethernet Interface         : GigabitEthernet4/0/1
Active  Ethernet Interface         : GigabitEthernet4/0/0
DOT1Q VLAN ID                      : 207
Total US pkts                      : 151269
Total US bytes                     : 211755224
Total DS pkts                      : 150502
Total DS bytes                     : 210463324
```

Step 6 To display the dot1q L2VPN uplink redundancy information, use the **show cable l2-vpn dot1q-nsi-redundancy** as shown in the following example:

Example:

```
Router# show cable l2-vpn dot1q-nsi-redundancy
Primary-NSI Backup-NSI Active-NSI Elapsed-after-SW
Te4/1/0       Te4/0/4    Te4/1/0    31m9s
Te4/1/2       Te4/0/5    Te4/1/2    59s
```

Enabling Voice-Call on a L2VPN CM

You can enable the Voice-Call Support on a L2VPN CM feature by registering a cable modem with a SID to VPN mapping cable modem configuration file (MPLS or 802.1q).

- If the L2VPN is on the primary service flow, you should use a cable modem configuration file with static secondary service flow and the classifiers should be configured on the secondary service flow for non-L2VPN packets.
- If the L2VPN is on the secondary service flow, then classifiers should be configured for L2VPN packets.

Note The cable modem configuration file based L2VPN configuration provides the flexibility to configure L2VPN on the primary or secondary service flow. However, we recommend that you configure L2VPN on the secondary service flow and the primary service flow is used for the default traffic.

Note In a CLI-based L2VPN configuration, the L2VPN is on the primary service flow; therefore the static secondary service flow should be used for the eMTAs.

Verifying Dynamic Service Flows

To verify dynamically created service flows on the Cisco CMTS router, use the **show interface cable service-flow** command.

Note To verify information about PacketCable operations, use **show packetcable** commands.

```
Router# show interface cable 5/1/0 service-flow
Sfid : 30191
Mac Address : 000a.739e.140a
Type : Secondary(Dynamic)
Direction : Upstream
Current State : Active
Current QoS Indexes [Prov, Adm, Act] : [0, 24, 24]
Active Time : 00:55
Sid : 7140
Admitted QoS Timeout : 200 seconds
Active QoS Timeout : 0 seconds
Packets : 1824
Bytes : 466944
Rate Limit Delayed Grants : 0
Rate Limit Dropped Grants : 0
Current Throughput : 68356 bits/sec, 32 packets/sec
Classifiers:
Classifier Id : 41
Service Flow Id : 30191
CM Mac Address : 000a.739e.140a
Direction : upstream
Activation State : active
Classifier Matching Priority : 128
PHSI : 1
Number of matches : -
IP Classification Parameters:
IP Source Address : 10.8.230.3
Source IP Address Mask : 255.255.255.255
Destination IP Address : 172.16.2.35
Destination IP Address Mask : 255.255.255.255
IP Protocol Type : 17
Source Port Low : 53456
Source Port High : 53456
Destination Port Low : 7052
Destination Port High : 7052
```

Configuration Examples for L2VPN over Cable

This section provides configuration examples for the L2VPN over Cable feature:

Example: Specifying the Ethernet NSI Interface

You can specify the Ethernet NSI within the CM configuration file, or using the **cable l2-vpn-service xconnect** global configuration command as shown in the following example:

```
cable l2-vpn-service xconnect nsi {dot1q|mpls}
```

Example: Enabling Voice Call Support on MPLS L2VPN

The following is a sample cable modem configuration file that enables voice call support on MPLS L2VPN. In this example the L2VPN is applied to the primary service flow.

```
03 (Net Access Control)         = 1
18 (Maximum Number of CPE)      = 16
43 (Vendor Specific Options)
   S08 (Vendor ID)      = ff ff ff
   S005 (Unknown sub-type)      = 01 04 32 30 32 30 02 07 04 05 01 0a 4c 02 01 2b 06 26 04
 00 00 01 90
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 2
   S03 (Service Flow Reference)         = 2
   S09 (IP Packet Encodings)
       T03 (IP Source Address)          = 050 001 005 000
       T04 (IP Source Mask)             = 255 255 255 000
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 3
   S03 (Service Flow Reference)         = 2
   S10 (Ethernet LLC Packet Classification Encodings)
       T02 (Source MAC Address)         = 00 e0 f7 5a c9 21
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 21
   S03 (Service Flow Reference)         = 21
   S05 (Rule Priority)                  = 5
   S09 (IP Packet Encodings)
       T05 (IP Destination Address)     = 050 001 005 000
       T06 (IP Destination Mask)        = 255 255 255 000
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 22
   S03 (Service Flow Reference)         = 21
   S05 (Rule Priority)                  = 5
   S10 (Ethernet LLC Packet Classification Encodings)
       T01 (Destination MAC Address)    = 00 e0 f7 5a c9 21 ff ff ff ff ff ff
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 1
   S06 (QoS Parameter Set Type)         = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID)  = ff ff ff
       T005 (Unknown sub-type)  = 01 04 32 30 32 30
24 (Upstream Service Flow Encodings)
```

```
   S01 (Service Flow Reference)         = 2
   S06 (QoS Parameter Set Type)         = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 20
   S06 (QoS Parameter Set Type)         = 7
   S07 (Traffic Priority)               = 0
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 21
   S06 (QoS Parameter Set Type)         = 7
   S07 (Traffic Priority)               = 1
29 (Privacy Enable)             = 1
```

Example: Enabling Voice Call Support on 802.1q L2VPN

The following is a sample cable modem configuration file that enables voice call support on 802.1q L2VPN. In this example the L2VPN is applied to the secondary service flow.

```
03 (Net Access Control)         = 1
43 (Vendor Specific Options)
   S08 (Vendor ID)      = ff ff ff
   S005 (Unknown sub-type)      = 01 05 02 34 56 00 01 02 04 02 02 00 44
18 (Maximum Number of CPE)      = 16
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 2
   S03 (Service Flow Reference)         = 2
   S10 (Ethernet LLC Packet Classification Encodings)
       T02 (Source MAC Address)         = 00 e0 14 e3 23 1c
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 4
   S03 (Service Flow Reference)         = 4
   S43 (Vendor Specific Options)
       T08 (Vendor ID)  = ff ff ff
       T005 (Unknown sub-type)  = 01 05 02 34 56 00 01
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T01 (IEEE 802.1P UserPriority)   = 00 07
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 1
   S06 (QoS Parameter Set Type)         = 7
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 2
   S06 (QoS Parameter Set Type)         = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID)  = ff ff ff
       T005 (Unknown sub-type)  = 01 05 02 34 56 00 01 08 01 01
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 3
   S06 (QoS Parameter Set Type)         = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 4
   S06 (QoS Parameter Set Type)         = 7
```

Example: Enabling Voice Call Support on CLI-based L2VPN

The following is a sample cable modem configuration file that enables voice call support on L2VPN configured using CLI. L2VPN configured using the CLI is always applied to the primary service flow.

```
03 (Net Access Control)         = 1
18 (Maximum Number of CPE)      = 16
22 (Upstream Packet Classification Encoding Block)
```

```
   S01 (Classifier Reference)           = 2
   S03 (Service Flow Reference)         = 2
   S09 (IP Packet Encodings)
       T03 (IP Source Address)          = 050 001 005 000
       T04 (IP Source Mask)             = 255 255 255 000
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 3
   S03 (Service Flow Reference)         = 2
   S10 (Ethernet LLC Packet Classification Encodings)
       T02 (Source MAC Address)         = 00 e0 f7 5a c9 21
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 21
   S03 (Service Flow Reference)         = 21
   S05 (Rule Priority)                  = 5
   S09 (IP Packet Encodings)
       T05 (IP Destination Address)     = 050 001 005 000
       T06 (IP Destination Mask)        = 255 255 255 000
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)           = 22
   S03 (Service Flow Reference)         = 21
   S05 (Rule Priority)                  = 5
   S10 (Ethernet LLC Packet Classification Encodings)
       T01 (Destination MAC Address)    = 00 e0 f7 5a c9 21 ff ff ff ff ff ff
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 1
   S06 (QoS Parameter Set Type)         = 7
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 2
   S06 (QoS Parameter Set Type)         = 77
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 20
   S06 (QoS Parameter Set Type)         = 7
   S07 (Traffic Priority)               = 0
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 21
   S06 (QoS Parameter Set Type)         = 7
   S07 (Traffic Priority)               = 1
29 (Privacy Enable)             = 1
```

Additional References

The following sections provide references related to the L2VPN Support over Cable feature.

Standards

Standard	Title
CM-SP-BPI+-I12-050812	*Baseline Privacy Plus Interface Specification* http://www.cablelabs.com/wp-content/uploads/specdocs/CM-SP-BPI+-C01-081104.p
CM-SP-L2VPN-I03-061222	*Business Services over DOCSIS (BSOD) Layer 2 Virtual Private Networks* http://www.cablelabs.com/wp-content/uploads/specdocs/CM-SP-L2VPN-I12-131120.
CM-SP-RFIv2.0-I11-060602	*Radio Frequency Interface Specification* http://www.cablelabs.com/wp-content/uploads/specdocs/CM-SP-RFIv2.0-C02-090422.

Standard	Title
IEEE 802.1ad	*IEEE 802.1ad-2005 IEEE Standards for Local and metropolitan area networks*— Virtual Bridged Local Area Networks http://www.ieee.org
IEEE 802.1q	*IEEE Std 802.1Q Virtual Bridged Local Area Networks* http://www.ieee.org

MIBs

MIB	MIBs Link
DOCS-L2VPN-MIB	To locate and download MIBs for selected platforms, Cisco IOS-XE releases, and feature sets, use Cisco MIB Locator found at the following URL: http://tools.cisco.com/ITDIT/MIBS/servlet/index

RFCs

RFC	Title
RFC 2685	Virtual Private Networks Identifier http://www.ietf.org/rfc/rfc2685.txt
RFC 4364	*BGP/MPLS IP Virtual Private Networks (VPNs)* http://www.ietf.org/rfc/rfc4364.txt

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

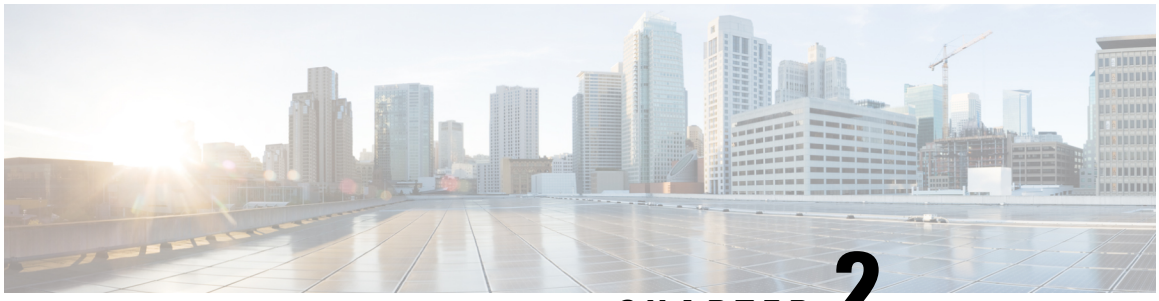
Feature Information for L2VPN Support over Cable

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 2: Feature Information for L2VPN Support Over Cable

Feature Name	Releases	Feature Information
L2VPN support over cable	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on the Cisco cBR Series Converged Broadband Router.



CHAPTER 2

L2VPN Over Port-Channel

The Layer 2 VPN (L2VPN) over port-channel feature supports IEEE 802.1Q (dot1q) L2VPN WAN interface port-channel. Using this feature, you can configure the dot1q L2VPN traffic to pass through port-channel uplink

Contents

Information About L2VPN Over Port-Channel

The Cisco cBR-8 supports L2VPN, where the Ethernet frames from the cable modem are cross connected to a specific VLAN interface. The VLAN ID to be inserted is specified. With the L2VPN over port-channel feature, you can now support port-channel uplink interface as well as the 10 Gb uplink interface.

TLS L2VPN

For the Transparent LAN Service (TLS) L2VPN, the dot1q maps contain the cable modem MAC address, the VLAN ID, and the outbound interface. Traffic received from a specific cable modem is tagged with a VLAN ID and is sent out from the uplink interface.

DOCSIS L2VPN

For the Data-over-Cable Service Interface Specifications (DOCSIS) L2VPN, cable modem (CM) configuration file holds the L2VPN encodings for both, the CM and the service flow. At the CMTS level you have to specify the default port-channel Network Side Interface (NSI). L2VPN encodings are passed by the CM to the CMTS during registration. The CMTS installs DOCSIS service flow VLAN mapping based on the information passed to it during the registration. For upstream traffic, the CMTS sends the dot1q VLAN tagged traffic out from the uplink interface. On downstream, the CMTS receives the dot1q tagged traffic from the aggregator. The CMTS replaces the VLAN header with a DOCSIS header to the corresponding service flow.

Benefits of L2VPN Over Port-Channel

By using the dot1q L2VPN, you can utilize the port-channel interface feature instead of a single 10 Gb port.

Restrictions for L2VPN Over Port-Channel

The CMTS dot1q L2VPN is designed to support traffic from customer premises equipment to the network or verse vice. For CMTS L2VPN NSI port, port-channel interface does not support VLAN redundancy.

How to Configure the L2VPN Over Port-Channel

This section describes how to configure L2VPN over port-channel on the Cisco cBR-8.

Configuring the Port-Channel Uplink Port for TLS L2VPN

For TLS L2VPN, you must configure the overall enable CLI and the dot1q map. In dot1q map, you have to designate the port-channel uplink port.

To configure the port-channel uplink port for TLS L2VPN, complete the following procedure:

```
cable l2-vpn-service xconnect nsi dot1q
cable dot1q-vc-map mac address port-channel number vlan id custom name
```

Configuring the Port-Channel Uplink Port for DOCSIS L2VPN

For DOCSIS L2VPN, you only have to configure the overall enable CLI with port-channel uplink port. The other L2VPN related parameters are setup by the CM configuration file type-length-value parsing.

To configure the port-channel uplink port for DOCSIS L2VPN, complete the following procedure:

```
configure terminal
cable l2-vpn-service xconnect nsi dot1q interface port-channel number
```

Verifying Port-Channel Configuration

Verify the Port-Channel Mapping

To verify the port-channel mapping, use the **show cable l2-vpn xconnect dot1q-vc-map** command as shown in the example below:

```
show cable l2-vpn xconnect dot1q-vc-map

MAC Address    Ethernet Interface      VLAN ID   Cable Intf  SID  Customer Name/VPNID
c8fb.26a5.551c Port-channel64            1200    Cable6/0/0  17   Topgun
```

View the Port-Channel Interface

To view the port-channel interface, use the **show cable l2-vpn xconnect dot1q-vc-map verbose** command as shown in the example below:

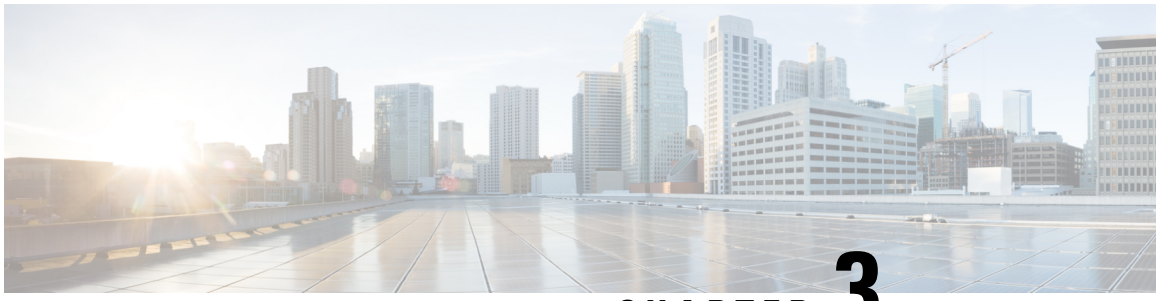
```
show cable l2-vpn xconnect dot1q-vc-map c8fb.26a5.551c verbose

