Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software

This module provides the conceptual and configuration information for MPLS Layer 2 virtual private networks (VPNs) on Cisco IOS XR software.

For the functionality of MPLS VPNs over IP Tunnels, see Implementing MPLS VPNs over IP Tunnels on Cisco IOS XR Software in Cisco IOS XR MPLS Configuration Guide.

Note

For more information about MPLS Layer 2 VPN on the Cisco IOS XR software and for descriptions of the commands listed in this module, see the “Related Documents” section. To locate documentation for other commands that might appear while executing a configuration task, search online in the Cisco IOS XR software master command index.

Feature History for Implementing MPLS Layer 2 VPN on Cisco IOS XR Configuration Module

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 3.4.0</td>
<td>This feature was introduced on the Cisco CRS-1 and Cisco XR 12000 Series Router.</td>
</tr>
<tr>
<td>Release 3.4.1</td>
<td>Support was added for:</td>
</tr>
<tr>
<td></td>
<td>• Virtual Circuit Connection Verification (VCCV) on L2VPN</td>
</tr>
<tr>
<td></td>
<td>• Layer 2 VPN (L2VPN) Quality of Service (QoS) for Ethernet-over-MPLS (EoMPLS) on the Cisco CRS-1</td>
</tr>
<tr>
<td></td>
<td>• QinQ mode and QinAny mode for EoMPLS on the Cisco XR 12000 Series Router</td>
</tr>
<tr>
<td>Release 3.5.0</td>
<td>Support was added for:</td>
</tr>
<tr>
<td></td>
<td>• EoMPLS Inter-AS mode</td>
</tr>
<tr>
<td></td>
<td>• Mac-in-Mac protocol</td>
</tr>
</tbody>
</table>
Prerequisites for Implementing MPLS L2VPN on Cisco IOS XR Software

You must be in a user group associated with a task group that includes the proper task IDs for MPLS L2VPN commands. For detailed information about user groups and task IDs, see the Configuring AAA Services on Cisco IOS XR Software module in the Cisco IOS XR System Security Configuration Guide.

Information About Implementing L2VPN

To implement MPLS L2VPN, you should understand the following concepts:

- L2VPN Overview, page MPC-217
- ATMoMPLS with L2VPN Capability, page MPC-217
- Virtual Circuit Connection Verification on L2VPN, page MPC-218
- Ethernet over MPLS, page MPC-218
- Quality of Service, page MPC-222
- High Availability, page MPC-223
- Preferred Tunnel Path, page MPC-223
L2VPN Overview

Layer 2 VPN (L2VPN) emulates the behavior of a LAN across an IP or MPLS-enabled IP network allowing Ethernet devices to communicate with each other as they would when connected to a common LAN segment.

As Internet service providers (ISPs) look to replace their Frame Relay or Asynchronous Transfer Mode (ATM) infrastructures with an IP infrastructure, there is a need for to provide standard methods of using an IP infrastructure to provide a serviceable L2 interface to customers; specifically, to provide standard ways of using an IP infrastructure to provide virtual circuits between pairs of customer sites.

Building a L2VPN system requires coordination between the ISP and the customer. The ISP provides L2 connectivity; the customer builds a network using data link resources obtained from the ISP. In an L2VPN service, the ISP does not require information about the customer's network topology, policies, routing information, point-to-point links, or network point-to-point links from other ISPs.

The ISP requires provider edge (PE) routers with the following capabilities:
- Encapsulation of L2 protocol data units (PDU) into Layer 3 (L3) packets.
- Interconnection of any-to-any L2 transports.
- Emulation of L2 quality-of-service (QoS) over a packet switch network.
- Ease of configuration of the L2 service.
- Support for different types of tunneling mechanisms (MPLS, L2TPv3, IPSec, GRE, and others).
- L2VPN process databases include all information related to circuits and their connections.

ATMoMPLS with L2VPN Capability

This feature is supported on the Cisco CRS-1 router and the Cisco XR 12000 Series Router.

These topics describe the ATM over MPLS (ATMoMPLS) with L2VPN feature:
- ATMoMPLS with L2VPN Overview, page MPC-217
- Layer 2 Local Switching Overview, page MPC-218
- ATM Adaptation Layer 5, page MPC-218

ATMoMPLS with L2VPN Overview

The ATMoMPLS feature supports ATM Adaptation Layer 5 (AAL5) transport. ATMoMPLS is a type of Layer 2 point-to-point connection over an MPLS core. ATMoMPLS and ATM local switching are supported only for ATM-to-ATM interface-to-interface switching combinations.

To implement the ATMoMPLS feature, the Cisco CRS-1 router plays the role of provider edge (PE) router at the edge of a provider network in which customer edge (CE) devices are connected to the Cisco CRS-1 routers.
Layer 2 Local Switching Overview

Local switching lets you to switch Layer 2 data between two interfaces of the same type (for example, ATM-to-ATM, or Frame Relay-to-Frame Relay) or between interfaces of different types (for example, Frame Relay to ATM) on the same router. The interfaces are on the same line card or on two different cards. During these types of switching, Layer 2 address is used instead of the Layer 3 address.

In addition, same-port local switching lets you to switch Layer 2 data between two circuits on the same interface.

ATM Adaptation Layer 5

AAL5 lets you transport AAL5 PDUs from various customers over an MPLS backbone. ATM AAL5 extends the usability of the MPLS backbone by enabling it to offer Layer 2 services in addition to already existing Layer 3 services. You can enable the MPLS backbone network to accept AAL5 PDUs by configuring the provider edge (PE) routers at both ends of the MPLS backbone.

To transport AAL5 PDUs over MPLS, a virtual circuit is set up from the ingress PE router to the egress PE router. This virtual circuit transports the AAL5 PDUs from one PE router to the other. Each AAL5 PDU is transported as a single packet.

Virtual Circuit Connection Verification on L2VPN

Virtual Circuit Connection Verification (VCCV) is an L2VPN Operations, Administration, and Maintenance (OAM) feature that allows network operators to run IP-based provider edge-to-provider edge (PE-to-PE) keepalive protocol across a specified pseudowire to ensure that the pseudowire data path forwarding does not contain any faults. The disposition PE receives VCCV packets on a control channel, which is associated with the specified pseudowire. The control channel type and connectivity verification type, which are used for VCCV, are negotiated when the pseudowire is established between the PEs for each direction.

Two types of packets can arrive at the disposition egress:

- Type 1—Specifies normal Ethernet-over-MPLS (EoMPLS) data packets.
- Type 2—Specifies VCCV packets.

Cisco IOS XR software supports Label Switched Path (LSP) VCCV Type 1, which uses an inband control word if enabled during signaling. The VCCV echo reply is sent as IPv4 that is the reply mode is IPv4. The reply is forwarded as IP, MPLS, or a combination of both.

VCCV pings counters that are counted in MPLS forwarding on the egress side. However, on the ingress side, they are sourced by the route processor and do not count as MPLS forwarding counters.

Ethernet over MPLS

Ethernet-over-MPLS (EoMPLS) provides a tunneling mechanism for Ethernet traffic through an MPLS-enabled L3 core and encapsulates Ethernet protocol data units (PDUs) inside MPLS packets (using label stacking) to forward them across the MPLS network.

EoMPLS features are described in the following subsections:

- Ethernet Port Mode, page MPC-219
- Ethernet Remote Port Shutdown, page MPC-219
Ethernet Port Mode

In Ethernet port mode, both ends of a pseudowire are connected to Ethernet ports. In this mode, the port is tunneled over the pseudowire or, using local switching (also known as an attachment circuit-to-attachment circuit cross-connect) switches packets or frames from one attachment circuit (AC) to another AC attached to the same PE node.

Figure 15 provides an example of Ethernet port mode.

Ethernet Remote Port Shutdown

Ethernet remote port shutdown provides a mechanism for the detection and propagation of remote link failure for port mode EoMPLS on a Cisco CRS-1 line card. This lets a service provider edge router on the local end of an Ethernet-over-MPLS (EoMPLS) pseudowire detect a cross-connect or remote link failure and cause the shutdown of the Ethernet port on the local customer edge router. Shutting down the Ethernet port on the local customer edge router prevents or mitigates a condition where that router would otherwise lose data by forwarding traffic continuously to the remote failed link, especially if the link were configured as a static IP route (see Figure 16).
To enable this functionality, see the `l2transport propagate` command in *Cisco IOS XR MPLS Command Reference*.