MAC Address                          : c8fb.26a5.551c
Customer Name                        : ats
Prim Sid                             : 17
Cable Interface                      : Cable6/0/0
Ethernet Interface                   : Port-channel64
DOT1Q VLAN ID                        : 1200
Total US pkts                        : 189
Total US bytes                       : 18200
Total DS pkts                        : 615
Total DS bytes                       : 39360
```

Feature Information for L2VPN Over Port-Channel

Table 3: Feature Information for L2VPN Over Port-Channel

Feature Name	Releases	Feature Information
L2VPN over port-channel	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on the Cisco cBR Series Converged Broadband Router.



CHAPTER 3

MPLS Pseudowire for Cable L2VPN

The Multiprotocol Label Switching (MPLS) Pseudowire for Cable Layer 2 Virtual Private Network (L2VPN) feature enables service providers to use a single, converged, Internet Protocol (IP)/MPLS network infrastructure to offer Ethernet data link layer (Layer 2) connectivity to two or more VPN customer sites.

Finding Feature Information

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 4: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Prerequisites for MPLS Pseudowire for Cable L2VPN

- Enable Baseline Privacy Interface Plus (BPI+) to provide a simple data encryption scheme to protect data sent to and from cable modems in a data over cable network.
- Enable Cisco Express Forwarding (CEF) to optimize network performance.
- Ensure that the primary and backup pseudowires on the remote provider edge (PE) routers have the same pseudowire type as the Cisco cable modem termination system (CMTS).
- Create the remote pseudowire using a pw-class with VLAN as the interworking for remote PEs, if the CMTS is using VLAN as pseudowire type.

Restrictions for MPLS Pseudowire for Cable L2VPN

The following are the general restrictions for the MPLS Pseudowire for Cable L2VPN feature:

- Supports only Ethernet over MPLS (EoMPLS) pseudowires per RFC 4448.
- Supports only point-to-point forwarding. Ethernet switching is not supported.
- Requires DOCSIS 2.0, 3.0 and 3.1-certified cable modems (CMs). This feature is not supported on DOCSIS 1.0-certified cable modems.
- Supports a maximum of four VPNs per cable modem.
- Supports a maximum of eight upstream service flows and eight downstream classifiers.
- Supports a maximum of 16000 EoMPLS pseudowires per Cisco CMTS router.
- Requires the backup pseudowire to be up on the remote PE for the Cisco CMTS to switchover.
- Requires the backup pseudowire to become active on the Cisco CMTS only after the primary pseudowire fails.

Note The CLI-based (static provisioning) L2VPN supports traffic forwarding to VPN only on primary upstream and downstream service flows. Hence only primary upstream and downstream service flows must be configured in the cable modem configuration file.

Information About MPLS Pseudowire for Cable L2VPN

The MPLS Pseudowire for Cable L2VPN feature enables Ethernet-based Layer 2 VPN service over an MPLS network by encapsulating and transmitting the Layer 2 protocol data units (PDUs) over pseudowires (PWs). This feature enables service providers to offer site-to-site connectivity to their business and enterprise customers.

Layer 2 services emulated over an MPLS network are commonly referred to as MPLS-based L2VPNs or MPLS L2VPNs. Subsequently, Ethernet service emulated over an MPLS network is referred to as Ethernet over MPLS (EoMPLS) service.

The MPLS Pseudowire for Cable L2VPN feature is fully compliant with CableLabs Business Services over DOCSIS (BSOD) L2VPN specification, and is an extension to the existing DOCSIS L2VPN features supported on Cisco CMTS routers.

The MPLS Pseudowire for Cable L2VPN feature provides the following capabilities:

- Transport Ethernet frames over an MPLS network.
- Handle a DOCSIS service flow as an attachment circuit that is mapped to an EoMPLS pseudowire.
- Enable the Cisco CMTS router to be the MPLS provider edge (PE) router.
- Enable forwarding of Ethernet frames over DOCSIS (between a CM and a Cisco CMTS router) to MPLS (towards Metropolitan Area Network or Wide Area Network).
- Provide a common framework to encapsulate and transport supported Layer 2 traffic types over an MPLS network.

The MPLS Pseudowire for Cable L2VPN feature differs from the existing DOCSIS L2VPN features such as 802.1q-based L2VPN (L2VPN Support over Cable). The MPLS Pseudowire for Cable L2VPN feature uses IP/MPLS network to transport layer 2 protocol data units (PDUs), whereas 802.1q-based L2VPN feature uses layer 2 Ethernet network to transport PDUs.

How MPLS Transports Layer 2 Packets

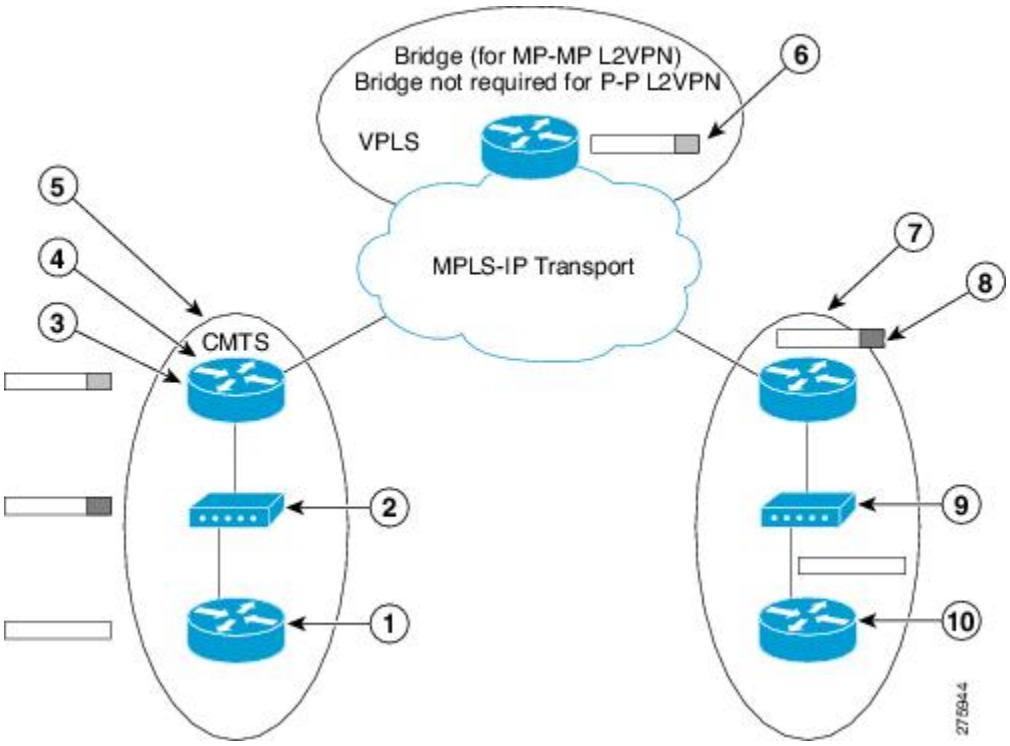
The MPLS subsystem removes DOCSIS encapsulation for Layer 2 Ethernet frames and adds MPLS labels at the ingress provider edge (PE) Cisco CMTS router. Then, the MPLS subsystem sends resulting MPLS packets to the corresponding PE router at the other end of the pseudowire. The PE routers must be configured for successful transmission of IP/MPLS packets between the two PE routers.

The cable modem classifies Ethernet frames from the customer premise equipment (CPE) in the upstream direction using upstream classifiers. Then, a DOCSIS header is added to these frames, and they are sent on a given upstream service flow with a different service identifier. On the Cisco CMTS router, the upstream packet is classified as an L2VPN packet based on the cable interface and service identifier. The Cisco CMTS router removes the DOCSIS header and adds an MPLS header. An MPLS header contains two MPLS labels: the outer label corresponding to the remote PE router and the inner label corresponding to the pseudowire label. The Cisco CMTS router forwards the MPLS packet towards the remote PE router, which is the other end of the pseudowire, over the MPLS network.

In the downstream direction, the Cisco CMTS router receives MPLS packets having only one MPLS header that contains the label that the Cisco CMTS router previously allocated for the corresponding EoMPLS pseudowire. The Cisco CMTS router uses the MPLS label to identify one of the L2VPN cable modems. Then, the Cisco CMTS router classifies the MPLS packet using the L2VPN downstream classifiers based on MPLS experimental (MPLS-EXP) bits in the MPLS header of the received MPLS packet, and removes the MPLS header. Then, the Cisco CMTS router sends the packet on the classified downstream service flow by adding the DOCSIS header. The cable modem then removes the DOCSIS header and delivers the Ethernet frame to the CPE.

A unique combination of a cable modem MAC address, VPN ID (if present in the CM configuration file), peer IP address, and a virtual circuit ID (VCID) identifies the MPLS pseudowire on the Cisco CMTS router.

Figure 2: Transporting Layer 2 Packets



The table illustrates how MPLS transports Layer 2 packets in a DOCSIS-based cable communications system.

1	A router sends an untagged Ethernet frame.	6	MPLS packets are label switched.
2	A CM adds a DOCSIS header to the frame.	7	The Cisco CMTS router receives an MPLS packet and looks up the MPLS forwarding table using the label value in the MPLS header.
3	The Cisco CMTS router removes the DOCSIS header from the frame.	8	The Cisco CMTS router replaces the MPLS header with DOCSIS header (containing the right SID value).
4	The Cisco CMTS router looks up the Service ID (SID) database using the SID value from the DOCSIS header and finds the MPLS header.	9	The DOCSIS header is removed.
5	The Cisco CMTS router adds the MPLS header to the frame.	10	The Ethernet frame is delivered untagged.

Supported Ethernet Encapsulation on UNI

The Ethernet User-Network Interface (UNI) is the connection between a cable modem and a customer premise equipment such as a router or a switch. The service provider may or may not use any encapsulation on the UNI.

The MPLS Pseudowire for Cable L2VPN feature supports the following transport types on an Ethernet UNI:

- Port-based UNI (independent of any VLAN)—The port-based UNI provides Metro Ethernet Forum (MEF)-defined Ethernet Private Line (EPL) service. In this transport type, an MPLS pseudowire is mapped to the Ethernet port.
- VLAN-based UNI—Ethernet VLAN using 802.1q encapsulation (including stacked VLANs). The VLAN-based UNI provides MEF-defined Ethernet Virtual Private Line (EVPL) service. In this transport type, the MPLS pseudowire is mapped to the 802.1q VLAN.

Note The Ethernet UNI must be attached to the Ethernet port of a cable modem.

Before configuring this feature, you should understand the following concepts:

MPLS Pseudowire

Pseudowire is a point-to-point Layer 2 connection between two PE routers. The MPLS Pseudowire for Cable L2VPN feature supports the following pseudowire types:

- Type-4 pseudowire—This is used to transport only VLAN tagged Layer 2 Ethernet frames.
- Type-5 pseudowire—This is used to transport VLAN tagged and untagged Layer 2 Ethernet frames. This is the default pseudowire type.

Bundle254 Interface

The bundle254 (Bu254) interface is an internal bundle interface on a Cisco CMTS router that is used as a circuit identifier for all MPLS pseudowires. This internal bundle interface is created automatically on a Cisco CMTS router when you enable the MPLS pseudowire functionality using the **cable l2-vpn-service xconnect** command. Only one Bu254 interface is created to handle all the MPLS pseudowires available on the Cisco CMTS router.

The output of the **show xconnect or show cable l2-vpn xconnect command displays** the circuit identifier created by the Cisco CMTS router for all the MPLS pseudowires**.**

Ingress Process

When an upstream packet received from a cable interface of the Cisco CMTS router is identified as an L2VPN packet based on the cable modem interface and Service ID (SID), the packet goes through the ingress process. The ingress process ensures that the DOCSIS header is removed, and an MPLS label header is added to the packet according to the MPLS pseudowire configuration and the packet is sent out from the Ethernet interface of the Cisco CMTS router. The ingress process is also known as the label imposition process.

Egress Process

When a downstream packet received from an Ethernet interface of the Cisco CMTS router is identified as an L2VPN packet by the innermost MPLS label, the packet goes through the egress process. The egress process ensures that the MPLS label header is deleted from the packet and the DOCSIS header is added to the packet. Then the packet is sent out from the cable interface of the Cisco CMTS router. The egress process is also known as the label disposition process.

MPLS Pseudowire Control Plane Process

When an L2VPN-compliant CM registers with a Cisco CMTS router and conveys the L2VPN related parameters to the router, the router follows the standard Label Distribution Protocol (LDP) procedures to set up an Ethernet over MPLS pseudowire with the remote PE router. When the L2VPN-compliant CM goes offline, the Cisco CMTS router brings down the pseudowire as well. If the Cisco CMTS router has no L2VPN-compliant CM registered, then the router tears down the targeted LDP session with the remote PE router.

L2VPN Pseudowire Redundancy

The L2VPN Pseudowire Redundancy feature enables a PE router to detect a pseudowire failure and reroute the Layer 2 service to a backup pseudowire that can continue to provide the service. The pseudowire redundancy can be implemented with either Cisco CMTS or a generic router as the PE router. When the primary pseudowire recovers from the failure, the L2VPN Pseudowire Redundancy feature provides the option to bring back the Layer 2 service to the primary pseudowire.

Each primary pseudowire can have up to three backup pseudowires, with unique priorities. For example, priority one cannot be given to two different pseudowires in the backup list. When the primary pseudowire goes down, the Cisco CMTS sends the traffic to the backup pseudowire with the highest priority. For a successful service transfer, the remote state of the backup pseudowire should already be ‘up’. Only the local state of the active pseudowire will be 'up' when the modem is BPI online. Similarly, if the backup pseudowire is in use, the local state of only that backup pseudowire will be 'up'.

If the active backup pseudowire goes down, the Cisco CMTS will use the next highest backup pseudowire whose remote state is ‘up’. However, the Cisco CMTS will not switchover from the lower priority pseudowire to the higher priority pseudowire when the backup pseudowire with the highest priority comes ‘up’. This is to prevent unnecessary switchovers between the backup pseudowires.

When the primary pseudowire recovers from the failure, the L2VPN Pseudowire Redundancy feature brings back the service to the primary pseudowire, after waiting for the time period set using the backup delay command. The local state of the active backup pseudowire will be marked as ‘down’ after the primary pseudowire comes up.

MPLS Pseudowire Provisioning Methods

The MPLS Pseudowire for Cable L2VPN feature supports the following provisioning methods for pseudowires:

Note Before performing the static or dynamic provisioning of MPLS pseudowires, you must enable MPLS on a Cisco CMTS router. For details on the tasks required to enable MPLS, see the How to Enable MPLS on a Cisco CMTS Router.

Static Provisioning Method for MPLS Pseudowires

The static provisioning method requires the MPLS pseudowire to be statically provisioned on the CMTS using the command line interface (CLI). This type of provisioning does not require the CM configuration file to use BSOD L2VPN-compliant TLVs. For details on how to statically provision MPLS pseudowires, see the *Static Provisioning of MPLS Pseudowires*.

Dynamic Provisioning Method for MPLS Pseudowires

The dynamic provisioning method is a CM configuration file-based provisioning method and is the recommended provisioning method for creating MPLS pseudowires. For details on how to dynamically provision MPLS pseudowires, see the Dynamic Provisioning of MPLS Pseudowires, on page 38.

The following are the benefits of dynamic provisioning of pseudowires:

- Multiple VPNs can be specified in a CM configuration file and a pseudowire can be provisioned for each VPN.
- Multiple upstream service flows and downstream classifiers can be associated with each VPN.
- Each upstream service flow can be tagged to an MPLS experimental (EXP) level for the egress WAN traffic.
- Downstream ingress WAN traffic can be classified based on the downstream MPLS-EXP range specified in each downstream classifier.
- The Cisco CMTS router will have finer control of MPLS quality of service (QoS) over cable and WAN interfaces.

For dynamic provisioning of MPLS pseudowires, you use an L2VPN-compliant CM configuration file that is stored on the Trivial File Transfer Protocol (TFTP) server. You use a common CM configuration file editor such as CableLabs Config File Editor, or a sophisticated provisioning backend system such as Broadband Access Center for Cable (BACC) to create CM configuration files.

This provisioning method requires the usage of CableLabs defined L2VPN encodings such as type, length, value (TLV) objects in the CM configuration file. These L2VPN encodings control L2VPN forwarding of upstream and downstream Ethernet frames.

You can specify the L2VPN encodings in the following ways:

- Per CM
- Per downstream classifier
- Per service flow
- Per upstream classifier

Note The CM L2VPN encoding is mandatory.

The CM L2VPN encoding contains many TLVs, out of which the two most important TLVs are VPN Identifier and NSI Encapsulation. To configure an MPLS pseudowire, you must set the NSI Encapsulation to MPLS. The other TLVs are used to specify the pseudowire identifiers in the form of source attachment individual identifier (SAII), target attachment individual identifier (TAII), and attachment group identifier (AGI).

The L2VPN encoding parameter is encoded as a general extension information (GEI) parameter in the CM configuration file. This indicates that the parameter is encoded as a subtype of the vendor-specific information type parameter using the vendor ID (0xFFFFFF).

The table lists the important CableLabs defined TLVs that are used at the top level of the CM configuration file for the MPLS Pseudowire for Cable L2VPN feature. See the BSOD specification, *Business Services over DOCSIS (BSOD) Layer 2 Virtual Private Networks,* from CableLabs for *a complete list of CableLabs defined TLVs.*

Table 5: CableLabs Defined L2VPN TLVs

TLV Name	Type	Length	Value and Description
Downstream Unencrypted Traffic (DUT) Control	45.1	1	Bit 0 DUT Filtering DUT Filtering = 0: Disable (default) DUT Filtering = 1: Enable DUT Filtering
Downstream Unencrypted Traffic (DUT) CMIM	45.2	N	DUT CMIM (optional) CM Interface Mask (CMIM) limiting outgoing interfaces of DUT traffic. If the DUT CMIM is omitted, its default value includes the eCM and all implemented eSAFE interfaces, but not any CPE interfaces.
VPN Identifier	43.5.1	1 to N	An opaque octet string that identifies an L2VPN. N is vendor- specific, and the valid range is from 6 to 255.
NSI Encapsulation Subtype	43.5.2	n	A single NSI encapsulation format code/length/value tuple. This TLV uses any of the following values: NSI encapsulation = 0 : Other NSI encapsulation = 1 : IEEE 802.1Q (specify VLAN ID) NSI encapsulation = 2 : IEEE 802.1AD (specify Q-in-Q) NSI encapsulation = 3 : MPLS peer (specify IPv4 or IPv6 address) The value must be set to 3 to ensure MPLS pseudowire usage. The address must identify the remote PE (by its IP address assigned to the loopback interface).
Attachment Group ID	43.5.5	0 to 16	Opaque byte string that identifies the CM or SF as an attachment circuit for IETF Layer 2 VPN signaling protocols.
Source Attachment Individual ID	43.5.6	0 to 16	Opaque byte string signaled as SAII circuit for IETF Layer 2 VPN signaling protocols.
Target Attachment Individual ID	43.5.7	0 to 16	Opaque byte string that identifies the CM or SF as an attachment circuit for IETF Layer 2 VPN signaling protocols.
Ingress User Priority	43.5.8	1	Ingress IEEE 802.1 user priority value in the range of 0 to 7 encoded in the least significant three bits. Higher values indicate higher priority.
User Priority Range	43.5.9	2	The lower user priority value of the user priority range is encoded in the least significant three bits of the first byte, and the higher value of the range is encoded in the least significant three bits of the second byte.

Cisco-Specific L2VPN TLVs

Even though CableLabs defined L2VPN TLVs are sufficient for dynamic provisioning of MPLS pseudowires, CMTS operators can use Cisco-specific TLVs at the top level of the CM configuration file to enable additional functions.

This table lists the new Cisco-specific TLVs that are defined for the MPLS Pseudowire for Cable L2VPN feature.

Table 6: Cisco-Specific L2VPN TLVs

TLV Name	Type	Length	Value	Description
MPLS-PW-TYPE	43.5.43.36	1	• 4 = Type-4 Ethernet VLAN • 5 = Type-5 Ethernet port	The Cisco CMTS router interprets this subtype as MPLS pseudowire type (Type-4 or Type-5). If this TLV value is not specified, then the router accepts the default value (5) for Type-5.
MPLS-VCID	43.5.43.38	4	4 bytes unsigned number = MPLS VCID	This subtype is interpreted as MPLS VCID. This TLV is ignored, and the value of TAII is used as VCID for the pseudowire, if the following conditions are met: • The CableLabs BSOD specification-compliant TLVs, SAII and TAII, are present in the CM configuration file. • Both are of 4 bytes length. • Value of SAII is equal to TAII.
MPLS-PEERNAME	43.5.43.39	N	ASCII encoded data	The Cisco CMTS router interprets this optional subtype as MPLS peer name in ASCII encoded data.

This table lists the new Cisco-specific type, length, values (TLVs) that are defined for the L2VPN Pseudowire Redundancy feature.

Table 7: Cisco-Specific L2VPN TLVs for Pseudowire Redundancy

TLV Name	Type	Length	Value	Description
BACKUP-PW	45.5.43.40	N	Backup pseudowire related parameters	The Cisco CMTS router interprets this subtype as related parameters for the MPLS backup pseudowire. This TLV indicates the start of a new backup pseudowire.
BACKUP-PEERIP	43.5.43.40.1	4	IP address of the backup peer (IPv4)	The Cisco CMTS router interprets this optional subtype as the peer IP address of the MPLS backup pseudowire. This TLV is an IPv4 address.
BACKUP-PEERNAME	43.5.43.40.2	N	ASCII encoded data	The Cisco CMTS router interprets this optional subtype as the MPLS backup peer name in ASCII encoded data. This TLV is resolved to IPv4 address through DNS.

TLV Name	Type	Length	Value	Description
BACKUP-MPLS-VCID	43.5.43.40.3	4	4 bytes unsigned number = MPLS VCID for backup pseudowire	The Cisco CMTS router interprets this subtype as the VCID of the backup pseudowire. This TLV is ignored, and the value of TAII is used as the VCID for the pseudowire, if the following conditions are met: • The CableLabs BSOD specification-compliant TLVs, SAII, and TAII, are present in the CM configuration file. • SAII, and TAII are of 4 bytes length. • Value of SAII is equal to TAII.
BACKUP-MPLS-PRIORITY	43.5.43.40.4	1	1 byte unsigned number = priority for the backup pseudowire	The Cisco CMTS router interprets this subtype as the MPLS priority. Each primary pseudowire can have up to three backup pseudowires, with unique priorities. The priority indicates the order in which the CMTS should switch to the backup peer when the primary peer is down.
BACKUP-ENABLE-DELAY	43.5.43.41	1	1 byte unsigned number = number of seconds	The Cisco CMTS router interprets this subtype as the number of seconds the backup pseudowire should wait to take over after the primary pseudowire goes down. If the TLV value is not specified, then the router uses the default value of 0 seconds.
BACKUP-DISABLE-DELAY	43.5.43.42	1	1 byte unsigned number = number of seconds	The Cisco CMTS router interprets this subtype as the number of seconds the primary pseudowire should wait to take over after the remote state of the primary pseudowire comes up. If the TLV value is not specified, then the router uses the default value of 0 seconds.
BACKUP-DISABLE-NEVER	43.5.43.43	1	1 byte unsigned number = never disable backup pseudowire	The Cisco CMTS router interprets this subtype as a flag indicating that the backup pseudowire should not be disabled even after the primary pseudowire comes up. If this TLV is not present, the router takes the default action of reverting back to the primary pseudowire.

How to Enable MPLS on a Cisco CMTS Router

Perform the following tasks in the same order to enable MPLS on a Cisco CMTS router:

Note Before performing the static or dynamic provisioning of MPLS pseudowires, you must enable MPLS on a Cisco CMTS router.

Configuring an LDP Router ID

The **mpls ldp router-id** command allows you to assign an interface IP address as the LDP router ID.

The normal process to determine the LDP router ID is as follows:

1. The router considers all the IP addresses of all operational interfaces.
2. If these addresses include loopback interface addresses, the router selects the largest loopback address. Configuring a loopback address helps ensure a stable LDP ID for the router, because the state of loopback addresses does not change. However, configuring a loopback interface and IP address on each router is not required.

The loopback IP address is not considered as the router ID of the local LDP ID under the following circumstances:

1. If the loopback interface has been explicitly shut down.
2. If the **mpls ldp router-id** command specifies that a different interface should be used as the LDP router ID.
3. If you use a loopback interface, make sure that the IP address for the loopback interface is configured with a /32 network mask. In addition, ensure that the routing protocol in use is configured to advertise the corresponding /32 network. Otherwise, the router selects the largest interface address.

The router might select a router ID that is not usable in certain situations. For example, the router might select an IP address that the routing protocol cannot advertise to a neighboring router. The router implements the router ID the next time it is necessary to select an LDP router ID. The effect of the **mpls ldp router-id** command is delayed until it is necessary to select an LDP router ID, which is the next time the interface is shut down or the address is deconfigured.

If you use the **force** keyword with the **mpls ldp router-id** command, the router ID takes effect more quickly. However, implementing the router ID depends on the current state of the specified interface:

- If the interface is up (operational) and its IP address is not currently the LDP router ID, the LDP router ID is forcibly changed to the IP address of the interface. This forced change in the LDP router ID tears down any existing LDP sessions, releases label bindings learned via the LDP sessions, and interrupts MPLS forwarding activity associated with the bindings.
- If the interface is down, the LDP router ID is forcibly changed to the IP address of the interface when the interface transitions to up. This forced change in the LDP router ID tears down any existing LDP sessions, releases label bindings learned via the LDP sessions, and interrupts MPLS forwarding activity associated with the bindings.

Before you begin

Ensure that the specified interface is operational before assigning it as the LDP router ID.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mpls ip** **Example:** `Router(config)# mpls ip`	Enables the dynamic MPLS forwarding function on the specified Gigabit Ethernet interface.
Step 4	**mpls ldp router-id loopback** interface-number [**force**] **Example:** `Router(config)# mpls ldp router-id loopback 2030 force`	Specifies the IP address of the loopback interface as the LDP router ID.
Step 5	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and enters privileged EXEC mode.

Configuring MPLS on a Gigabit Ethernet Interface

MPLS forwarding and Label Distribution Protocol must be enabled on 1-port or 10-port GE interfaces of the Cisco CMTS router to ensure that the router establishes MPLS label-switched path (LSP) to the remote PE routers. This section explains how to enable MPLS forwarding and LDP on a Gigabit Ethernet interface.

Note Configuration steps are similar for 1-port and 10-port GE interfaces.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface gigabitethernet** slot/subslot/port **Example:** `Router(config)# interface gigabitethernet 3/0/0`	Enters interface cable configuration mode and specifies the Gigabit Ethernet interface.
Step 4	**mpls ip** **Example:** `Router(config-if)# mpls ip`	Enables the dynamic MPLS forwarding function on the specified Gigabit Ethernet interface.
Step 5	**end** **Example:** `Router(config-if)# end`	Exits interface cable configuration mode and enters privileged EXEC mode.

Configuring an MPLS Label Distribution Protocol

The MPLS label distribution protocol (LDP) allows the construction of highly scalable and flexible IP VPNs that support multiple levels of services. This section explains how to configure an MPLS label distribution protocol on a Gigabit Ethernet interface.

MPLS LDP graceful-restart may also be configured for faster L2VPN traffic recovery after a LDP session disruption. For more information see the MPLS LDP Graceful Restart guide.

Note Ensure that the loopback interface with the IP address is present on each PE router using the **show ip interface brief** command before configuring an MPLS label distribution protocol. This loopback interface identifies the Cisco CMTS router as the peer IP address of the pseudowire.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface gigabitethernet** slot/subslot/port **Example:** `Router(config)# interface gigabitethernet 3/0/0`	Enters interface cable configuration mode and specifies the Gigabit Ethernet interface.
Step 4	**mpls label protocol ldp** **Example:** `Router(config-if)# mpls label protocol ldp`	Enables MPLS LDP parameters on the specified Gigabit Ethernet interface.
Step 5	**end** **Example:** `Router(config-if)# end`	Exits interface cable configuration mode and enters privileged EXEC mode.

Enabling the Cisco CMTS Support for MPLS Pseudowire for Cable L2VPN

You must enable the MPLS tunnel traffic on the network side of the interface to support configuration of MPLS pseudowires on a Cisco CMTS router.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**cable l2-vpn-service xconnect nsi mpls** **Example:** `Router(config)# cable l2-vpn-service xconnect nsi mpls`	Enables the MPLS tunnel traffic, where:
Step 4	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and enters privileged EXEC mode.

How to Provision MPLS Pseudowires

You can provision MPLS pseudowires in the following ways:

Note Before performing the static or dynamic provisioning of MPLS pseudowires, you must enable MPLS on a Cisco CMTS router.

Dynamic Provisioning of MPLS Pseudowires

The dynamic provisioning method supports the following types of configurations:

- BSOD Specification-Based MPLS Pseudowire Provisioning
- Type-4 MPLS Pseudowire Provisioning Using the CM Configuration File
- Type-5 MPLS Pseudowire Provisioning Using the CM Configuration File

See the Configuration Examples for Dynamic Provisioning of MPLS Pseudowires for details about the dynamic provisioning method using the CM configuration file.

Note We recommend that you use the dynamic provisioning method instead of the static provisioning method for MPLS pseudowires.

Static Provisioning Method for MPLS Pseudowires

The static provisioning method requires the MPLS pseudowire to be statically provisioned on the CMTS using the command line interface (CLI). This type of provisioning does not require the CM configuration file to use BSOD L2VPN-compliant TLVs. For details on how to statically provision MPLS pseudowires, see the *Static Provisioning of MPLS Pseudowires*.

How to Configure L2VPN Pseudowire Redundancy

The L2VPN Pseudowire Redundancy feature enables you to switch to backup pseudowires when the primary pseudowire fails. The feature also allows the Cisco CMTS to resume operation on the primary pseudowire after it comes back up.

Configuring the Backup Pseudowire

You can configure up to three backup pseudowires for a primary pseudowire. The priority of each backup pseudowire has to be unique.

A backup pseudowire is uniquely identified by a combination of IP address or hostname and VCID. Only the IP address or hostname and VCID can be configured for the backup peer, the remaining parameters are the same as the primary pseudowire.

Backup pseudowires can also be configured using the DOCSIS configuration files.

Perform the steps given below to configure a backup pseudowire.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**cable l2vpn** *mac-address* **Example:** `Router(config)# cable l2vpn 0011.0011.0011`	Specifies L2VPN MAC address and enters L2VPN *configuration mode.*
Step 4	**service instance** *id service-type* **Example:** `Router(config-l2vpn)# service instance 1 ethernet`	Specifies the service instance ID and enters Ethernet service configuration mode.

	Command or Action	Purpose
Step 5	**xconnect** *peer-ip-address vc-id* **encapsulation mpls** **Example:** `Router(config-ethsrv)# xconnect 10.2.2.2 22 encapsulation mpls`	Specifies the tunneling method to encapsulate the data in the MPLS pseudowire and enters xconnect configuration mode.
Step 6	**backup peer** *peer-ip-address vc-id* [priority value] **Example:** `Router(config-xconn)# backup peer 10.3.3.3 33 priority 2`	Specifies the backup pseudowire and its priority. The priority keyword is optional, if only one backup pseudowire is configured. When multiple backup pseudowires are configured, it is required.
Step 7	**end** **Example:** `Router(config-xconn)# end`	Exits xconnect configuration mode and enters Privileged EXEC mode.

Configuring Backup Delay

Perform the steps given below to configure the period the backup pseudowire should wait to take over after the primary pseudowire goes down. You can also specify how long the primary pseudowire should wait after it becomes active to take over from the backup pseudowire.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**cable l2vpn** *mac-address* **Example:** `Router(config)# cable l2vpn 0011.0011.0011`	Specifies the L2VPN MAC address and enters L2VPN *configuration mode.* • *mac-address*—MAC address of a CM.

	Command or Action	Purpose
Step 4	**service instance** *id service-type* **Example:** `Router(config-l2vpn)# service instance 1 ethernet`	Specifies the service instance ID and enters Ethernet service configuration mode. • *id*—Service instance ID. • service-type—Service type for the instance.
Step 5	**xconnect** *peer-ip-address vc-id* **encapsulation mpls** **Example:** `Router(config-ethsrv)# xconnect 10.2.2.2 22 encapsulation mpls`	Specifies the tunneling method to encapsulate the data in the MPLS pseudowire and enters xconnect configuration mode. • *peer-ip-address*—IP address of the remote PE router. The remote router ID can be any IP address, as long as it is reachable. • *vc-id*—32-bit identifier of the virtual circuit between the PE routers. • **encapsulation mpls**—Specifies MPLS as the tunneling method.

Step 6	Do one of the following:
	• **backup delay** enable-delay-period {disable-delay-period | never}
	•
	Example:
	`Router(config-xconn)# backup delay 10 10`
	Example:
	`Router(config-xconn)# backup delay 10 never`
		Specifies the period to wait before enabling or disabling the backup pseudowire.
		• *enable-delay-period*—Number of seconds the backup pseudowire should wait to take over after the primary pseudowire goes down. The valid range is from 0 to 180 seconds, with a default value of 0.
		• *disable-delay-period*—Number of seconds the primary pseudowire should wait after it becomes active to take over from the backup pseudowire. The valid range is from 0 to 180 seconds, with a default value of 0.
		• **never**—Specifies the primary pseudowire should not be reactivated after moving to the backup pseudowire.

Step 7	**end** **Example:** `Router(config-xconn)# end`	Exits xconnect configuration mode and enters privileged EXEC mode.

Performing Manual Switchover

Perform the steps given below to perform a manual switchover to the primary or backup pseudowire. The **xconnect backup force-switchover** command can also be used to forcefully switch to the backup pseudowire for planned outages of the primary remote peer.

Note A manual switchover can be made only to an available member in the redundancy group. If the pseudowire specified in the command is not available, the command will be rejected.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**xconnect backup force-switchover peer 10.10.1.1 123** **Example:** `Router# xconnect backup force-switchover peer 10.10.1.1 123`	Specifies that the router should switch to the backup or to the primary pseudowire.

Troubleshooting Tips

The following commands help you troubleshoot an improper MPLS pseudowire configuration:

- **show ip interface brief**—Helps verify that the loopback interface with the IP address is present on each PE router.
- **show mpls l2transport vc**—Helps verify information about primary and backup pseudowires that have been enabled to route Layer 2 packets on a router.
- **show xconnect all**—Helps verify information about all xconnect attachment circuits and primary and backup pseudowires.
- **show cable l2-vpn xconnect mpls-vc-map**—Helps verify that the primary and backup pseudowires are configured properly.

Configuration Examples for MPLS Pseudowire for Cable L2VPN

The following sections provide MPLS pseudowire configuration examples for the static and dynamic provisioning methods:

Configuration Example for Static Provisioning of MPLS Pseudowires

The following example shows CLI-based provisioning of an MPLS pseudowire:

```
Router> enable
Router# configure terminal
Router(config)# cable l2vpn 0000.396e.6a68 customer2
Router(config-l2vpn)# service instance 2000 ethernet
Router(config-ethsrv)# xconnect 101.1.0.2 221 encapsulation mpls pw-type 4
Router(config-ethsrv)# cable set mpls-experimental 7
```

Configuration Examples for Dynamic Provisioning of MPLS Pseudowires

The following sections provide MPLS pseudowire provisioning examples based on BSOD CableLabs specification, Type-4, and Type-5 TLVs using the CM configuration file:

BSOD Specification-Based MPLS Pseudowire Provisioning: Example

The following example shows an MPLS pseudowire configuration based on BSOD CableLabs specification:

```
03 (Net Access Control) = 1
43 (Vendor Specific Options)
   S08 (Vendor ID) = ff ff ff
   S005 (L2VPN sub-type)
 =
        T01 (VPN Id) = 02 34 56 00 02 # VPNID=0234650002
        T02 (NSI) = 04 05 01 0a 4c 01 01# [04=mpls] [05=len][01=ipv4][IP=10.76.1.1]
        T05 (AGI) = 01 01 07 d1 # AGI = 0x010107d1
        T06 (SAII) = 00 00 07 d1 # SAII = TAII = VCID = 0x7d1 = 2001
        T07 (TAII) = 00 00 07 d1
18 (Maximum Number of CPE) = 16
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 1
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type)  =
            S01 (VPNID) =   02 34 56 00 02
            S08 (UserPrio) = 01

24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 2
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type) =
            S01 (VPNID) = 02 34 56 00 02
            S08 (UserPrio) = 04
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 3
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type) =
            S01 (VPNID) =   02 34 56 00 02
            S08 (UserPrio) = 05
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 4
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type) =
            S01 (VPNID) = 02 34 56 00 02
            S08 (UserPrio) = 06
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 2
   S03 (Service Flow Reference) = 2
   S05 (Rule Priority) = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask) = 00 20 ff
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 3
```

```
   S03 (Service Flow Reference) = 3
   S05 (Rule Priority) = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask) = 21 40 ff
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 4
   S03 (Service Flow Reference) = 4
   S05 (Rule Priority) = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)  = 41 ff ff
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 11
   S06 (QoS Parameter Set Type) = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 12
   S06 (QoS Parameter Set Type) = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 13
   S06 (QoS Parameter Set Type) = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 14
   S06 (QoS Parameter Set Type) = 7
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 12
   S03 (Service Flow Reference) = 12
   S05 (Rule Priority) = 3
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T01 (IEEE 802.1P UserPriority)   = 00 02
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type)

              S01 (VPNID) = 02 34 56 00 02
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 13
   S03 (Service Flow Reference) = 13
   S05 (Rule Priority) = 3
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T01 (IEEE 802.1P UserPriority) = 03 04
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type)
              S01 (VPNID) = 02 34 56 00 02
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 14
   S03 (Service Flow Reference) = 14
   S05 (Rule Priority) = 3
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T01 (IEEE 802.1P UserPriority) = 05 06
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T005 (L2VPN sub-type)
              S01 (VPNID) = 02 34 56 00 02
```

Type-4 MPLS Pseudowire Provisioning Using the CM Configuration File: Example

The following example shows a CM configuration file-based provisioning of a Type-4 MPLS pseudowire:

```
03 (Net Access Control) = 1
43 (Vendor Specific Options)
   S08 (Vendor ID) = ff ff ff
   S005 (L2VPN Options) =
```

```
T001 (VPN ID) = 02 34 56 00 02 # VPN-ID = "0234560002"
T043 (Cisco Vendor Specific) = 2b 16
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO
S036 (MPLSPWTYPE) = 24 01 04 # MPLSPWTYPE= Type4 - Ethernet-vlan Type
S039 (MPLSPEERNAME) = 27 06 63 37 36 30 30 32 MPLSPEERNAME= "c76002" in ascii
S038 (MPLSVCID) = 26 04 00 00 07 d1 = 2001 VCID
43 (Vendor Specific Options)
   S08 (Vendor ID) = ff ff ff
   S005 (L2VPN Options) =
T001 (VPN ID) = 02 34 56 00 03 # VPN-ID = "0234560003"
T043 (Cisco Vendor Specific) =  2b 16
S008 (Vendor ID) = 00 00 0c Vendor ID = "00 00 0C" - CISCO
S036 (MPLSPWTYPE) = 24 01 04 MPLSPWTYPE= Type4 - Ethernet-vlan Type
S039 (MPLSPEERNAME) = 27 06 63 37 36 30 30 32 # MPLSPEERNAME= "c76002" in ascii
S038 (MPLSVCID) = 26 04 00 00 0b b9 # = 3001 VCID
43 (Vendor Specific Options)
   S08 (Vendor ID) = ff ff ff
   S005 (L2VPN Options) =
T001 (VPN ID) = 02 34 56 00 04 # VPN-ID = "0234560004"
T043 (Cisco Vendor Specific) = 2b 16
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO
S036 (MPLSPWTYPE) = 24 01 04 # MPLSPWTYPE= Type4 - Ethernet-vlan Type
S039 (MPLSPEERNAME) = 27 06 63 37 36 30 30 32 # MPLSPEERNAME= "c76002" in ascii
S038 (MPLSVCID) = 26 04 00 00 0f a1 # = 4001 VCID
18 (Maximum Number of CPE) = 16
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 1
   S06 (QoS Parameter Set Type) = 7
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 2
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) = 02 34 56 00 02
       T043 (Cisco Vendor Specific) = 2b 0A
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO

S034 (MPLS-EXP-SET) = 22 05 # MPLSEXP-INGRESS= 5
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 3
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) = 02 34 56 00 03
       T043 (Cisco Vendor Specific) =  2b 0A
S008 (Vendor ID) = 00 00 0c
# Vendor ID = "00 00 0C" - CISCO

S034 (MPLS-EXP-SET) = 22 06

# MPLSEXP-INGRESS= 6
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 4
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) = 02 34 56 00 04
       T043 (Cisco Vendor Specific) = 2b 0A
S008 (Vendor ID) = 00 00 0c
# Vendor ID = "00 00 0C" - CISCO

S034 (MPLS-EXP-SET) = 22 04

# MPLSEXP-INGRESS= 4
```

```
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 2
   S03 (Service Flow Reference) = 2
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T02 (IEEE 802.1Q VLAN ID) = 7d 00
   S05 (Rule Priority) = 2
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 3
   S03 (Service Flow Reference) = 3
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T02 (IEEE 802.1Q VLAN ID) = bb 80
   S05 (Rule Priority) = 3
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 4
   S03 (Service Flow Reference) = 4
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T02 (IEEE 802.1Q VLAN ID) = fa 00
   S05 (Rule Priority) = 4
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 11
   S06 (QoS Parameter Set Type) = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 12
   S06 (QoS Parameter Set Type) = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 13
   S06 (QoS Parameter Set Type) = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 14
   S06 (QoS Parameter Set Type) = 7
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 12
   S03 (Service Flow Reference) = 12
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T02 (IEEE 802.1Q VLAN ID) = 7d 00
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) = 02 34 56 00 02
       T043 (Cisco Vendor Specific) = 2b 0B
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO
S035 (MPLS-EXP_RANGE) = 23 02  03 # MPLSEXP-EGRESS_RANGE= 2 - 3
   S05 (Rule Priority) = 2
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 13
   S03 (Service Flow Reference) = 13
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T02 (IEEE 802.1Q VLAN ID) = bb 80
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) =  02 34 56 00 03
       T043 (Cisco Vendor Specific) = 2b 0B
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO

S035 (MPLS-EXP-RANGE) = 23 04  05 # MPLSEXP-EGRESS_RANGE= 4 - 5
   S05 (Rule Priority = 3
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference) = 14
   S03 (Service Flow Reference) = 14
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
       T02 (IEEE 802.1Q VLAN ID) = fa 00
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) = 02 34 56 00 04
       T043 (Cisco Vendor Specific) = 2b 0B
```

```
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO

S035 (MPLS-EXP-RANGE) = 23 00  01 # MPLSEXP-EGRESS_RANGE= 0 - 1
   S05 (Rule Priority) = 4
```

Type-5 MPLS Pseudowire Provisioning Using the CM Configuration File: Example

The following example shows a CM configuration file-based provisioning of a Type-5 MPLS pseudowire:

```
03 (Net Access Control) = 1
43 (Vendor Specific Options)
   S08 (Vendor ID) = ff ff ff
   S005 (L2VPN Options) =
T001 (VPN ID) = 02 34 56 00 02 # VPN-ID = "0234560002"
T043 (Cisco Vendor Specific) = 2b 16
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO
S036 (MPLSPWTYPE) = 24 01 05 # MPLSPWTYPE= Type5 - Ethernet-Port Type
S039 (MPLSPEERNAME) = 27 06 63 37 36 30 30 32 # MPLSPEERNAME= "c76002" in ascii
S038 (MPLSVCID) = 26 04 00 00 07 d1 # = 2001 VCID
45 (L2VPN CMIM) = 02 04 ff ff ff ff 01 01 01
18 (Maximum Number of CPE) = 16
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference) = 1
   S06 (QoS Parameter Set Type) = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID) = ff ff ff
       T001 (VPN ID) = 02 34 56 00 02 # VPN-ID = "0234560002"
       T043 (Cisco Vendor Specific) = 2b 0A
S008 (Vendor ID) = 00 00 0c # Vendor ID = "00 00 0C" - CISCO
S034 (MPLS-EXP-SET) = 22 04 # MPLS-EXP-SET at INGRESS= 4
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference) = 12
   S06 (QoS Parameter Set Type) = 7
```

Configuration Examples for L2VPN Pseudowire Redundancy

The following sections provide L2VPN pseudowire redundancy configuration examples using the CM configuration file:

Example: Configuring Backup Pseudowire Peer and VC ID

The following example shows how to provision a file-based backup peer router based on the CM configuration:

PE Router 1

```
cable l2vpn 0025.2e2d.7252
 service instance 1 ethernet
  encapsulation default
  xconnect 10.76.2.1 400 encapsulation mpls
   backup peer 10.76.2.1 600 priority 4
```

PE Router2

```
cable l2vpn 0011.0011.0011
 service instance 1 ethernet
  encapsulation default
```

```
  xconnect 10.2.2.2 22 encapsulation mpls
   backup peer 10.3.3.3 33 priority 2
   backup delay 10 10
```

Example: Configuring Backup Delay

The following example shows how to configure a backup delay to determine how much time should elapse before a secondary line status change after a primary line status has been changed.