**Note**

Ethernet remote port shutdown is supported only on the Cisco CRS-1 router.

### VLAN Mode

In VLAN mode, each VLAN on a customer-end to provider-end link can be configured as a separate L2VPN connection using virtual connection (VC) type 4 or VC type 5. VC type 5 is the default mode.

As illustrated in *Figure 17*, the Ethernet PE associates an internal VLAN-tag to the Ethernet port for switching the traffic internally from the ingress port to the pseudowire; however, before moving traffic into the pseudowire, it removes the internal VLAN tag.

**Note**

Because the port is in trunk mode, the VLAN PE doesn't remove the VLAN tag and forwards the frames through the port with the added tag.
Inter-AS Mode

Inter-AS is a peer-to-peer type model that allows extension of VPNs through multiple provider or multi-domain networks. This lets service providers peer up with one another to offer end-to-end VPN connectivity over extended geographical locations.

EoMPLS support can assume a single AS topology where the pseudowire connecting the PE routers at the two ends of the point-to-point EoMPLS cross-connects resides in the same autonomous system; or multiple AS topologies in which PE routers can reside on two different ASs using i-BGP and e-BGP peering.

Figure 18 illustrates MPLS over Inter-AS with a basic double AS topology with iBGP/LDP in each AS.

QinQ Mode

In QinQ mode, each CE VLAN is carried into an SP VLAN. QinQ mode should use VC type 5, but VC type 4 is also supported. On each Ethernet PE, you must configure both the inner (CE VLAN) and outer (SP VLAN).

Figure 19 illustrates QinQ using VC type 4.
QinAny Mode

In the QinAny mode, the service provider VLAN tag is configured on both the ingress and the egress nodes of the provider edge VLAN. QinAny mode is similar to QinQ mode using a Type 5 VC, except that the customer edge VLAN tag is carried in the packet over the pseudowire, as the customer edge VLAN tag is unknown.

Note
The QinAny mode is supported on the Cisco XR 12000 Series Router only.

Mac-in-Mac Protocol (Provide Backbone Bridging)

The Mac-in-Mac (or, Provider Backbone Bridging) protocol lets service providers scale networks using Ethernet technology to maintain management and operational simplicity, and reduce operating costs.

Mac-In-Mac encapsulates the customer MAC header with a service provider MAC header. Instead of using additional Q-tags to separate end customers, a 24-bit service tag in the service provider encapsulating MAC header is used, which provides support for up to 16-million service instances.

Note
Mac-In-Mac is standardized as IEEE 802.1ah.

Quality of Service

Using L2VPN technology, you can assign a quality of service (QoS) level to both Port and VLAN modes of operation.

L2VPN technology requires that QoS functionality on PE routers be strictly L2-payload-based on the edge-facing interfaces (also know as attachment circuits). Figure 20 illustrates L2 and L3 QoS service policies in a typical L2VPN network.

Figure 20  L2VPN QoS Feature Application

Figure 21 shows four packet processing paths within a provider edge device where a QoS service policy can be attached. In an L2VPN network, packets are received and transmitted on the edge-facing interfaces as L2 packets and transported on the core-facing interfaces as MPLS (EoMPLS) or IP (L2TP) packets.
High Availability

L2VPN uses control planes in both route processors and line cards, as well as forwarding plane elements in the line cards.

Note

The l2tp_mgr process does not support high availability.

The availability of L2VPN meets the following requirements:

- A control plane failure in either the route processor or the line card will not affect the circuit forwarding path.
- The router processor control plane supports failover without affecting the line card control and forwarding planes.
- L2VPN integrates with existing Label Distribution Protocol (LDP) graceful restart mechanism.

Preferred Tunnel Path

Preferred tunnel path functionality lets you map pseudowires to specific traffic-engineering tunnels. Attachment circuits are cross-connected to specific MPLS traffic engineering tunnel interfaces instead of remote PE router IP addresses (reachable using IGP or LDP). Using preferred tunnel path, it is always assumed that the traffic engineering tunnel that transports the L2 traffic runs between the two PE routers (that is, its head starts at the imposition PE router and its tail terminates on the disposition PE router).

Note

- Currently, preferred tunnel path configuration applies only to MPLS encapsulation.
- The fallback enable option is supported only on the Cisco XR 12000 Series Router.
How to Implement L2VPN

This section describes the tasks required to implement L2VPN:

- Configuring an Interface or Connection for L2VPN, page MPC-224
- Configuring Static Point-to-Point Cross-Connects, page MPC-226
- Configuring Dynamic Point-to-Point Cross-Connects, page MPC-228
- Configuring Inter-AS, page MPC-230
- Configuring L2VPN Quality of Service, page MPC-230
- Configuring Preferred Tunnel Path, page MPC-234

Configuring an Interface or Connection for L2VPN

Perform this task to configure an interface or a connection for L2VPN.

SUMMARY STEPS

1. configure
2. interface type interface-id
3. l2transport
4. exit
5. interface type interface-id
6. dot1q native vlan vlan-id
7. end
   or
   commit
8. show interface type interface-id

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2 interface type interface-id</td>
<td>Enters interface configuration mode and configures an interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

**Step 3**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>l2transport</td>
<td>Enables L2 transport on the selected interface.</td>
</tr>
</tbody>
</table>

**Example:**

RP/0/RP0/CPU0:router(config-if)# l2transport

**Step 4**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
</tbody>
</table>

**Example:**

RP/0/RP0/CPU0:router(config-if-l2)# exit

**Step 5**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface type interface-id</td>
<td>Enters interface configuration mode and configures an interface.</td>
</tr>
</tbody>
</table>

**Example:**

RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0

**Step 6**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>dot1q native vlan vlan ID</td>
<td>Assigns the native VLAN ID of a physical interface trunking 802.1Q VLAN traffic.</td>
</tr>
</tbody>
</table>

**Example:**

RP/0/RP0/CPU0:router(config-if)# dot1q vlan 1

**Step 7**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>end or commit</td>
<td>Saves configuration changes.</td>
</tr>
</tbody>
</table>

**Example:**

RP/0/RP0/CPU0:router(config-if)# end

or

RP/0/RP0/CPU0:router(config-if)# commit

- When you issue the end command, the system prompts you to commit changes:
  Uncommitted changes found, commit them before exiting(yes/no/cancel)?
  [cancel]:
  - Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

**Step 8**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show interface type interface-id</td>
<td>Displays the configuration settings you committed for the interface.</td>
</tr>
</tbody>
</table>

**Example:**

RP/0/RP0/CPU0:show interface

GigabitEthernet0/0/0/0/0
Configuring Static Point-to-Point Cross-Connects

Perform this task to configure static point-to-point cross-connects.

Please consider the following information about cross-connects when you configure static point-to-point cross-connects:

- An cross-connect is uniquely identified with the pair; the cross-connect name must be unique within a group.
- A segment (an attachment circuit or pseudowire) is unique and can belong only to a single cross-connect.
- A static VC local label is globally unique and can be used in one pseudowire only.
- No more than 16,000 cross-connects can be configured per router.

**Note**

Static pseudowire connections do not use LDP for signaling.

**SUMMARY STEPS**

1. configure
2. l2vpn
3. xconnect group group name
4. p2p xconnect name
5. interface type interface-id
6. neighbor A.B.C.D pw-id pseudowire ID
7. mpls static label local {value} remote {value}
8. end
   or
9. commit

9. show l2vpn xconnect group group name

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>
| **Example:**
RP/0/RP0/CPU0:router# configure |
| **Step 2** l2vpn | Enters L2VPN configuration mode. |
| **Example:**
RP/0/RP0/CPU0:router(config)# l2vpn |
### Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software