```
cable l2vpn 0011.0011.0011
 service instance 1 ethernet
  encapsulation default
  xconnect 10.2.2.2 22 encapsulation mpls
   backup delay 10 10
```

Example: L2VPN Backup MPLS Pseudowire Provisioning Using the CM Configuration File

The following example shows how to provision an L2VPN Backup MPLS pseudowire based on the CM configuration file:

```
03 (Net Access Control)         = 1
18 (Maximum Number of CPE)      = 3
43 (Vendor Specific Options)
   S08 (Vendor ID)      = ff ff ff
   S005 (Unknown sub-type)      = 01 04 32 30 32 30 02 07 04 05 01 0a 4c 02 01 2b 15 26 04
 00 00 00 14 28 10 01 05 01 0a 4c 02 01 03 04 00 00 07 08 04 01 05 28 0d 01 05 01 0a 4c 02
 03 03 04 00 00 00 15 28 10 01 05 01 0a 4c 02 01 03 04 00 00 b1 8e 04 01 01 29 01 03 2a 01
 01
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 4
   S06 (QoS Parameter Set Type)         = 7
   S08 (Max Sustained Traffic Rate)     = 2000000
   S09 (Max Traffic Burst)              = 3200
   S15 (Service Flow Sched Type)        = 2
   S43 (Vendor Specific Options)
       T08 (Vendor ID)      = ff ff ff
       T005 (Unknown sub-type)      = 01 04 32 30 32 30
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)         = 2
   S06 (QoS Parameter Set Type)         = 7
   S08 (Max Sustained Traffic Rate)     = 3000000
   S09 (Max Traffic Burst)              = 250000
29 (Privacy Enable)             = 1
```

Verifying the MPLS Pseudowire Configuration

Use the following **show** commands to verify the MPLS pseudowire configuration:

- **show mpls ldp discovery**
- **show cable l2-vpn xconnect**
- **show xconnect**
- **show mpls l2transport vc**

To verify the mapping between the MPLS pseudowire and virtual circuits for all cable modems, use the **show cable l2-vpn xconnect** command as shown in the following example:

```
Router# show cable l2-vpn xconnect mpls-vc-map
MAC Address     Peer IP Address VCID Type Prio CktID     Cable Intf  SID Customer Name/VPNID
0023.bee1.eb48  123.1.1.1       30   Prim*     Bu254:4101 Cable3/0/0  3
38c8.5cac.4a62  123.1.1.1       20   Prim*     Bu254:4100 Cable3/0/0  4   customer1
602a.d083.2e1c  123.1.1.1       60   Prim*     Bu254:4102 Cable3/0/0  5
```

To verify the mapping between the MPLS pseudowire and virtual circuits for all cable modems when pseudowire redundancy is not configured, use the **show cable l2-vpn xconnect mpls-vc-map** command as shown in the following example:

```
Router# show cable l2-vpn xconnect mpls-vc-map
MAC Address    Peer IP Address VCID Type  Prio CktID     Cable Intf SID Customer Name/VPNID
0025.2e2d.7252 10.76.2.1       400  Prim*     Bu254:400  Cable8/0/3 1
0014.f8c1.fd46 10.2.3.4        1000 Prim*     Bu254:1000 Cable8/0/0 1   2020
0014.f8c1.fd46 10.76.2.1       1800 Prim*     Bu254:1800 Cable8/0/0 1   2021
```

To verify the mapping between the MPLS pseudowire and virtual circuits for all cable modems when pseudowire redundancy is configured, use the **show cable l2-vpn xconnect mpls-vc-map** command as shown in the following example:

```
Router# show cable l2-vpn xconnect mpls-vc-map
MAC Address    Peer IP Address VCID Type  Prio CktID     Cable Intf SID Customer Name/VPNID
602a.d083.2e1c 123.1.1.1       60   Prim*     Bu254:4102 Cable3/0/0 5
38c8.5cac.4a62 123.1.1.1       20   Prim*     Bu254:4103 Cable3/0/0 4   000232303230
               156.1.3.1       30   Bkup  3   Bu254:4103
               123.1.1.1       50   Bkup  8   Bu254:4103
38c8.5cac.4a62 156.1.3.1       56   Prim*     Bu254:4104 Cable3/0/0 4   000232303231
               123.1.1.1       40   Bkup  1   Bu254:4104
```

To obtain the state of all virtual circuits associated with an MPLS pseudowire when pseudowire redundancy is not configured, use the **show cable l2-vpn xconnect mpls-vc-map state** command as shown in the following example:

```
Router# show cable l2-vpn xconnect mpls-vc-map state
MAC Address      Peer IP Address VCID Type Prio State Customer Name/VPNID State
602a.d083.2e1c   123.1.1.1       60   Prim*    UP                        UP
38c8.5cac.4a62   123.1.1.1       20   Prim*    UP    000232303230        UP
38c8.5cac.4a62   156.1.3.1       56   Prim*    UP    000232303231        UP
```

To obtain the state of all virtual circuits associated with an MPLS pseudowire when pseudowire redundancy is configured, use the **show cable l2-vpn xconnect mpls-vc-map state** command as shown in the following example:

```
Router# show cable l2-vpn xconnect mpls-vc-map state
MAC Address      Peer IP Address VCID Type   Prio State Customer Name/VPNID State
602a.d083.2e1c   123.1.1.1       60   Prim*       UP                        UP
38c8.5cac.4a62   123.1.1.1       20   Prim*       UP    000232303230        UP
                 156.1.3.1       30   Bkup   3    UP    000232303230        STDBY
                 123.1.1.1       50   Bkup   8    DOWN  000232303230        STDBY
38c8.5cac.4a62   156.1.3.1       56   Prim*       UP    000232303231        UP
                 123.1.1.1       40   Bkup   1    UP    000232303230        STDBY

When the local state of the modem is DOWN, the L2VPN is not configured on the WAN interface
 and the remote state of the L2VPN will be shown as OFF.
```

```
Router#show cable l2-vpn xconnect mpls-vc-map state
MAC Address     Peer IP Address VCID Type   Prio State Customer Name/VPNID State
602a.d083.2e1c  123.1.1.1       60   Prim*       OFF               DOWN
38c8.5cac.4a62  123.1.1.1       20   Prim*       UP    000232303230 UP
38c8.5cac.4a62  156.1.3.1       56   Prim*       UP    000232303231 UP
```

To verify information about the MPLS pseudowire mapping for a particular MAC address of a CM when pseudowire redundancy is configured, use the **show cable l2-vpn xconnect mpls-vc-map** command as shown in the following example:

```
Router# show cable l2-vpn xconnect mpls-vc-map 0025.2e2d.7252
MAC Address    Peer IP Address VCID Type  Prio CktID     Cable Intf SID Customer Name/VPNID
0025.2e2d.7252 10.76.2.1       400  Prim*      Bu254:400 Cable8/0/3 1
               10.76.2.1       600  Bkup  4    Bu254:600
```

To verify the detailed information about the MPLS pseudowire mapping for a CM when pseudowire redundancy is configured, use the **show mpls l2-vpn xconnect mpls-vc-map verbose** command as shown in the following examples.

The following example shows the information for a modem for which pseudowires were configured using backup peer command:

```
Router# show cable l2-vpn xconnect mpls-vc-map 0025.2e2d.7252 verbose
MAC Address                         : 0025.2e2d.7252
Customer Name                       :
Prim Sid                            : 1
Cable Interface                     : Cable8/0/3
MPLS-EXP                            : 0
PW TYPE                             : Ethernet
Backup enable delay                 : 0 seconds
Backup disable delay                : 0 seconds
Primary peer
Peer IP Address (Active)            : 10.76.2.1
XConnect VCID                       : 400
Circuit  ID                         : Bu254:400
Local State                         : UP
Remote State                        : UP
Backup peers
Peer IP Address                     : 10.76.2.1
XConnect VCID                       : 600
Circuit  ID                         : Bu254:600
Local State                         : STDBY
Remote State                        : UP
Priority                            : 4
Total US pkts                       : 0
Total US bytes                      : 0
Total US pkts   discards            : 0
Total US bytes  discards            : 0
Total DS pkts                       : 0
Total DS bytes                      : 0
Total DS pkts   discards            : 0
Total DS bytes  discards            : 0
```

The following example shows the information for a modem for which pseudowires were created using the modem configuration file:

```
Router# show cable l2-vpn xconnect mpls-vc-map 0014.f8c1.fd46 verbose
MAC Address                         : 0014.f8c1.fd46
```

```
Prim Sid                            : 3
Cable Interface                     : Cable8/0/0
L2VPNs provisioned                  : 1
DUT Control/CMIM                    : Disable/0x8000FFFF
VPN ID                              : 2020
L2VPN SAID                          : 12289
Upstream SFID Summary               : 15
Downstream CFRID[SFID] Summary      : Primary SF
CMIM                                : 0x60
PW TYPE                             : Ethernet
MPLS-EXP                            : 0
Backup enable delay                 : 3 seconds
Backup disable delay                : 1 seconds
Primary peer
Peer IP Address (Active)            : 10.2.3.4
XConnect VCID                       : 1000
Circuit  ID                         : Bu254:1000
Local State                         : UP
Remote State                        : UP

Backup peers
Peer IP Address                     : 10.2.3.4
XConnect VCID                       : 21
Circuit  ID                         : Bu254:21
Local State                         : STDBY
Remote State                        : DOWN
Priority                            : 2
Peer IP Address                     : 10.76.2.1
XConnect VCID                       : 1800
Circuit  ID                         : Bu254:1800
Local State                         : STDBY
Remote State                        : DOWN
Priority                            : 5
Peer IP Address                     : 10.76.2.1
XConnect VCID                       : 45454
Circuit  ID                         : Bu254:45454
Local State                         : STDBY
Remote State                        : DOWN
```

To verify information about all attachment circuits and pseudowires for online modems, use the **show xconnect** command as shown in the following example:

```
Router# show xconnect all
Legend:    XC ST=Xconnect State  S1=Segment1 State  S2=Segment2 State
  UP=Up       DN=Down           AD=Admin Down      IA=Inactive
  SB=Standby  RV=Recovering     NH=No Hardware
XC ST  Segment 1                         S1 Segment 2                          S2
------+---------------------------------+--+---------------------------------+--
UP     ac   Bu254:2001(DOCSIS)           UP mpls 10.76.1.1:2001                UP
UP     ac   Bu254:2002(DOCSIS)           UP mpls 10.76.1.1:2002                UP
UP     ac   Bu254:2004(DOCSIS)           UP mpls 10.76.1.1:2004                UP
DN     ac   Bu254:22(DOCSIS)             UP mpls 101.1.0.2:22                  DN
```

To verify information about MPLS virtual circuits and static pseudowires that have been enabled to route Layer 2 packets on a Cisco CMTS router, use the **show mpls l2transport vc** command as shown in the following example:

```
Router# show mpls l2transport vc
Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Bu254          DOCSIS 2002                10.76.1.1       2002       UP
```

```
Bu254          DOCSIS 2003               10.76.1.1       2003       UP
Bu254          DOCSIS 2004               10.76.1.1       2004       DOWN
Bu254          DOCSIS 2017               10.76.1.1       2017       UP
Bu254          DOCSIS 2018               10.76.1.1       2018       UP
Bu254          DOCSIS 2019               10.76.1.1       2019       UP
```

Additional References

Standards

Standard	Title
CM-SP-L2VPN-I08-080522	*Business Services over DOCSIS (BSOD) Layer 2 Virtual Private Networks*
L2VPN-N-10.0918-2	*L2VPN MPLS Update*

MIBs

MIB	MIBs Link
• DOCS-L2VPN-MIB • CISCO-IETF-PW-MIB • CISCO-CABLE-L2VPN-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://tools.cisco.com/ITDIT/MIBS/servlet/index

RFCs

RFC	Title
RFC 3985	*Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture*
RFC 4385	*Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN*
RFC 4446	*IANA Allocations for Pseudowire Edge-to-Edge Emulation (PWE3)*
RFC 4447	*Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*
RFC 4448	*Encapsulation Methods for Transport of Ethernet over MPLS Networks*
RFC 5085	*Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

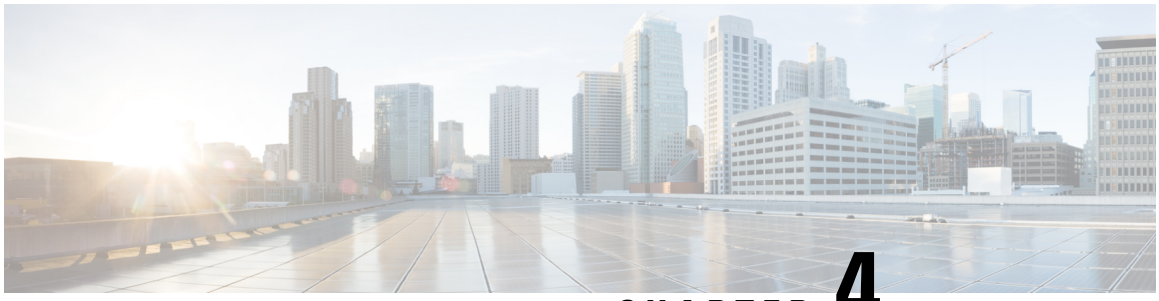
Feature Information for MPLS Pseudowire for Cable L2VPN

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 8: Feature Information for MPLS Pseudowire for Cable L2VPN

Feature Name	Releases	Feature Information
MPLS Pseudowire for Cable L2VPN	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on theCisco cBR Series Converged Broadband Routers.



CHAPTER 4

MPLS VPN Cable Enhancements

This feature module describes the Multiprotocol Label Switching Virtual Private Network (MPLS VPN) and cable interface bundling features. It explains how to create a VPN using MPLS protocol, cable interfaces, bundle interfaces and sub bundle interfaces. VPNs can be created in many ways using different protocols.

Finding Feature Information

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 9: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Feature Overview

Using MPLS VPN technology, service providers can create scalable and efficient private networks using a shared hybrid fiber coaxial (HFC) network and Internet protocol (IP) infrastructure.

The cable MPLS VPN network consists of:

- The Multiple Service Operator (MSO) or cable company that owns the physical infrastructure and builds VPNs for the Internet Service Providers (ISPs) to move traffic over the cable and IP backbone.
- ISPs that use the HFC network and IP infrastructure to supply Internet service to cable customers.

Each ISP moves traffic to and from a subscriber's PC, through the MSO's physical network infrastructure, to the ISP's network. MPLS VPNs, created in Layer 3, provide privacy and security by constraining the distribution

of a VPN's routes only to the routers that belong to its network. Thus, each ISP's VPN is insulated from other ISPs that use the same MSO infrastructure.

An MPLS VPN assigns a unique VPN Routing/Forwarding (VRF) instance to each VPN. A VRF instance consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine the contents of the forwarding table.

Each PE router maintains one or more VRF tables. It looks up a packet's IP destination address in the appropriate VRF table, only if the packet arrived directly through an interface associated with that table.

MPLS VPNs use a combination of BGP and IP address resolution to ensure security. See *Configuring Multiprotocol Label Switching.*

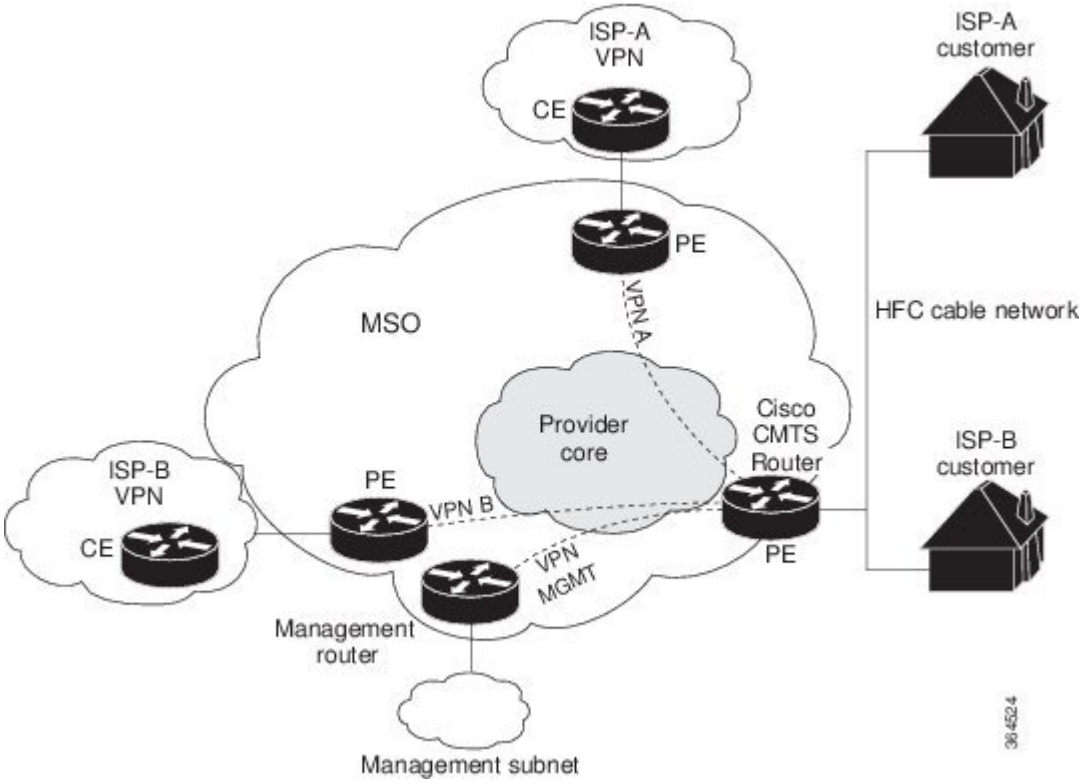
The table shows a cable MPLS VPN network. The routers in the network are:

- Provider (P) router—Routers in the core of the provider network. P routers run MPLS switching, and do not attach VPN labels (MPLS label in each route assigned by the PE router) to routed packets. VPN labels are used to direct data packets to the correct egress router.
- Provider Edge (PE) router— Router that adds the VPN label to incoming packets based on the interface or subinterface on which they are received. A PE router attaches directly to a CE router. In the MPLS-VPN approach, each Cisco CMTS router acts as a PE router.
- Customer (C) router—Router in the ISP or enterprise network.
- Customer Edge (CE) router—Edge router on the ISP's network that connects to the PE router on the MSO's network. A CE router must interface with a PE router.

The MPLS network has a unique VPN that exclusively manages the MSOs devices called the management VPN. It contains servers and devices that other VPNs can access. The management VPN connects the Cisco CMTS router to a PE router, which connects to management servers such as Cisco Network Registrar (CNR) and Time of Day (ToD) servers. A PE router connects to management servers and is a part of the management VPN. Regardless of the ISP they belong to, the management servers serve the Dynamic Host Configuration Protocol (DHCP), DNS (Domain Name System), and TOD requests coming from PCs or cable modems.

Note When configuring MPLS VPNs, you must configure the first subinterface created as a part of the management VPN.

Figure 3: MPLS VPN Network



Cable VPN configuration involves an:

- MSO domain that requires a direct peering link to each enterprise network (ISP), provisioning servers for residential and commercial subscribers, and dynamic DNS for commercial users. The MSO manages cable interface IP addressing, Data-over-Cable Service Interface Specifications (DOCSIS) provisioning, CM hostnames, routing modifications, privilege levels, and usernames and passwords.
- ISP or enterprise domain that includes the DHCP server for subscriber or telecommuter host devices, enterprise gateway within the MSO address space, and static routes back to the telecommuter subnets.

Note Cisco recommends that the MSO assign all addresses to the end user devices and gateway interfaces. The MSO can also use split management to let the ISP configure tunnels and security.

In an MPLS VPN configuration, the MSO must configure the following:

- CMTS
- P routers
- PE routers
- CE routers
- One VPN per ISP DOCSIS servers for all cable modem customers. The MSO must attach DOCSIS servers to the management VPN, and make them visible.

The MSO must configure the Cisco CMTS routers that serve the ISP, and remote PE routers connecting to the ISP, as PE routers in the VPN.

The MSO must determine the primary IP address range for all cable modems.

The ISP must determine the secondary IP address range for subscriber PCs.

To reduce security breaches and differentiate DHCP requests from cable modems in VPNs or under specific ISP management, MSOs can use the **cable helper-address** command in Cisco IOS-XE software. The MSO can specify the host IP address to be accessible only in the ISP's VPN. This lets the ISP use its DHCP server to allocate IP addresses. Cable modem IP address must be accessible from the management VPN.

The MPLS VPN approach of creating VPNs for individual ISPs or customers requires subinterfaces to be configured on the virtual bundle interface. Each ISP requires one subinterface. The subinterfaces are tied to the VPN Routing/Forwarding (VRF) tables for their respective ISPs. The first subinterface must be created on the cable interface bound to the management VPN.

To route a reply from the CNR back to the cable modem, the PE router that connects to the CNR must import the routes of the ISP VPN into the management VPN. Similarly, to forward management requests (such as DHCP renewal to CNR) to the cable modems, the ISP VPN must export and import the appropriate management VPN routes.

You can group all of the cable interfaces on a Cisco CMTS router into a single bundle so that only one subnet is required for each router. When you group cable interfaces, no separate IP subnet or each individual cable interface is required. This grouping avoids the performance, memory, and security problems in using a bridging solution to manage subnets, especially for a large number of subscribers.

Subinterfaces allow traffic to be differentiated on a single physical interface, and assigned to multiple VPNs. You can configure multiple subinterfaces, and associate an MPLS VPN with each subinterface. You can split a single physical interface (the cable plant) into multiple subinterfaces, where each subinterface is associated with a specific VPN. Each ISP requires access on a physical interface and is given its own subinterface. Create a management subinterface to support cable modem initialization from an ISP.

Using each subinterface associated with a specific VPN (and therefore, ISP) subscribers connect to a logical subinterface, which reflects the ISP that provides their subscribed services. When properly configured, subscriber traffic enters the appropriate subinterface and VPN.

Benefits

- MPLS VPNs give cable MSOs and ISPs a manageable way of supporting multiple access to a cable plant. Service providers can create scalable and efficient VPNs across the core of their networks. MPLS VPNs provide systems support scalability in cable transport infrastructure and management.
- Each ISP can support Internet access services from a subscriber's PC through an MSO's physical cable plant to their networks.
- MPLS VPNs allow MSOs to deliver value-added services through an ISP, and thus, deliver connectivity to a wider set of potential customers. MSOs can partner with ISPs to deliver multiple services from multiple ISPs and add value within the MSO's own network using VPN technology.
- Subscribers can select combinations of services from various service providers.
- The MPLS VPN cable features set build on CMTS DOCSIS 1.0 and DOCSIS 1.0 extensions to ensure services are reliably and optimally delivered over the cable plant. MPLS VPN provides systems support domain selection, authentication per subscriber, selection of QoS, policy-based routing, and ability to reach behind the cable modem to subscriber end devices for QoS and billing while preventing session spoofing.
- MPLS VPN technology ensures both secure access across the shared cable infrastructure and service integrity.

- Cable interface bundling eliminates the need for an IP subnet on each cable interface. Instead, an IP subnet is only required for each cable interface bundle. All cable interfaces in a Cisco CMTS router can be added to a single bundle.

Restrictions

- Each subinterface on the CMTS requires an address range from the ISP and from the MSO. These two ranges must not overlap and must be extensible to support an increased number of subscribers for scalability.

Note This document does not address allocation and management of MSO and ISP IP addresses. See *Configuring Multiprotocol Label Switching* for this information.

- The **cable source-verify dhcp** command enables Dynamic Host Control Protocol (DHCP) Lease query protocol from the CMTS to DHCP server to verify IP addresses of upstream traffic, and prevent MSO customers from using unauthorized, spoofed, or stolen IP addresses.
- When using only MPLS VPNs, create subinterfaces on the virtual bundle, assign it an IP address, and provide VRF configuration for each ISP. When you create subinterfaces and configure only MPLS VPNs, the cable interface bundling feature is independent of the MPLS VPN.
- When using cable interface bundling:
  - Define a virtual bundle interface and associate any cable physical interface to the virtual bundle.
  - Specify all generic IP networking information (such as IP address, routing protocols, and switching modes) on the virtual bundle interface. Do not specify generic IP networking information on bundle subsidiary interfaces.
  - An interface that has a subinterface(s) defined over it is not allowed to be a part of the bundle.
  - Specify generic (not downstream or upstream related) cable interface configurations, such as source-verify or ARP handling, on the virtual bundle interface. Do not specify generic configuration on bundle subsidiary interfaces.
- Interface bundles can only be configured using the command line interface (including the CLI-based HTML configuration).

Prerequisites

Before configuring IP-based VPNs, complete the following tasks:

- Ensure your network supports reliable broadband data transmission. Your plant must be swept, balanced, and certified based on National Television Standards Committee (NTSC) or appropriate international cable plant recommendations. Ensure your plant meets all DOCSIS or European Data-over-Cable Service Interface Specifications (EuroDOCSIS) downstream and upstream RF requirements.
- Ensure your Cisco router is installed following instructions in the Hardware Installation Guide and the Regulatory Compliance and Safety Information guide.
- Ensure your Cisco router is configured for basic operations.

- The chassis must contain at least one port adapter to provide backbone connectivity and one Cisco cable modem card to serve as the RF cable TV interface.

Other Important Information

- Ensure all other required headend or distribution hub routing and network interface equipment is installed, configured, and operational based on the services to support. This includes all routers, servers (DHCP, TFTP, and ToD), network management systems, other configuration or billing systems and backbone, and other equipment to support VPN.
- Ensure DHCP and DOCSIS configuration files have been created and pushed to appropriate servers such that each cable modem, when initialized, can transmit a DHCP request, receive an IP address, obtain TFTP and ToD server addresses, and download a DOCSIS configuration file. Configure each subinterface to connect to the ISP's VPN.
- Ensure DOCSIS servers are visible on the management VPN.
- Be familiar with your channel plan to assign appropriate frequencies. Outline your strategies for setting up bundling or VPN solution sets if applicable to your headend or distribution hub. Obtain passwords, IP addresses, subnet masks, and device names as appropriate.
- Create subinterfaces off of a virtual bundle interface. Configure each subinterface to connect to the ISP network.

The MPLS VPN configuration steps assume the following:

- IP addressing has already been determined and there are assigned ranges in the MSO and ISP network for specific subinterfaces.
- The MSO is using CNR and has configured it (using the **cable helper-address** command) to serve appropriate IP addresses to cable modems based on the cable modem MAC address. The CMTS forwards DHCP requests to the CNR based on the **cable helper-address** settings. The CNR server determines the IP address to assign the cable modem using the client-classes feature, which let the CNR assign specific parameters to devices based on MAC addresses.
- ISP CE routers are configured (using the **cable helper-address** command) to appropriately route relevant IP address ranges into the VPN.
- P and PE routers are already running Cisco Express Forwarding (CEF).
- MPLS is configured on the outbound VPN using the **tag switching ip** command in interface configuration mode.

Configuration Tasks

To configure MPLS VPNs, perform the following tasks:

Creating VRFs for each VPN

To create VRFs for each VPN, perform the following steps beginning in the router configuration mode.

Note Since only the CMTS has logical subinterfaces, assignments of VRFs on the other PE devices will be to specific physical interfaces.

Procedure

	Command or Action	Purpose
Step 1	Router(config)# **vrf definition** *mgmt-vpn*	Enters VRF configuration mode (config-vrf)# and maps a VRF table to the VPN (specified by *mgmt-vpn*). The management VPN is the first VPN configured.
Step 2	Router(config-vrf)# **rd** *mgmt-rd*	Creates a routing and forwarding table by assigning a route distinguisher to the management VPN.
Step 3	Router(config-vrf)# **route-target** {**export**| **import**| **both**} *mgmt-rd*	Exports and/or imports all routes for the management VPNs route distinguisher. This determines which routes will be shared within VRFs.
Step 4	Router(config-vrf)# **route-target import** isp1-vpn-rd	Imports all routes for the VPNs (*isp1-vpn)* route distinguisher.
Step 5	Router(config-vrf)# **route-target import** isp2-vpn-rd	Imports all routes for the VPNs (*isp2-vpn)* route distinguisher.
Step 6	Router(config-vrf)# **vrf definition** isp1-vpn	Creates a routing and forwarding table by assigning a route distinguisher to *isp1-vpn* .
Step 7	Router(config-vrf)# **rd** *mgmt-rd*	Creates a routing and forwarding table by assigning a route distinguisher (mgmt-rd) to the management VPN (mgmt-vpn).
Step 8	Router(config-vrf)# **route-target export** isp1-vpn-rd	Exports all routes for the VPNs (*isp1-vpn)* route distinguisher.
Step 9	Router(config-vrf)# **route-target import** isp1-vpn-rd	Imports all routes for the VPNs (*isp1-vpn)* route distinguisher.
Step 10	Router(config-vrf)# **route-target import** mgmt-vpn-rd	Exports all routes for the VPNs (*mgmt-vpn)* route distinguisher.
Step 11	Router(config-vrf)# **vrf definition** isp2-vpn	Creates a routing and forwarding table by assigning a route distinguisher to *isp2-vpn* .
Step 12	Router(config-vrf)# **route-target export** isp2-vpn-rd	Exports all routes for the VPNs (*isp2-vpn)* route distinguisher.
Step 13	Router(config-vrf)# **route-target import** isp2-vpn-rd	Imports all routes for the VPNs (*isp2-vpn)* route distinguisher.
Step 14	Router(config-vrf)# **route-target import** mgmt-vpn-rd	Imports all routes for the VPNs (*mgmt-vpn)* route distinguisher.

Defining Subinterfaces on a Virtual Bundle Interface and Assigning VRFs

To create a logical cable subinterface, perform the following steps beginning in the global configuration mode. Create one subinterface for each VPN (one per ISP). The first subinterface created must be configured as part of the management VPN (with the lowest subinterface number).

Procedure

	Command or Action	Purpose
Step 1	Router# **configure terminal**	Enters configuration mode.
Step 2	Router(config)# **interface bundle n.***x*	Enters virtual bundle interface configuration mode and defines the first (management) subinterface with the lowest subinterface number.
Step 3	Router(config-subif)# **description** *string*	Identifies the subinterface as the management subinterface.
Step 4	Router(config-subif)# **vrf forwarding** *mgmt-vpn*	Assigns the subinterface to the management VPN (the MPLS VPN used by the MSO to supply service to customers).
Step 5	Router(config-subif)# **ip address** *ipaddress mask*	Assigns the subinterface an IP address and a subnet mask.
Step 6	Router(config-subif)# **cable helper-address** *ip-address* **cable-modem**	Forwards DHCP requests from cable modems to the IP address listed.
Step 7	Router(config-subif)# **cable helper-address** *ip-address* **host**	Forwards DHCP requests from hosts to the IP address listed.
Step 8	Router(config-if)# **interface bundle n.***x*	Defines an additional subinterface for the ISP (such as isp1).
Step 9	Router(config-subif)# **description** *string*	Identifies the subinterface (such as subinterface for *isp1-vpn)* .
Step 10	Router(config-subif)# **vrf forwarding** *isp1-vpn*	Assigns the subinterface to *isp1-vpn* VPN.
Step 11	Router(config-subif)# **ip address** *ipaddress mask*	Assigns the subinterface an IP address and a subnet mask.
Step 12	Router(config-subif)# **cable helper-address** *ip-address* **cable-modem**	Forwards DHCP requests from cable modems to the IP address listed.
Step 13	Router(config-subif)# **cable helper-address** *ip-address* **host**	Forwards DHCP requests from hosts to the IP address listed.
Step 14	Router(config-if)# **interface bundle n.***x*	Defines an additional subinterface for the ISP (such as isp2).
Step 15	Router(config-subif)# **description** *string*	Identifies the subinterface (such as subinterface for *isp2-vpn)* .
Step 16	Router(config-subif)# **vrf forwarding** *isp2-vpn*	Assigns the subinterface to *isp2-vpn* VPN.
Step 17	Router(config-subif)# **ip address** *ipaddress mask*	Assigns the subinterface an IP address and a subnet mask.

	Command or Action	Purpose
Step 18	Router(config-subif)# **cable helper-address** *ip-address* **cable-modem**	Forwards DHCP requests from cable modems to the IP address listed.
Step 19	Router(config-subif)# **cable helper-address** *ip-address* **host**	Forwards DHCP requests from hosts to the IP address listed.
Step 20	Router(config)# **exit**	Returns to configuration mode.

Configuring Cable Interface Bundles

To assign a cable interface to a bundle, perform the following steps beginning in the interface configuration mode.

Procedure

	Command or Action	Purpose
Step 1	Router(config)# **interface cable** *slot/port*	Enters the cable interface configuration mode. IP addresses are not assigned to this interface. They are assigned to the logical subinterfaces created within this interface.
Step 2	Router(config-if)# **cable bundle***bundle-number*	Defines the interface as the bundle interface.
Step 3	Router(config)# **interface cable** *slot/subslot/port*	Enters the cable interface configuration mode for another cable interface. IP addresses are not assigned to this interface. They are assigned to the logical subinterfaces created within this interface.
Step 4	Router(config-if)# **cable bundle** *bundle-number*	Adds the interface to the bundle specified by *bundle-number* .

Configuring Subinterfaces and MPLS VPNs on a Virtual Bundle Interface

To configure subinterfaces on a virtual bundle interface and assign each subinterface a Layer 3 configuration:

Configure cable interface bundles.

Define subinterfaces on the virtual bundle interface and assign a Layer 3 configuration to each subinterface.

Create one subinterface for each customer VPN (one per ISP).

Configuring MPLS in the P Routers in the Provider Core

To configure MPLS in the P routers in the provider core, perform the following steps.

Procedure

	Command or Action	Purpose
Step 1	Router# **configure terminal**	Enters configuration mode.
Step 2	Router(config)#**ip cef**	Enables Cisco Express Forwarding (CEF) operation. For information about CEF configuration and command syntax, see Cisco Express Forwarding Overview and Configuring Cisco Express Forwarding.
Step 3	Router(config)#**interface Tengigabitethernet** *slot/subslot/port*	Enters GigabitEthernet interface configuration mode.
Step 4	Router(config-if)#**ip address** *ip-address mask*	Defines the primary IP address range for the interface.
Step 5	Router(config-if)#**mpls ip**	Enables the interface to be forwarded to an MPLS packet.
Step 6	Router(config-if)#**exit**	Returns to global configuration mode.
Step 7	Router(config)#**mpls label-protocol ldp**	Enables Label Distribution Protocol (LDP). For information about LDP and MPLS, see Configuring Multiprotocol Label Switching.
Step 8	Router(config)# **exit**	Returns to the configuration mode.

Verifying the MPLS VPN Configuration

Use the following commands to verify MPLS VPN operations on PE routers. For more MPLS VPN verification commands, see Configuring Multiprotocol Label Switching.

Procedure

	Command or Action	Purpose
Step 1	Router# **show ip vrf**	Displays the set of VRFs and interfaces.
Step 2	Router# **show ip route vrf [vrf-name]**	Displays the IP routing table for a VRF.
Step 3	Router# **show ip protocols vrf [vrf-name]**	Displays the routing protocol information for a VRF.
Step 4	Router# **show ip route vrf** *vrf-name*	Displays the Local and Remote CE devices that are in the PE routing table.
Step 5	Router# **show mpls forwarding-table**	Displays entries for a VPN Routing/Forwarding instance.

What to do next

For more verification instructions, see the MPLS: Layer 3 VPNs Configuration Guide.

Configuration Examples

This section provides the following configuration examples:

VRF Definition Configuration

```
vrf definition Basketball
 rd 100:2
 route-target export 100:2
 route-target import 100:0
 route-target import 100:2
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
vrf definition Football
 rd 100:1
 route-target export 100:1
 route-target import 100:0
 route-target import 100:1
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
vrf definition MGMT
 rd 100:0
 route-target export 100:0
 route-target import 100:0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
vrf definition Mgmt-intf
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
vrf definition Tennis
 rd 100:4
 route-target export 100:4
 route-target import 100:0
 route-target import 100:4
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
vrf definition Volleyball
 rd 100:3
 route-target export 100:3
```

```
 route-target import 100:0
 route-target import 100:3
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
```

Cable Bundle SubInterface Configuration

```
interface Bundle255
 description Bundle Master Interface
 no ip address
 cable arp filter request-send 3 2
 cable arp filter reply-accept 3 2

interface Bundle255.1
 description Management Interface
 vrf forwarding MGMT
 ip address 112.51.0.1 255.255.0.0
 cable helper-address 20.11.0.162
 ipv6 address 2001:100:112:B001::1/64

interface Bundle255.2
 vrf forwarding Basketball
 ip address 112.54.0.1 255.255.0.0 secondary
 ip address 112.53.0.1 255.255.0.0
 cable helper-address 20.11.0.62
 cable helper-address 20.11.0.162
 ipv6 address 2001:100:112:B003::1/64
 ipv6 address 2001:100:112:B004::1/64

interface Bundle255.3
 vrf forwarding Football
 ip address 112.56.0.1 255.255.0.0 secondary
 ip address 112.55.0.1 255.255.0.0
 cable helper-address 20.11.0.62
 cable helper-address 20.11.0.162
 ipv6 address 2001:100:112:B005::1/64
 ipv6 address 2001:100:112:B006::1/64

interface Bundle255.4
 vrf forwarding Volleyball
 ip address 112.58.0.1 255.255.0.0 secondary
 ip address 112.57.0.1 255.255.0.0
 cable helper-address 20.11.0.62
 cable helper-address 20.11.0.162
 ipv6 address 2001:100:112:B007::1/64
 ipv6 address 2001:100:112:B008::1/64

interface Bundle255.5
 vrf forwarding Tennis
 ip address 112.61.0.1 255.255.0.0 secondary
 ip address 112.60.0.1 255.255.0.0 secondary
 ip address 112.59.0.1 255.255.0.0
 cable helper-address 20.11.0.162
 ipv6 address 2001:100:112:B009::1/64
 ipv6 address 2001:100:112:B00A::1/64
```

PE WAN Interface Configuration

```
mpls label protocol ldp
mpls ldp nsr
mpls ldp graceful-restart

interface TenGigabitEthernet4/1/1
 description WAN connection to cBR8
 mtu 4470
 ip address 100.6.120.5 255.255.255.252
 ip router isis hub
 ipv6 address 2001:100:6:120::5:1/112
 ipv6 enable
 mpls ip
 mpls traffic-eng tunnels
 cdp enable
 isis circuit-type level-1
 isis network point-to-point
 isis csnp-interval 10
 hold-queue 400 in
 ip rsvp bandwidth 1000000
end
```

PE BGP Configuration

```
router bgp 100
 bgp router-id 100.120.120.120
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 timers bgp 5 60
 neighbor 100.100.4.4 remote-as 100
 neighbor 100.100.4.4 ha-mode sso
 neighbor 100.100.4.4 update-source Loopback0
 neighbor 100.100.4.4 ha-mode graceful-restart
 !
 address-family ipv4
  redistribute connected
  redistribute static route-map static-route
  redistribute rip
  neighbor 100.100.4.4 activate
  neighbor 100.100.4.4 send-community extended
  neighbor 100.100.4.4 next-hop-self
  neighbor 100.100.4.4 soft-reconfiguration inbound
  maximum-paths ibgp 2
 exit-address-family
 !
 address-family vpnv4
  neighbor 100.100.4.4 activate
  neighbor 100.100.4.4 send-community extended
 exit-address-family
 !
 address-family ipv6
  redistribute connected
  redistribute rip CST include-connected
  redistribute static metric 100 route-map static-route-v6
  neighbor 100.100.4.4 activate
  neighbor 100.100.4.4 send-community extended
  neighbor 100.100.4.4 send-label
```