#### How to Implement L2VPN

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td><code>xconnect group group name</code></td>
<td>Enters the name of the cross-connect group.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-l2vpn)# xconnect group vlan_grp_1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td><code>p2p xconnect name</code></td>
<td>Enters a name for the point-to-point cross-connect.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-l2vpn-xc)# p2p vlan1</td>
<td></td>
</tr>
</tbody>
</table>
| 5    | `interface type interface-id` | Specifies the interface type ID. The choices are:  
  - GigabitEthernet: GigabitEthernet/IEEE 802.3 interfaces.  
  - TenGigE: TenGigabitEthernet/IEEE 802.3 interfaces. |
|      | **Example:** RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# interface GigabitEthernet0/0/0/0.1 |         |
| 6    | `neighbor A.B.C.D pw-id pseudowire ID` | Configures the pseudowire segment for the cross-connect.  
  Optionally, you can disable the control word or set the transport-type to "Ethernet" or "VLAN". |
|      | **Example:** RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 2.2.2.2 pw-id 2000 |         |
| 7    | `mpls static label local {value} remote {value}` | Configures local and remote label ID values. |
|      | **Example:** RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p-pw)# mpls static label local 699 remote 890 |         |
Configuring Dynamic Point-to-Point Cross-Connects

Perform this task to configure dynamic point-to-point cross-connects.

Note

For dynamic cross-connects, LDP must be up and running.

<table>
<thead>
<tr>
<th>SUMMARY STEPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. configure</td>
</tr>
<tr>
<td>2. l2vpn</td>
</tr>
<tr>
<td>3. xconnect group group name</td>
</tr>
<tr>
<td>4. p2p xconnect name</td>
</tr>
<tr>
<td>5. interface type interface-id</td>
</tr>
<tr>
<td>6. neighbor A.B.C.D pw-id pseudowire ID</td>
</tr>
<tr>
<td>7. end or commit</td>
</tr>
</tbody>
</table>
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters the configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> l2vpn</td>
<td>Enters L2VPN configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config)# l2vpn</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> xconnect group group name</td>
<td>Enters the name of the cross-connect group.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-l2vpn)# xconnect group grp_1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> p2p xconnect name</td>
<td>Enters a name for the point-to-point cross-connect.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-l2vpn-xc)# p2p vlan1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> interface type interface-id</td>
<td>Specifies the interface type ID. The choices are:</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)# interface GigabitEthernet0/0/0.1</td>
<td></td>
</tr>
<tr>
<td>- GigabitEthernet: GigabitEthernet/IEEE 802.3 interfaces.</td>
<td></td>
</tr>
<tr>
<td>- TenGigE: TenGigabitEthernet/IEEE 802.3 interfaces.</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Inter-AS

The Inter-AS configuration procedure is identical to the L2VPN cross-connect configuration tasks (see “Configuring Static Point-to-Point Cross-Connects” section on page MPC-226 and “Configuring Dynamic Point-to-Point Cross-Connects” section on page MPC-228) except that the remote PE IP address used by the cross-connect configuration is now reachable through iBGP peering.

Note

You must be knowledgeable about IBGP, EBGP, and ASBR terminology and configurations to complete this configuration.

Configuring L2VPN Quality of Service

This section describes how to configure L2VPN quality of service (QoS) in port mode and VLAN mode.

Restrictions

The l2transport command cannot be used with any IP address, L3, or CDP configuration.
Configuring an L2VPN Quality of Service Policy in Port Mode

This procedure describes how to configure an L2VPN QoS policy in port mode.

**Note**

In port mode, the interface name format does not include a subinterface number; for example, GigabitEthernet0/1/0/1.

**SUMMARY STEPS**

1. `configure`
2. `interface type interface-id.subinterface`
3. `l2transport`
4. `service-policy [input | output] [policy-map-name]`
5. `end`
   or `commit`
6. `show qos interface type interface-path-id.subinterface service-policy [input | output] [policy-map-name]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>configure</code></td>
<td>Enters the configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>interface type interface-id.subinterface</code></td>
<td>Specifies the interface attachment circuit.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0.1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>l2transport</code></td>
<td>Configures an interface or connection for L2 switching.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-if)# l2transport</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> `service-policy [input</td>
<td>output] [policy-map-name]`</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config-if)# service-policy input servpol1</td>
<td></td>
</tr>
</tbody>
</table>
How to Implement L2VPN in VLAN Mode

This procedure describes how to configure a L2VPN QoS policy in VLAN mode.