```
exit-address-family
!
address-family vpnv6
 neighbor 100.100.4.4 activate
 neighbor 100.100.4.4 send-community extended
exit-address-family
!
address-family ipv4 vrf Basketball
 redistribute connected
exit-address-family
!
address-family ipv6 vrf Basketball
 redistribute connected
 redistribute static metric 100
exit-address-family
!
address-family ipv4 vrf Football
 redistribute connected
exit-address-family
!
address-family ipv6 vrf Football
 redistribute connected
 redistribute static metric 100
exit-address-family
!
address-family ipv4 vrf MGMT
 redistribute connected
exit-address-family
!
address-family ipv6 vrf MGMT
 redistribute connected
exit-address-family
!
address-family ipv4 vrf Tennis
 redistribute connected
 redistribute static route-map static-route
 redistribute rip
exit-address-family
!
address-family ipv6 vrf Tennis
 redistribute connected
 redistribute rip CST include-connected
 redistribute static metric 100 route-map static-route-v6
exit-address-family
!
address-family ipv4 vrf Volleyball
 redistribute connected
 redistribute static route-map static-route
 redistribute rip
exit-address-family
!
address-family ipv6 vrf Volleyball
 redistribute connected
 redistribute rip CST include-connected
 redistribute static metric 100 route-map static-route-v6
exit-address-family
```

Additional References

Standards

Standard	Title
DOCSIS 1.0	*DOCSIS 1.0*

MIBs

MIB	MIBs Link
CISCO-DOCS-REMOTE-QUERY.my	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://tools.cisco.com/ITDIT/MIBS/servlet/index

RFCs

RFC	Title
RFC 1163	A Border Gateway Protocol
RFC 1164	Application of the Border Gateway Protocol in the Internet
RFC 2283	Multiprotocol Extensions for BGP-4
RFC 2547	BGP/MPLS VPNs
RFC 2233	DOCSIS OSSI Objects Support
RFC 2669	Cable Device MIB

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

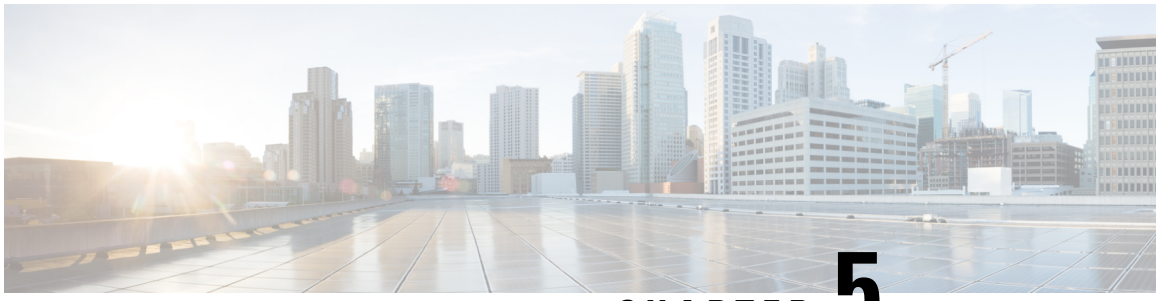
Feature Information for MPLS VPN Cable Enhancements

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 10: Feature Information for MPLS VPN Cable Enhancements

Feature Name	Releases	Feature Information
Multiprotocol Label Switching Virtual Private Network (MPLS VPN)	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on the Cisco cBR Series Converged Broadband Router.



CHAPTER 5

Multicast VPN and DOCSIS 3.0 Multicast QoS Support

The CMTS enhanced multicast new features are consistent with DOCSIS 3.0 specifications and include:

- Enhanced multicast echo in which the Layer 3 multicast switching path uses a Cisco Packet Processor (CPP) parallel express forwarding multicast routing table.
- Enhanced multicast quality of service (MQoS) framework that specifies a group configuration (GC) to define a session range of multicast addresses and rule priorities and its associated multicast VPN (MVPN).
- Intelligent multicast admission control to include multicast service flows.
- Enhanced multicast VPN feature to configure and support multicast traffic in a multiprotocol label switching (MPLS)-VPN environment.

Finding Feature Information

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 11: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Restrictions for the Multicast VPN and DOCSIS 3.0 Multicast QoS Support

The type of service (ToS) parameter is not recognized by the Cisco cBR series routers.

To avail 40000 multicast sessions, a minimum of one bundle should be present for each LC.

Information About the Multicast VPN and DOCSIS 3.0 Multicast QoS Support

IP multicast—transmission of the same information to multiple cable network recipients—improves bandwidth efficiency and allows service providers to offer differentiated quality of service for different types of traffic. Enhanced multicast introduces multicast improvements as mandated by the introduction of DOCSIS 3.0 specifications.

Note DOCSIS 3.0 standards retain backwards compatibility with the DOCSIS 2.0 multicast mode of operation.

The Cisco cBR routers support 40000 DSG multicast sessions per chassis.

The following are the benefits of CMTS enhanced multicast are:

Enhanced Quality of Service

In the new multicast QoS (MQoS) framework, you can specify a group configuration (GC) that defines a session range of multicast addresses and rule priorities and its associated multicast VPN (MVPN). For every GC, there is attached a group QoS configuration (GQC) and a group encryption rule.

Based on the session range, rule priority, and MVPN, a multicast service flow is admitted into a GC and the associated GQC and group encryption rule are applied to the flow. In MQoS implementation, the source address of the multicast session is not checked because the current implementation for cable-specific multicast supports IGMP Version 2 but not IGMP Version 3. The downstream service flow, service identifier (SID), and MAC-rewrite string are created at the time of a new IGMP join (or static multicast group CLI on the interface) and MQoS is applied to the new multicast group join.

The benefits of enhanced QoS are the following:

- Group classifiers can be applied at cable interface level and also at bundle interface level.
- Group service flow (GSF) definition is based on service class names. The GSF is similar to individual service flows and commonly includes the minimum rate and maximum rate parameters for the service class. GSF is shared by all cable modems on a particular downstream channel set (DCS) that is matched to the same group classifier rule (GCR). A default service flow is used for multicast flows that do not match to any GCR. A GSF is always in the active state.
- CMTS replicates multicast packets and then classifies them.
- Single-stage replication and two-stage replication are supported.
- Enhanced QoS is compatible and integrated with DOCSIS Set-Top Gateway (DSG).

Intelligent Multicast Admission Control

Admission control allows you to categorize service flows into buckets. Examples of categories are the service class name used to create the service flow, service flow priority, or the service flow type such as unsolicited grant service (UGS). Bandwidth limits for each bucket can also be defined. For example, you can define bucket 1 for high priority packet cable service flows and specify that bucket 1 is allowed a minimum of 30 percent and a maximum of 50 percent of the link bandwidth.

Intelligent multicast admission control includes additional features such as the inclusion of multicast service flows using the GSF concept. GSFs are created based on the rules as defined in the GQC table. The rules link the multicast streams to a GSF through the session range. The service class name in the rule defines the QoS for that GSF. Additionally, another attribute is added to the rules and the group configuration table to specify the application type to which each GSF belongs. In this way, the QoS associated with each GSF is independent of the bucket category for the GSF.

The benefits of intelligent multicast admission control are the following:

- There is explicit acknowledgment of the establishment of each multicast session.
- Admission control does not consume additional bandwidth for multicast flows once the first flow is established.
- Service flows are cleaned up as the multicast session is torn down.

Multicast Session Limit Support

In a multicast video environment, you can limit the number of multicast sessions admitted onto a particular service flow. The multicast session limit feature—which adds functionality on top of the multicast QoS infrastructure—enables you to specify the number of multicast sessions to be admitted on a particular service flow. If the current number of sessions has reached the defined limit, new sessions will be forwarded but they will make use of the default multicast service flow until a session ends to free up a slot for new sessions.

Multicast Virtual Private Network

The new multicast VPN (MVPN) feature allows you to configure and support multicast traffic in a multiprotocol label switching (MPLS)-VPN environment. This feature supports routing and forwarding of multicast packets for each individual VPN virtual routing and forwarding (VRF) instance, and also provides a mechanism to transport VPN multicast packets across the service provider backbone.

MVPN allows you to connect multiple remote sites or devices over either a Layer 3 or Layer 2 VPN. A Layer 3 VPN enables the routing of traffic inside the VPN. A Layer 2 VPN provides a bridging transport mechanism for traffic between remote sites belonging to a customer. To support multicast over Layer 3 VPNs, each VPN receives a separate multicast domain with an associated MVPN routing and forwarding (mVRF) table maintained by the provider edge (PE) router. In a cable environment, the PE router is a routing CMTS. The provider network builds a default multicast distribution tree (default-MDT) for each VPN between all the associated mVRF-enabled PE routers. This tree is used to distribute multicast traffic to all PE routers.

To enable maximum security and data privacy in a VPN environment, the CMTS distinguishes between multicast sessions on the same downstream interface that belong to different VPNs. To differentiate multicast traffic between different VPNs, the CMTS implements a per-VRF subinterface multicast security association identifier (MSAID) allocation feature that is BPI+ enabled. The MSAID is allocated for each cable bundle group for each subinterface. A multicast group has a specific MSAID for each VRF instance.

How to Configure the Multicast VPN and DOCSIS 3.0 Multicast QoS Support

This section contains the following procedures:

Configuring a QoS Profile for a Multicast Group

To configure a QoS profile that can be applied to a QoS group configuration, use the **cable multicast group-qos** command. You must configure a QoS profile before you can add a QoS profile to a QoS multicast group.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**cable multicast group-qos** *number* **scn** *service-class-name* **control** { **single** | **aggregate** [**limit** *max-sessions*]} **Example:** `Router(config)#: cable multicast group-qos 2 scn name1 control single`	Configures a QoS profile that can be applied to a multicast QoS group. **Note** If a number is not specified, a default QoS profile is applied. The default group qos configuration creates a default multicast service flow for each cable interface that is used when a multicast session does not match any classifiers of a GC on the interface.

Configuring a Multicast QoS Group

You can specify a group configuration (GC) that defines a session range of multicast addresses and rule priorities and its associated multicast VPN (MVPN). For every GC, there is attached a group QoS configuration and a group encryption rule.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**configureterminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**cable multicast group-qos** *number* **scn** *service-class-name* **control** {**single** | **aggregate** [limit *max-sessions*]} **Example:** `Router(config-mqos)# cable multicast group-qos 5 scn name1 control single`	(Optional) Configures a QoS profile that can be applied to a multicast QoS group. **Note** If a number is not specified, a default QoS profile is applied. The default group qos configuration creates a default multicast service flow for each cable interface that is used when a multicast session does not match any classifiers of a GC on the interface.
Step 4	**cable multicast qos group** *id* **priority** *value* [**global**] **Example:** `Router(config)# cable multicast qos group 2 priority 6`	Configures a multicast QoS group and enters multicast QoS configuration mode.
Step 5	**session-range** *ip-address ip-mask* **Example:** `Router(config-mqos)# session-range 224.10.10.10 255.255.255.224`	Specifies the session range IP address and IP mask of the multicast QoS group. You can configure multiple session ranges.
Step 6	**tos** *low-byte high-byte mask* **Example:** `Router(config-mqos)# tos 1 6 15`	(Optional) Specifies the minimum type of service (ToS) data bytes, maximum ToS data bytes, and mask for a multicast QoS group.
Step 7	**vrf***name* **Example:** `Router(config-mqos)# vrf name1`	(Optional) Specifies the name for the virtual routing and forwarding (VRF) instance. **Note** If a multicast QoS (MQoS) group is not defined for this VRF, you will see an error message. You must either define a specific MQoS group for each VRF, or define a default MQoS group that can be assigned in those situations where no matching MQoS group is found. See the Configuring a Default Multicast QoS Group for VRF, on page 79.
Step 8	**application-id***number* **Example:** `Router(config-mqos)# application-id 25`	(Optional) Specifies the application identification number of the multicast QoS group. This value is configured to enable admission control to the multicast QoS group.

Configuring a Default Multicast QoS Group for VRF

Each virtual routing and forwarding (VRF) instance that is defined must match a defined MQoS group to avoid multicast stream crosstalk between VRFs. To avoid potential crosstalk, define a default MQoS group that is assigned to the VRF whenever the multicast traffic in the VRF does not match an existing MQoS group.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**cable multicastgroup-qos***number* **scn***service-class-name* **control** {**single** | **aggregate** [**limit** *max-sessions*]} **Example:** `Router(config-mqos)# cable multicast group-qos 5 scn name1 control single`	(Optional) Configures a QoS profile that can be applied to a multicast QoS group.
Step 4	**cable multicast qos group** *id* **priority 255 global** **Example:** `Router(config)# cable multicast qos group 2 priority 255 global`	Configures a default multicast QoS group and enters multicast QoS configuration mode.
Step 5	**session-range 224.0.0.0 224.0.0.0** **Example:** `Router(config-mqos)# session-range 224.0.0.0 224.0.0.0`	Specifies the session-range IP address and IP mask of the default multicast QoS group. By entering 224.0.0.0 for the IP address and the IP mask you cover all possible multicast sessions.
Step 6	**vrf***name* **Example:** `Router(config-mqos)# vrf name1`	Specifies the name of the virtual routing and forwarding (VRF) instance.
Step 7	**application-id***number* **Example:** `Router(config-mqos)# application-id 5`	(Optional) Specifies the application identification number of the multicast QoS group. This value is configured to enable admission control to the multicast QoS group.

Verifying Configuration of the Multicast VPN and DOCSIS 3.0 Multicast QoS Support

To verify the configuration of the Multicast VPN and DOCSIS 3.0 Multicast QoS Support feature, use the **show** commands described below.

- To show the configuration parameters for multicast sessions on a specific bundle, use the **show interface bundle** *number* **multicast-sessions** command as shown in the following example:

```
Router# show interface bundle 1 multicast-sessions
Multicast Sessions on Bundle1
 Group            Interface    GC  SAID SFID  GQC GEn RefCount GC-Interface State
 234.1.1.45       Bundle1.1    1   8193  ---   1   5   1        Bundle1     ACTIVE
 234.1.1.46       Bundle1.1    1   8193  ---   1   5   1        Bundle1     ACTIVE
 234.1.1.47       Bundle1.1    1   8193  ---   1   5   1        Bundle1     ACTIVE
Aggregate Multicast Sessions on Bundle1
 Aggregate Sessions for SAID 8193 GQC 1 CurrSess 3
 Group            Interface    GC  SAID SFID AggGQC GEn RefCount GC-Interface
 234.1.1.45       Bundle1.1    1   8193  ---   1      5   1        Bundle1
 234.1.1.46       Bundle1.1    1   8193  ---   1      5   1        Bundle1
 234.1.1.47       Bundle1.1    1   8193  ---   1      5   1        Bundle1
```

- To show the configuration parameters for multicast sessions on a specific cable, use the **show interface cable** *ip-addr* **multicast-sessions** command as shown in the following example:

```
Router# show interface cable 7/0/0 multicast-sessions
Default Multicast Service Flow 3 on Cable7/0/0
Multicast Sessions on Cable7/0/0
 Group            Interface    GC  SAID SFID  GQC GEn RefCount GC-Interface State
 234.1.1.45       Bundle1.1    1   8193 24     1   5   1        Bundle1     ACTIVE
 234.1.1.46       Bundle1.1    1   8193 24     1   5   1        Bundle1     ACTIVE
 234.1.1.47       Bundle1.1    1   8193 24     1   5   1        Bundle1     ACTIVE
Aggregate Multicast Sessions on Cable7/0/0
 Aggregate Sessions for SAID 8193 SFID 24 GQC 1 CurrSess 3
 Group            Interface    GC  SAID SFID AggGQC GEn RefCount GC-Interface
 234.1.1.45       Bundle1.1    1   8193 24     1      5   1        Bundle1
 234.1.1.46       Bundle1.1    1   8193 24     1      5   1        Bundle1
 234.1.1.47       Bundle1.1    1   8193 24     1      5   1        Bundle1
```

- To show the MSAID multicast group subinterface mapping, use the **show interface cable** *address* **modem** command as shown in the following example:

```
Router# show interface cable 6/1/0 modem
SID   Priv Type      State          IP address      method  MAC address     Dual
      bits                                                                  IP
9     11   modem     online(pt)     101.1.0.6       dhcp    0006.28f9.8c79  N
9     11   host      unknown        111.1.1.45      dhcp    0018.1952.a859  N
10    10   modem     online(pt)     101.1.0.5       dhcp    0006.5305.ac19  N
10    10   host      unknown        111.1.0.3       dhcp    0018.1952.a85a  N
13    10   modem     online(pt)     101.1.0.3       dhcp    0014.f8c1.fd1c  N
8195  10   multicast unknown        224.1.1.51      static  0000.0000.0000  N
8195  10   multicast unknown        224.1.1.49      static  0000.0000.0000  N
8195  10   multicast unknown        224.1.1.50      static  0000.0000.0000  N
```

Configuration Examples for the Multicast VPN and DOCSIS 3.0 Multicast QoS Support

This section provides the following configuration examples:

Example: Configuring Group QoS and Group Encryption Profiles

Note To add group QoS and group encryption profiles to a QoS group, you must configure each profile first before configuring the QoS group.

In the following example, QoS profile 3 and encryption profile 35 are configured.

```
configure terminal
cable multicast group-qos 3 scn name1 control single
cable multicast group-encryption 35 algorithm 56bit-des
```

Example: Configuring a QoS Group

In the following example, QoS group 2 is configured with a priority of 6 and global application. To QoS group 2, QoS profile 3 and encryption profile 35 are applied. Other parameters are configured for QoS group 2 including application type, session range, ToS, and VRF.