**Step 5**

**SUMMARY STEPS**

1. `configure`
2. `interface type interface-id.subinterface l2transport`
3. `service-policy [input | output] [policy-map-name]`
4. `end` or `commit`

**Note**

In VLAN mode, the interface name must include a subinterface; for example, GigabitEthernet0/1/0/1.1; and the `l2transport` command must follow the interface type on the same CLI line (for example, "interface GigabitEthernet0/0/0.1 l2transport").
### Detailed Steps

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RP0/CPU0:router# configure</td>
</tr>
<tr>
<td>Enters the configuration mode.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>interface type interface-id.subinterface l2transport</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0.1 l2transport</td>
</tr>
</tbody>
</table>
| Configures an interface or connection for L2 switching.  
**Note** In VLAN Mode, you must enter the `l2transport` keyword on the same line as the interface. |
| **Step 3** | service-policy [input | output] [policy-map-name] |
| **Example:** | RP/0/RP0/CPU0:router(config-if)# service-policy input servpol1 |
| Attaches a QoS policy to an input or output interface to be used as the service policy for that interface. |
| **Step 4** | end or commit |
| **Example:** | RP/0/RP0/CPU0:router(config-if)# end |
| or |
| RP/0/RP0/CPU0:router(config-if)# commit |
| Saves configuration changes.  
- When you issue the `end` command, the system prompts you to commit changes:  
  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  
  - Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.  
  - Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.  
  - Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.  
- Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session. |
Configuring Preferred Tunnel Path

This procedure describes how to configure a preferred tunnel path.

Note
The tunnel used for the preferred path configuration is an MPLS Traffic Engineering (MPLS-TE) tunnel.

SUMMARY STEPS

1. configure
2. l2vpn
3. pw-class \{name\}
4. encapsulation mpls
5. preferred-path \{interface\} \{tunnel-te value\} \{fallback disable\}
6. end
   or
   commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1  configure</td>
<td>Enters the configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2  l2vpn</td>
<td>Enters L2VPN configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# l2vpn</td>
<td></td>
</tr>
<tr>
<td>Step 3  pw-class {name}</td>
<td>Configures the pseudowire class name.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-l2vpn)# pw-class path1</td>
<td></td>
</tr>
<tr>
<td>Step 4  encapsulation mpls</td>
<td>Configures the pseudowire encapsulation to MPLS.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls</td>
<td></td>
</tr>
<tr>
<td>Step 5  preferred-path {interface} {tunnel-te value} {fallback disable}</td>
<td>Configures preferred path tunnel settings. If the fallback disable configuration is used and once the TE tunnel is configured as the preferred path goes down, the corresponding pseudowire can also go down.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-l2vpn-pwc-encap-mpls)# preferred-path interface tunnel 11 fallback disable</td>
<td></td>
</tr>
</tbody>
</table>
In the following example, two traffic classes are created and their match criteria are defined. For the first traffic class called class1, ACL 101 is used as the match criterion. For the second traffic class called class2, ACL 102 is used as the match criterion. Packets are checked against the contents of these ACLs to determine if they belong to the class.

This section includes the following configuration examples:

- L2VPN Interface Configuration: Example, page MPC-236
- Point-to-Point Cross-connect Configuration: Examples, page MPC-236
- Inter-AS: Example, page MPC-236
- L2VPN Quality of Service: Example, page MPC-238
- Preferred Path: Example, page MPC-238
L2VPN Interface Configuration: Example

The following example shows how to configure an L2VPN interface:

```config
configure
    interface GigabitEthernet0/0/0/0.1 l2transport
dot1q vlan 1
end
```

Point-to-Point Cross-connect Configuration: Examples

This section includes configuration examples for both static and dynamic point-to-point cross-connects.

Static Configuration

The following example shows how to configure a static point-to-point cross-connect:

```config
configure
    l2vpn
    xconnect group vlan_grp_1
    p2p vlan1
    interface GigabitEthernet0/0/0/0.1
    neighbor 2.2.2.2 pw-id 2000
    commit
```

Dynamic Configuration

The following example shows how to configure a dynamic point-to-point cross-connect:

```config
configure
    l2vpn
    xconnect group vlan_grp_1
    p2p vlan1
    interface GigabitEthernet0/0/0/0.1
    neighbor 2.2.1.1 pw-id 1
    commit
```