```
cable multicast qos group 2 priority 6 global
group-encryption 35
group-qos 3
session-range 224.10.10.01 255.255.255.254
tos 1 6 15
vrf vrf-name1
application-id 44
```

Additional References

The following sections provide references related to the Multicast VPN and DOCSIS 3.0 Multicast QoS Support.

Related Documents

Related Topic	Document Title
CMTS cable commands	*Cisco CMTS Cable Command Reference* http://www.cisco.com/c/en/us/td/docs/cable/cmts/cmd_ref/b_cmts_cable_cmd_ref.html

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 2236	*Internet Group Management Protocol, Version 2*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

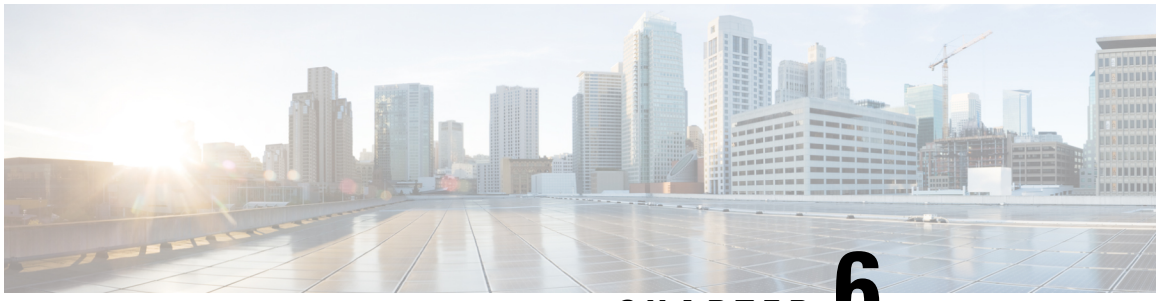
Feature Information for Multicast VPN and DOCSIS3.0 Multicast QoS Support

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 12: Feature Information for Multicast VPN and DOCSIS3.0 Multicast QoS Support

Feature Name	Releases	Feature Information
Multicast VPN and DOCSIS3.0 multicast QoS support	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.5.1 on theCisco cBR Series Converged Broadband Routers.



CHAPTER 6

EtherChannel for the Cisco CMTS

This document describes the features, benefits and configuration of Cisco EtherChannel technology on the Cisco Cable Modem Termination System (CMTS).

EtherChannel is a technology by which to configure and aggregate multiple physical Ethernet connections to form a single logical port with higher bandwidth. The first EtherChannel port configured on the Cisco CMTS serves as the EtherChannel bundle primary by default, and each subsidiary interface interacts with the network using the MAC address of the EtherChannel bundle primary.

EtherChannel ports reside on a routing or bridging end-point. The router or switch uses EtherChannel to increase bandwidth utilization in either half- or full-duplex mode, and load balances the traffic across the multiple physical connections.

EtherChannel on the Cisco CMTS supports inter-VLAN routing with multiple devices and standards, and supports Ten Gigabit EtherChannel (GEC) on the Cisco cBR series routers.

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Contents

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 13: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Restrictions for EtherChannel on the Cisco CMTS

- EtherChannel on the Cisco CMTS is limited to Network Layer 3 functions, and does not support Data-Link Layer 2 EtherChannel functions as with certain other Cisco product platforms.
- The Port Aggregation Protocol (PAgP) is not supported on the Cisco CMTS as with other Cisco product platforms (such as the CatOS switch).
- Only the IEEE 802.1Q trunking protocol is supported on the Cisco CMTS. ATM trunking is not supported on the Cisco cBR series routers.
- The maximum supported links per bundle is 8.
- EtherChannel on Cisco CMTS supports only physical ports or interfaces that have the same speed.
- EtherChannel on the Cisco cBR series routers does not support MQC QOS. You can use Equal Cost Multi Path (ECMP) load balancing instead of EtherChannel.
- Layer 3 configurations on member interfaces of EtherChannel are not supported.
- MAC Address Accounting feature on port channel is not supported.

Information About EtherChannel on the Cisco CMTS

This section contains the following:

Introduction to EtherChannel on the Cisco CMTS

EtherChannel is based on proven industry-standard technology. The Cisco CMTS supports EtherChannel with several benefits, including the following:

- EtherChannel on the Cisco CMTS supports subsecond convergence times.
- EtherChannel can be used to connect two switch devices together, or to connect a router with a switch.
- A single EtherChannel connection supports a higher bandwidth between the two devices.
- The logical port channels on either Cisco CMTS platform provide fault-tolerant, high-speed links between routers, switches, and servers.
- EtherChannel offers redundancy and high availability on the Cisco CMTS. Failure of one connection causes a switch or router to use load balancing across the other connections in the EtherChannel.
- Load balancing on the Cisco CMTS supports dynamic link addition and removal without traffic interruption.
- EtherChannel supports inter-VLAN trunking. Trunking carries traffic from several VLANs over a point-to-point link between the two devices. The network provides inter-VLAN communication with trunking between the Cisco CMTS router and one or more switches. In a campus network, trunking is configured over an EtherChannel link to carry the multiple VLAN information over a high-bandwidth channel.

Cisco Ten Gigabit EtherChannel on the Cisco cBR Series Routers

Cisco Ten Gigabit EtherChannel (GEC) is high-performance Ethernet technology that provides gigabit-per-second transmission rates. It provides flexible, scalable bandwidth with resiliency and load sharing across links for switches, router interfaces, and servers.

Ten GEC on the Cisco cBR series routers with the following EtherChannel capabilities:

- Supports IEEE 802.1Q encapsulation for inter-VLAN networking.
- Supports a maximum of eight physical Ten Gigabit Ethernet ports to be combined as one logical EtherChannel link.
- Supports bandwidth up to 40 Gbps (half duplex) for a combined total of up to 80 Gbps (full duplex).

How to Configure EtherChannel on the Cisco CMTS

This section contains the following:

Configuring Ten Gigabit EtherChannel on the Cisco CMTS

Before you begin

- Ten Gigabit Ethernet cabling is completed and the ports are operational on the router and network.
- LAN interfaces are configured and operational on the router and network, with IP addresses and subnet masks.

Note The Cisco cBR series routers support up to eight physical connectors to be configured as one logical Ten GEC port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *n*
4. **exit**
5. **interface tengigabitethernet** *slot/subslot/port*
6. **shutdown**
7. Use one of the following commands:
   - For static Ten GEC configuration, use the **channel-group** *number* command.
   - For dynamic Ten GEC configuration, use the **channel-group** *number* **mode** {**active** | **passive**} command.
8. **no shutdown**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface port-channel** *n* **Example:** `Router(config)# interface port-channel 1`	Creates an EtherChannel interface. The first EtherChannel interface configured becomes the bundle primary for all ports in the EtherChannel group. The MAC address of the first EtherChannel interface is the MAC address for all EtherChannel interfaces in the group. To remove an EtherChannel interface from the EtherChannel group, use the **no** form of this command. If the first EtherChannel interface in the group is later removed, the second EtherChannel interface in the group becomes the bundle primary by default. Repeat this step on every EtherChannel port to be bundled into a Ten GEC group. This configuration must be present on all EtherChannel interfaces before the EtherChannel group can be configured.
Step 4	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 5	**interface tengigabitethernet** *slot/subslot/port* **Example:** `Router# interface gigabitethernet 4/1/0`	Selects the Ten Gigabit Ethernet interface that you wish to add as a member EtherChannel link in the EtherChannel bundle, and enters interface configuration mode. **Note** We recommend that the link being added to the Cisco CMTS EtherChannel be shut down prior to configuring it as a member of the EtherChannel. Use the **shutdown** command in interface configuration mode immediately before completing the following steps in this procedure.
Step 6	**shutdown** **Example:** `Router(config-if)# shutdown`	Shuts down the interface selected in step 5 before configuring it as a member of the EtherChannel.
Step 7	Use one of the following commands: • For static Ten GEC configuration, use the **channel-group** *number* command. • For dynamic Ten GEC configuration, use the **channel-group** *number* **mode** {**active** | **passive**} command.	Adds the Ten Gigabit Ethernet interface to the EtherChannel Group, associating that interface with an EtherChannel link. To remove an EtherChannel group and the associated ports from the Cisco CMTS, use the **no** form of this command.

	Command or Action	Purpose

	Example:
	`Router(config-if)# channel-group 1`
	or
	`Router(config-if)# channel-group 1 mode active`	

Step 8	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface on which EtherChannel is configured.
Step 9	**end** **Example:** `Router(config# end`	Returns to privileged EXEC mode. IP traffic should be visible on the network with completion of the above steps.

Troubleshooting Tips

Once interface operations are confirmed (prior to this procedure), and EtherChannel configurations have been verified (next procedure), any difficulty experienced through the EtherChannel links may pertain to inter-VLAN or IP routing on the network, or perhaps very high bandwidth consumption.

What to Do Next

Additional IP, access list, inter-VLAN or load balancing configurations may be made to the Cisco CMTS and these changes will be supported in the running EtherChannel configuration without service disruption from EtherChannel.

Verifying EtherChannel on the Cisco CMTS

Links can be added or removed from an EtherChannel interface without traffic interruption. If an Ethernet link in an EtherChannel interface fails, traffic previously carried over the failed link switches to the remaining links within the EtherChannel. There are a number of events that can cause a link to be added or removed including adding or removing a link using commands and simulating link failure and recovery (as with (no)shutdown links).

Cisco EtherChannel supports online insertion and removal (OIR) of field-replaceable units (FRUs) in the Cisco CMTS chassis. Ports that remain active during OIR of one FRU will take over and support the traffic bandwidth requirements without service disruption. However, OIR is not described in this procedure.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**show interface port-channel** *n* **Example:** `Router# show interface port-channel 1`	Verifies the EtherChannel configuration on the Cisco CMTS for the selected EtherChannel group.

Configuration Examples for EtherChannel on the Cisco CMTS

The following example illustrates Ten Gigabit EtherChannel information for the port-channel interface of 2.

This configuration is comprised of three Ten GEC port channels as follows:

- Member 0 is the Ten GEC interface bundle primary.
- Member 2 is the final subsidiary interface in this Ten GEC group.
- These three port-channel interfaces (members) comprise one Ten GEC group that is set up with a Ten GEC peer on the network.

```
Router# show interface port-channel 2
Port-channel2 is up, line protocol is up
  Hardware is GEChannel, address is 8888.8888.8888 (bia 0000.0000.0000)
  Internet address is 101.101.101.1/16
  MTU 1500 bytes, BW 3000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
    No. of members in this channel: 3
    No. of configured  members in this channel: 3
    No. of passive members in this channel: 0
    No. of active members in this channel: 3
        Member 0 : TenGigabitEthernet4/1/0 , Full-duplex, 1000Mb/s
        Member 1 : TenGigabitEthernet4/1/1 , Full-duplex, 1000Mb/s
        Member 2 : TenGigabitEthernet4/1/2 , Full-duplex, 1000Mb/s
    No. of Non-active members in this channel: 0
  Last input 00:00:02, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/225/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/120 (size/max)
  30 second input rate 17292000 bits/sec, 9948 packets/sec
  30 second output rate 17315000 bits/sec, 9935 packets/sec
     866398790 packets input, 3324942446 bytes, 0 no buffer
     Received 2 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast, 0 pause input
     0 input packets with dribble condition detected
     866394055 packets output, 3323914794 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
     0 output buffer failures, 0 output buffers swapped out
```

Additional References

Related Documents

Related Topic	Document Title
EtherChannel for Cisco Products	• Cisco EtherChannel home page http://www.cisco.com/warp/public/cc/techno/lnty/etty/fsetch/index.shtml • Cisco EtherChannel Technology white paper http://www.cisco.com/en/US/tech/tk389/tk213/technologies_white_paper09186a0080092944.shtml
Configuring Additional Devices for EtherChannel	• *Configuring EtherChannel and 802.1Q Trunking Between a Catalyst 2950 and a Router (inter-VLAN Routing)* http://www.cisco.com/c/en/us/support/docs/switches/catalyst-2950-series-switches/24042-158.html • *Configuring EtherChannel and 802.1Q Trunking Between Catalyst 2900XL/3500XL and Catalyst 2940, 2950/2955, and 2970 Switches* http://www.cisco.com/c/en/us/support/docs/switches/catalyst-2900-xl-series-switches/21041-131.html

Standards and RFCs

Standards	Title
IEEE Std 802.1Q, 2003 Edition	IEEE Std 802.1Q, 2003 Edition (Incorporates IEEE Std 802.1Q-1998, IEEE Std 802.1u-2001, IEEE Std 802.1v-2001, and IEEE Std 802.1s-2002) http://ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=27089

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support

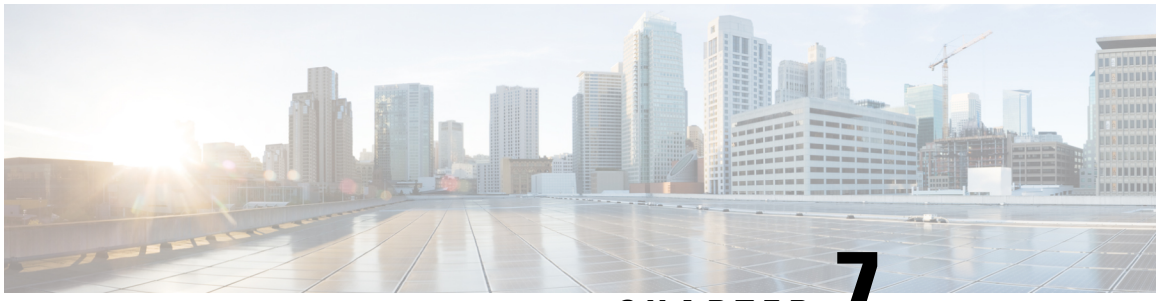
Feature Information for EtherChannel on Cisco CMTS

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 14: Feature Information for EtherChannel on Cisco CMTS

Feature Name	Releases	Feature Information
EtherChannel on Cisco CMTS	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on the Cisco cBR Series Converged Broadband Router.



CHAPTER 7

Flow-Based per Port-Channel Load Balancing

The Flow-Based per Port-Channel Load Balancing feature allows different flows of traffic over a Ten Gigabit EtherChannel (GEC) interface to be identified based on the packet header and then mapped to the different member links of the port channel. This feature enables you to apply flow-based load balancing and VLAN-manual load balancing to specific port channels.

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Contents

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 15: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Restrictions for Flow-Based per Port-Channel Load Balancing

- Supports up to 64 Ten GEC interfaces.
- Supports up to 8 member links per Ten GEC interface.
- dot1q L2VPN is not supported over a port-channel with load-balancing vlan configured.

Information About Flow-Based per Port-Channel Load Balancing

Flow-Based Load Balancing

Flow-based load balancing identifies different flows of traffic based on the key fields in the data packet. For example, IPv4 source and destination IP addressees can be used to identify a flow. The various data traffic flows are then mapped to the different member links of a port channel. After the mapping is done, the data traffic for a flow is transmitted through the assigned member link. The flow mapping is dynamic and changes when there is any change in the state of a member link to which a flow is assigned. The flow mappings can also change if member links are added to or removed from the GEC interface. Multiple flows can be mapped to each member link.

Buckets for Flow-Based Load Balancing

Load balancing dynamically maps traffic flows to the member links of a Ten GEC interface through the concept of buckets. The various defined traffic flows are mapped to the buckets and the buckets are evenly distributed among the member links. Each port channel maintains 16 buckets, with one active member link associated with each bucket. All traffic flows mapped to a bucket use the member link to which the bucket is assigned.

The router creates the buckets-to-member links mappings when you apply flow-based load balancing to a port channel and the port channel has at least one active member link. The mappings are also created when the first member link is added, or comes up, and the load-balancing method is set to flow-based.

When a member link goes down or is removed from a port channel, the buckets associated with that member link are redistributed among the other active member links in a round-robin fashion. When a member link comes up or is added to a port channel, some of the buckets associated with other links are assigned to this link.

If you change the load-balancing method, the bucket-to-member link mappings for flow-based load balancing are deleted. The mappings are also deleted if the port channel is deleted or the last member link in the port channel is deleted or goes down.

Load Balancing on Port Channels

GEC interfaces can use either dynamic flow-based load balancing or VLAN-manual load balancing. You can configure the load-balancing method globally for all port channels or directly on specific port channels. The global configuration applies only to those port channels for which you have not explicitly configured load balancing. The port-channel configuration overrides the global configuration.

Flow-based load balancing is enabled by default at the global level. You must explicitly configure VLAN load balancing or the load-balancing method is flow-based.

The table below lists the load-balancing method that is applied to port channels based on the configuration:

Table 16: Flow-Based Load Balancing Configuration Options

Global Configuration	Port-Channel Configuration	Load Balancing Applied
Not configured	Not configured	Flow-based
	Flow-based	Flow-based
	VLAN-manual	VLAN-manual
VLAN-manual	Not configured	VLAN-manual
	Flow-based	Flow-based
	VLAN-manual	VLAN-manual

The table below lists the configuration that results if you change the global load-balancing method.

Table 17: Results When Global Configuration Changes

Port-Channel Configuration	Global Configuration		Action Taken at Port-Channel
—	From	To	—
Not configured	Not configured	VLAN-manual	Changed from flow-based to VLAN-manual
	VLAN-manual	Not configured	Changed from VLAN-manual to flow-based
Configured	Any	Any	No change

The table below lists the configuration that results if you change the port-channel load-balancing method.

Table 18: Results When Port-Channel Configuration Changes

Port-Channel Configuration	Global Configuration		Action Taken at Port-Channel
—	From	To	—

Port-Channel Configuration	Global Configuration		Action Taken at Port-Channel
Not configured	Not configured	VLAN-manual	Changed from flow-based to VLAN-manual
	Not configured	Flow-based	No action taken
	VLAN-manual	Flow-based	Changed from VLAN-manual to flow-based
	VLAN-manual	Not configured	Changed from VLAN-manual to flow-based
	Flow-based	VLAN-manual	Changed from flow-based to VLAN-manual
	Flow-based	Not configured	No action taken
Configured	Not configured	VLAN-manual	No action taken
	Not configured	Flow-based	Changed from VLAN-manual to flow-based
	VLAN-manual	Flow-based	Changed from VLAN-manual to flow-based
	VLAN-manual	Not configured	No action taken
	Flow-based	VLAN-manual	Changed from flow-based to VLAN-manual
	Flow-based	Not configured	Changed from flow-based to VLAN-manual

How to Enable Flow-Based per Port-Channel Load Balancing

Configuring Load Balancing on a Port Channel

To configure load balancing on a port channel, perform the following steps. Repeat these steps for each GEC interface.

Before you begin

If you have already configured your desired load-balancing method globally and want to use that method for all port channels, you need not perform this task. To configure load balancing globally, use the **port-channel load-balancing vlan-manual** command. If you do not configure the global command, flow-based load balancing is applied to all port channels.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *channel-number*
4. **load-balancing** {**flow** | **vlan**}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface port-channel** *channel-number* **Example:** `Router(config)# interface port-channel 1`	Enters interface configuration mode and defines the interface as a port channel.
Step 4	**load-balancing** {**flow**	**vlan**} **Example:** `Router(config-if)# load-balancing flow`	Applies a load-balancing method to the specific port channel. • If you do not configure this command, the port channel uses the global load-balancing method configured with the **port-channel load-balancing vlan-manual** command. The global default is flow-based.
Step 5	**end** **Example:** `Router(config-if)# end`	Exits configuration mode.

Verifying Load Balancing Configuration on a Ten GEC Interface

- **show running-config interface port-channel** *channel-number* —Displays the port channel configuration.

  Following is a sample output of this command:

```
Router# show running-config interface port-channel 62
Building configuration...

Current configuration : 108 bytes
!
```

```
interface Port-channel62
 ip address 12.1.1.1 255.255.255.0
 ipv6 address 2001:12:1:1::1/64
 mpls
```

- **show etherchannel load-balancing** — Displays the load balancing method applied to each port channel.

  The following is a sample output of this command:

```
Router# show etherchannel load-balancing

EtherChannel Load-Balancing Method:
Global LB Method: flow-based

  Port-Channel:                       LB Method
    Port-channel62                  :  flow-based
    Port-channel63                  :  flow-based
```

- **show interfaces port-channel** *channel-number* **etherchannel** —Displays the bucket distribution currently in use.

  The following is a sample output for an interface with load balancing set to flow-based:

```
Router(config)# show interface port-channel 62 etherchannel

 All IDBs List contains 8 configured interfaces
  Port: TenGigabitEthernet4/1/0 (index: 0)
  Port: TenGigabitEthernet4/1/1 (index: 1)
  Port: TenGigabitEthernet4/1/2 (index: 2)
  Port: TenGigabitEthernet4/1/3 (index: 3)
  Port: TenGigabitEthernet4/1/4 (index: 4)
  Port: TenGigabitEthernet4/1/5 (index: 5)
  Port: TenGigabitEthernet4/1/6 (index: 6)
  Port: TenGigabitEthernet4/1/7 (index: 7)

 Active Member List contains 8 interfaces
  Port: TenGigabitEthernet4/1/0
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/1
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/2
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/3
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/4
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/5
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/6
    LACP Mode: Active

  Port: TenGigabitEthernet4/1/7
    LACP Mode: Active

 Passive Member List contains 0 interfaces
 Load-Balancing method applied: flow-based
```

```
Bucket Information for Flow-Based LB:
Interface:                                Buckets
   TenGigabitEthernet4/1/0:
                          Bucket 0 , Bucket 1
   TenGigabitEthernet4/1/1:
                          Bucket 2 , Bucket 3
   TenGigabitEthernet4/1/2:
                          Bucket 4 , Bucket 5
   TenGigabitEthernet4/1/3:
                          Bucket 6 , Bucket 7
   TenGigabitEthernet4/1/4:
                          Bucket 8 , Bucket 9
   TenGigabitEthernet4/1/5:
                          Bucket 10, Bucket 11
   TenGigabitEthernet4/1/6:
                          Bucket 12, Bucket 13
   TenGigabitEthernet4/1/7:
                          Bucket 14, Bucket 15
```

Configuration Examples for Flow-Based per Port-Channel Load Balancing

Example: Flow-Based Load Balancing

The following example shows a configuration where flow-based load balancing is configured on port-channel 2 while the VLAN-manual method is configured globally:

```
!
no aaa new-model
port-channel load-balancing vlan-manual
ip source-route
.
.
.
interface Port-channel2
 ip address 10.0.0.1 255.255.255.0
 no negotiation auto
 load-balancing flow
!
interface Port-channel2.10
 ip rsvp authentication key 11223344
 ip rsvp authentication
!
interface Port-channel2.50
 encapsulation dot1Q 50
!
interface TenGigabitEthernet4/1/0
 no ip address
 negotiation auto
 cdp enable
 channel-group 2
!
```

Additional References

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

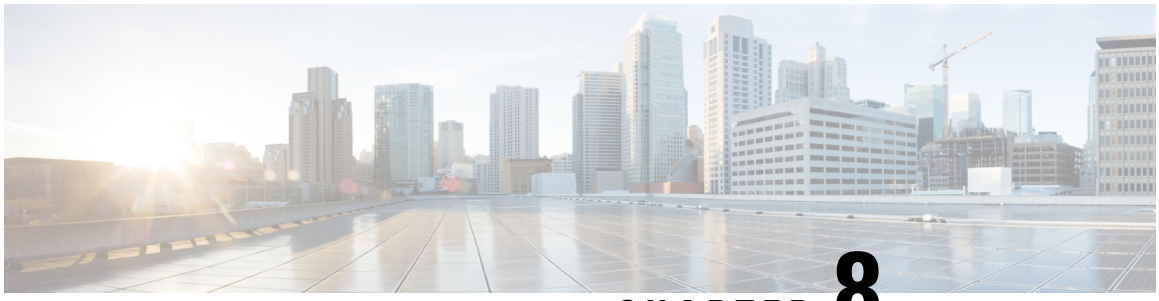
Feature Information for Flow-Based per Port-Channel Load Balancing

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 19: Feature Information for Flow-Based per Port-Channel Load Balancing

Feature Name	Releases	Feature Information
Flow-based per port-channel Load balancing	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on Cisco cBR Series Converged Broadband Routers.



CHAPTER 8

MPLS QoS via TLV for non-L2VPN Service Flow

The MPLS QoS via TLV for non-L2VPN Service Flow feature allows to mark TC bits for MPLS L3VPN imposition packets and classify downstream packets based on TC bits of MPLS disposition packets, using vendor-specific TLVs.

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 20: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

Restrictions for MPLS QoS via TLV for non-L2VPN Service Flow

- This feature supports only IPv4. It will not support IPv6.
- This feature does not support SNMP.
- This feature does not support dynamic service flows.
- Only up to four VPNs and eight upstream service flows per CM can be configured.
- For a VPN, only a maximum of eight DS classifiers (using TC bits in the range from 0 to 7) can be configured.
- If TC bits downstream classifiers are configured for a VPN, then the downstream MPLS packets belonging to the VPN are processed only on TC bits classification. It will not process general IP header field classification.

Information About MPLS QoS via TLV for non-L2VPN Service Flow

The MPLS QoS via TLV for non-L2VPN Service Flow feature is a QoS enhancement based on MPLS Traffic Class (TC) bits for MPLS L3VPN. The MPLS TC bits were previously known as MPLS EXP bits. RFC 5462 has renamed the MPLS EXP field to MPLS TC field.

For upstream service flow encoding, use Cisco-specific TLV to set TC bits value for MPLS imposition packets. For downstream classifier encoding, use Cisco-specific TLV to implement downstream classification based on TC bits of MPLS disposition packets.

Configuring MPLS QoS via TLV for non-L2VPN Service Flow

Note This feature is configured using a cable modem configuration file and is dependent on the general configuration of the L3VPN.

This section describes how to configure traffic class bits for MPLS imposition and disposition packets and on how to use vendor-specific TLVs with AToM L2VPN and MPLS L3VPN.

Traffic Class for MPLS Imposition Packets

The table lists the vendor-specific TLV to be included in the cable modem configuration file to configure TC bits for MPLS imposition packets. The MPLS-TC-SET TLV is defined in the upstream and is associated with the VPN RD in upstream service flow encoding.

Table 21: TLV to Configure TC Bits for MPLS Imposition Packets

TLV Name	SubType	Length	Value
MPLS-TC-SET TLV	43.5.43.34	1	Imposition MPLS-TC-SET bits

Traffic Classification for MPLS Disposition Packets

The table lists the vendor-specific TLV to be included in the cable modem configuration file to classify DS packets based on TC bits of MPLS disposition packets.

The MPLS-TC-RANGE TLV is defined only under DS classifier encodings. It supports multi-downstream flow in a CM belonging to the same MPLS L3VPN, associated with the VPN RD in downstream classifier encoding.

Table 22: TLV to Classify TC Bits for MPLS Disposition Packets

TLV Name	SubType	Length	Value
MPLS-TC-RANGE	43.5.43.35	2	MPLS-TC-low and MPLS-TC-high

Using Vendor-Specific TLVs with AToM L2VPN and MPLS L3VPN

If both AToM L2VPN (L2 MPLS) and MPLS L3VPN (L3 MPLS) are using the same set of TLVs (MPLS-TC-SET and MPLS-TC-RANGE), then you should differentiate them. Configure the TLVs for upstream service flow encoding and downstream classifier encodings as indicated below:

Upstream Service Flow Encoding

- For L2VPN, configure MPLS-TC-SET (43.5.43.34) and L2VPN ID (43.5.1).
- For MPLS L3VPN, configure MPLS-TC-SET (43.5.43.34) and VPN RD (43.5.1).

Note Do not configure the TLVs for L2VPN and MPLS L3VPN at the same time for upstream service flow encodings, as it will result in a TLV error.

Downstream Classifier Encoding

- L2VPN—Configure MPLS-TC-RANGE (43.5.43.35) and L2VPN ID (43.5.1).
- MPLS L3VPN—Configure MPLS-TC-RANGE (43.5.43.35) and VPN RD (43.5.1).

Configuration Examples

This section provides the following configuration examples:

Example: Upstream Service Flow Marking TLV

The following example shows a sample CM configuration TLV for the provisioning of TC bits for MPLS imposition packets:

```
24 (Upstream Service Flow Encoding)
   S01 (Service Flow Reference)         = 2
   S06 (QoS Parameter Set Type)         = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID)  = 00 00 0c
       T004 (VPN Route Distinguisher) = xx xx xx xx xx xx xx xx
            S005 (Vendor specific L2VPN TLV)
            S043 (Cisco Vendor Specific)
       T034 (MPLS-TC-SET) = 04 # MPLSTC-SET = 4
```

Example: Downstream Packet Classification TLV

The following example shows a sample CM configuration TLV for classifying downstream packets based on TC bits of MPLS disposition packets:

```
23 (Downstream Packet Classification Encoding)
   S01 (Classifier Reference)           = 13
   S03 (Service Flow Reference)         = 13
   S11 (IEEE 802.1P/Q Packet Classification Encodings)
   S43 (Vendor Specific Options)
       T08 (Vendor ID)  = 00 00 0c
```

```
    S004 (VPN Route Distinguisher) = xx xx xx xx xx xx xx xx
    S005 (Vendor specific L2VPN TLV)
        S043 (Cisco Vendor Specific)
        S035 (MPLS-TC-RANGE)   = 04  05      # MPLSTC-EGRESS_RANGE= 4 - 5
```

Example: MPLS QoS Configuration File

The following example shows a cable modem being configured to mark TC bits for MPLS L3VPN imposition packets and classify downstream packets based on TC bits of MPLS L3VPN disposition packets, using vendor-specific TLVs:

```
CM-CONFIG
=========
03 (Net Access Control)                      = 1
18 (Maximum Number of CPE)       = 16
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 2
   S03 (Service Flow Reference)        = 2
   S05 (Rule Priority)                             = 2
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)             = 00 20 ff
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 3
   S03 (Service Flow Reference)        = 3
   S05 (Rule Priority)                             = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)             = 40 80 ff
22 (Upstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 4
   S03 (Service Flow Reference)        = 4
   S05 (Rule Priority)                             = 4
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)             = a0 e0 ff
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 12
   S03 (Service Flow Reference)        = 12
   S05 (Rule Priority)                             = 2
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)             = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)            = 00 00 0c
       T004 (Unknown sub-type)          = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)          = 2b 09 08 03 00 00 0c 23 02 01 01
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)               = 13
   S03 (Service Flow Reference)        = 13
   S05 (Rule Priority)                             = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)             = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)            = 00 00 0c
       T004 (Unknown sub-type)          = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)          = 2b 09 08 03 00 00 0c 23 02 02 02
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)               = 14
   S03 (Service Flow Reference)        = 14
   S05 (Rule Priority)                             = 4
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)             = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)            = 00 00 0c
```

```
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 09 08 03 00 00 0c 23 02 03 03
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 1
   S06 (QoS Parameter Set Type)     = 7
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 2
   S06 (QoS Parameter Set Type)     = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 08 08 03 00 00 0c 22 01 04
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 3
   S06 (QoS Parameter Set Type)     = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 08 08 03 00 00 0c 22 01 05
24 (Upstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 4
   S06 (QoS Parameter Set Type)     = 7
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 08 08 03 00 00 0c 22 01 06
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 11
   S06 (QoS Parameter Set Type)     = 7
   S07 (Traffic Priority)                        = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 12
   S06 (QoS Parameter Set Type)     = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 13
   S06 (QoS Parameter Set Type)     = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 14
   S06 (QoS Parameter Set Type)     = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 15
   S06 (QoS Parameter Set Type)     = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 16
   S06 (QoS Parameter Set Type)     = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 17
   S06 (QoS Parameter Set Type)     = 7
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 18
   S06 (QoS Parameter Set Type)     = 7
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 19
   S03 (Service Flow Reference)        = 19
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)            = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 09 08 03 00 00 0c 23 02 00 00
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 15
   S03 (Service Flow Reference)        = 15
   S05 (Rule Priority)                           = 3
```

```
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)            = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 09 08 03 00 00 0c 23 02 04 04
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 16
   S03 (Service Flow Reference)        = 16
   S05 (Rule Priority)                             = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)            = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 09 08 03 00 00 0c 23 02 05 05
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 17
   S03 (Service Flow Reference)        = 17
   S05 (Rule Priority)                             = 3
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)            = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 09 08 03 00 00 0c 23 02 06 06
23 (Downstream Packet Classification Encoding Block)
   S01 (Classifier Reference)              = 18
   S03 (Service Flow Reference)        = 18
   S09 (IP Packet Encodings)
       T01 (IP Type of Srv Rng & Mask)            = 00 ff ff
   S43 (Vendor Specific Options)
       T08 (Vendor ID)           = 00 00 0c
       T004 (Unknown sub-type)         = 00 00 00 01 00 00 00 01
       T005 (Unknown sub-type)         = 2b 09 08 03 00 00 0c 23 02 07 07
25 (Downstream Service Flow Encodings)
   S01 (Service Flow Reference)        = 19
   S06 (QoS Parameter Set Type)     = 7
#<EOF>
```

Additional References

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

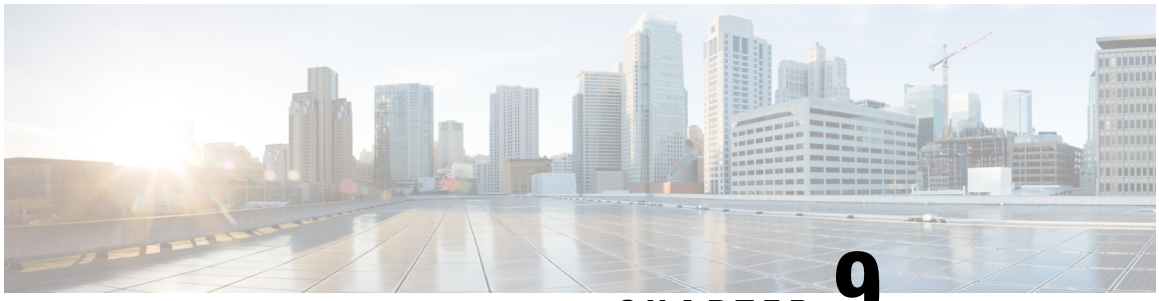
Feature Information for MPLS QoS via TLV for non-L2VPN Service Flow

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 23: Feature Information for MPLS QoS via TLV for non-L2VPN Service Flow

Feature Name	Releases	Feature Information
MPLS QoS via TLV for non-L2VPN Service Flow	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on the Cisco cBR Series Converged Broadband Routers.
MPLS QoS via TLV for non-L2VPN Service Flow	Cisco IOS XE Everest 16.6.1	This feature was integrated into Cisco IOS XE Everest 16.6.1 on the Cisco cBR Series Converged Broadband Routers.



CHAPTER 9

IPsec Security Support

IPsec is a security framework of open standards developed by the IETF. IPsec enables security for information that is send over unprotected networks. IPsec acts at the network layer, protecting and authenticating IP packets between participating IPsec devices (“peers”), such as Cisco routers.

Finding Feature Information

Your software release may not support all the features that are documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. The Feature Information Table at the end of this document provides information about the documented features and lists the releases in which each feature is supported.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/. An account on http://www.cisco.com/ is not required.

Hardware Compatibility Matrix for the Cisco cBR Series Routers

Note The hardware components that are introduced in a given Cisco IOS-XE Release are supported in all subsequent releases unless otherwise specified.

Table 24: Hardware Compatibility Matrix for the Cisco cBR Series Routers

Cisco CMTS Platform	Processor Engine	Interface Cards
Cisco cBR-8 Converged Broadband Router	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 Supervisor: • PID—CBR-SUP-250G • PID—CBR-CCAP-SUP-160G	**Cisco IOS-XE Release 16.5.1 and Later Releases** Cisco cBR-8 CCAP Line Cards: • PID—CBR-LC-8D30-16U30 • PID—CBR-LC-8D31-16U30 • PID—CBR-RF-PIC • PID—CBR-RF-PROT-PIC • PID—CBR-CCAP-LC-40G • PID—CBR-CCAP-LC-40G-R • PID—CBR-CCAP-LC-G2-R • PID—CBR-SUP-8X10G-PIC • PID—CBR-2X100G-PIC Digital PICs: • PID—CBR-DPIC-8X10G • PID—CBR-DPIC-2X100G Cisco cBR-8 Downstream PHY Module: • PID—CBR-D31-DS-MOD Cisco cBR-8 Upstream PHY Modules: • PID—CBR-D31-US-MOD

IPsec Security Support

Cisco IOS XE Amsterdam 17.2.x provides limited support for up to 16 Gbps encrypted IPsec that is sent or forwarded by cBR8.

IPsec is a security framework of open standards developed by the IETF. IPsec enables security for information that is send over unprotected networks. IPsec acts at the network layer, protecting and authenticating IP packets between participating IPsec devices (“peers”), such as Cisco routers.

IPsec is mainly for securing lawful intercept (LI) traffic from cBR8 to MAC Domain profile. The IPsec feature now supports:

- AES-CBC-128 encryption

- HMAC-SHA-256 authentication
- ESP tunnel mode
- IKEv2 with certificate or preshared key
- PFS (Perfect Forward Secrecy)

IPsec Security Limitations

The IPsec feature for Cisco IOS XE Amsterdam 17.2.1 has the following limitations:

- Only supported on SUP 160.
- The RX path of IPsec tunnel only supports minimum control traffic. The traffic is punted to IOSd, and is heavily rate-limited. The default limit is 200 packets/second (configurable).

Configuring IPsec Security

To configure the IPsec security, complete the following steps:

1. Use the **crypto ipsec transform-set <ts-name> esp-aes esp-sha256-hmac** command. However, note that only the following options are supported:
   - Support for **esp-aes esp-sha256-hmac**
   - Support for *mode tunnel*

   You can optionally use **set pfs <dh-group-name>** to enable perfect forward secrecy in IPsec profile.
2. Use the **crypto ipsec profile <profile-name>**, where the IKEv2 profile is set into IPsec profile.
3. Use the **tunnel protection ipsec profile** tunnel interface.

To view your IPsec information, use the **show crypto ipsec sa detail** command:

```
Router# show crypto ipsec sa detail
Load for five secs: 3%/0%; one minute: 8%; five minutes: 4%
Time source is NTP, 12:40:49.195 EDT Wed Feb 26 2020


interface: Tunnel101
    Crypto map tag: Tunnel101-head-0, local addr 102.0.0.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (102.0.0.2/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (102.0.0.1/255.255.255.255/47/0)
   current_peer 102.0.0.1 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
```

```
    #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
    #pkts invalid prot (recv) 0, #pkts verify failed: 0
    #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
    #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
    ##pkts replay failed (rcv): 0
    #pkts tagged (send): 0, #pkts untagged (rcv): 0
    #pkts not tagged (send): 0, #pkts not untagged (rcv): 0
    #pkts internal err (send): 0, #pkts internal err (recv) 0

     local crypto endpt.: 102.0.0.2, remote crypto endpt.: 102.0.0.1
     plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb TenGigabitEthernet4/1/0
     current outbound spi: 0xBD3A2CBF(3174706367)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0xC67787E8(3329722344)
        transform: esp-aes esp-sha256-hmac ,
        in use settings ={Tunnel, }
        conn id: 2, flow_id: SW:2, sibling_flags FFFFFFFF80000040, crypto map:
Tunnel101-head-0
         sa timing: remaining key lifetime (k/sec): (4242079/86293)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xBD3A2CBF(3174706367)
        transform: esp-aes esp-sha256-hmac ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, sibling_flags FFFFFFFF80000040, crypto map:
Tunnel101-head-0
         sa timing: remaining key lifetime (k/sec): (4242079/86293)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
```

Configuring Transform Sets for IKEv2

You can choose to configure the IKEv2 using eitheir of the following options:

- IKEv2 with pre-shared key. This includes the following options:
  - **crypto ikev2 proposal <proposal-name>**.
  - **crypto ikev2 policy <policy-name>**
  - **crypto ikev2 keyring <keyring-name>**

    Set keyring in IKEv2 profile. A configuration example using IKEv2 with pre-shared key is as shown:

```
crypto ikev2 proposal li-ikev2-proposal
 encryption aes-cbc-128
 integrity sha256
```

```
 group 5 2
crypto ikev2 policy li-ikev2-policy
 match address local 102.0.0.2
 proposal li-ikev2-proposal
crypto ikev2 keyring li-kyr
 peer li-peer
 address 102.0.0.1 255.255.255.0
 identity address 102.0.0.2
 pre-shared-key key1
 !
crypto ikev2 profile li-profile
 match address local interface TenGigabitEthernet4/1/7
 match identity remote address 102.0.0.1 255.255.255.255
 authentication remote pre-share
 authentication local pre-share key key1
 keyring local li-kyr
crypto ipsec transform-set TS esp-aes esp-sha256-hmac
 mode tunnel
crypto ipsec profile li-ipsec-gre
 set security-association lifetime seconds 86400
 set transform-set TS
 set pfs group14
 set ikev2-profile li-profile
```

- IKEv2 with certificate authority. This includes the following steps:
  1. Generate the RSA key pair.
  2. Configure the PKI trustpoint. This requires the CA server supporting SCEP (Simple Certificate Enrollment Protocol).

     Configure **crypto pki trustpoint** to enroll to CA. Note that the *subject-name* will be used for authentication in the example
  3. Set the certificate map in IKEv2 profile by configuring **crypto pki certificate map <map-name> <id>** to match the certificate content.
  4. Enroll the certificate.

To view your IPsec information, use the **show crypto ikev2 sa detail** command:

```
Router# show crypto ikev2 sa detail
Load for five secs: 3%/0%; one minute: 8%; five minutes: 4%
Time source is NTP, 12:40:57.672 EDT Wed Feb 26 2020

 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         102.0.0.2/500         102.0.0.1/500         none/none            READY
      Encr: AES-CBC, keysize: 128, PRF: SHA256, Hash: SHA256, DH Grp:5, Auth sign: RSA,
Auth verify: RSA
      Life/Active Time: 86400/115 sec
      CE id: 1001, Session-id: 1
      Status Description: Negotiation done
      Local spi: A3C274EBBD7FFF2F       Remote spi: AA160367FFD29C2D
      Local id: hostname=tb34-cBR8.cisco.com,cn=ANSSI Test CBR8
      Remote id: hostname=cCMTS-bcl-ASR1K6-2,cn=ANSSI Test ASR1K
      Local req msg id:  2              Remote req msg id:  0
      Local next msg id: 2              Remote next msg id: 0
      Local req queued:  2              Remote req queued:  0
      Local window:      5              Remote window:      5
```

```
      DPD configured for 0 seconds, retry 0
      Fragmentation not  configured.
      Dynamic Route Update: enabled
      Extended Authentication not configured.
      NAT-T is not detected
      Cisco Trust Security SGT is disabled
      Initiator of SA : Yes

IPv6 Crypto IKEv2  SA
```

Note The IPsec and IKEv2 are configured in the same way as ASR 1000. Go through the ASR 1000 Internet Key Exchange for IPsec VPNs Configuration Guide for more information. The following limitations apply:

- Supported encryption
- Authentication algorithms
- ESP tunnel mode

Feature Information for IPsec Security Support

Use Cisco Feature Navigator to find information about the platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to the https://cfnng.cisco.com/ link. An account on the Cisco.com page is not required.

Note The following table lists the software release in which a given feature is introduced. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 25: Feature Information for IPsec Security Support

Feature Name	Releases	Feature Information
IPsec Security Support	Cisco IOS XE Amsterdam 17.2.1	This feature was integrated into Cisco IOS XE Amsterdam 17.2.1 on the Cisco cBR Series Converged Broadband Routers.

INDEX

E

M