Inter-AS: Example

The following example shows how to set up an AC to AC cross connect from AC1 to AC2:

```config
router-id Loopback0

interface Loopback0
    ipv4 address 127.0.0.1 255.255.255.255

interface GigabitEthernet0/1/0/0.1 l2transport dot1q vlan 1

interface POS0/0/0/3
    ipv4 address 127.0.0.1 255.255.255.0
    keepalive disable

interface POS0/0/0/4
    ipv4 address 127.0.0.1 255.255.255.0
    keepalive disable

router ospf 100
```
log adjacency changes detail
area 0
  interface Loopback0
  !
  interface POS0/0/0/3
  !
  interface POS0/0/0/4
  !
  !
router bgp 100
  address-family ipv4 unicast
    allocate-label all
  !
  neighbor 40.0.0.5
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
  !
  address-family ipv4 labeled-unicast
  !
  !
l2vpn
  xconnect group xc1
    p2p ac2ac1
      interface GigabitEthernet0/1/0/0.1
      neighbor 20.0.0.5 pw-id 101
    !
    p2p ac2ac2
      interface GigabitEthernet0/1/0/0.2
      neighbor 20.0.0.5 pw-id 102
    !
    p2p ac2ac3
      interface GigabitEthernet0/1/0/0.3
      neighbor 20.0.0.5 pw-id 103
    !
    p2p ac2ac4
      interface GigabitEthernet0/1/0/0.4
      neighbor 20.0.0.5 pw-id 104
    !
    p2p ac2ac5
      interface GigabitEthernet0/1/0/0.5
      neighbor 20.0.0.5 pw-id 105
    !
    p2p ac2ac6
      interface GigabitEthernet0/1/0/0.6
      neighbor 20.0.0.5 pw-id 106
    !
    p2p ac2ac7
      interface GigabitEthernet0/1/0/0.7
      neighbor 20.0.0.5 pw-id 107
    !
    p2p ac2ac8
      interface GigabitEthernet0/1/0/0.8
      neighbor 20.0.0.5 pw-id 108
    !
    p2p ac2ac9
      interface GigabitEthernet0/1/0/0.9
      neighbor 20.0.0.5 pw-id 109
    !
    p2p ac2ac10
      interface GigabitEthernet0/1/0/0.10
      neighbor 20.0.0.5 pw-id 110
mpls ldp
router-id Loopback0
log
neighbor
!
interface POS0/0/0/3
!
interface POS0/0/0/4
!
end

L2VPN Quality of Service: Example

The following example shows how to attach a service-policy to an L2 interface in port mode:

```
configure
  interface type interface-id
  l2transport
  service-policy [input | output] [policy-map-name]
commit
```

Preferred Path: Example

The following example shows how to configure preferred tunnel path:

```
configure
  l2vpn
  pw-class path1
  encapsulation mpls
    preferred-path interface tunnel value fallback disable
```

Additional References

For additional information related to implementing MPLS Layer 2 VPN, refer to the following references:

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XR L2VPN command reference document</td>
<td><strong>MPLS Virtual Private Network Commands on Cisco IOS XR Software module in Cisco IOS XR MPLS Command Reference</strong></td>
</tr>
<tr>
<td>MPLS VPN-related commands</td>
<td><strong>MPLS Virtual Private Network Commands on Cisco IOS XR Software module in Cisco IOS XR MPLS Command Reference</strong></td>
</tr>
<tr>
<td>MPLS Layer 2 VPNs</td>
<td><strong>Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software module in Cisco IOS XR MPLS Configuration Guide</strong></td>
</tr>
</tbody>
</table>
### Standards

<table>
<thead>
<tr>
<th>Standards¹</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.</td>
<td>—</td>
</tr>
</tbody>
</table>

1. Not all supported standards are listed.

### MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>—</td>
<td>To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: <a href="http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml">http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml</a></td>
</tr>
</tbody>
</table>

### RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 3931</td>
<td><em>Layer Two Tunneling Protocol - Version 3 (L2TPv3)</em></td>
</tr>
<tr>
<td>RFC 4447</td>
<td><em>Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)</em>, April 2006</td>
</tr>
<tr>
<td>RFC 4448</td>
<td><em>Encapsulation Methods for Transport of Ethernet over MPLS Networks</em>, April 2006</td>
</tr>
</tbody>
</table>
## Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.</td>
<td><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>