



## L2VPN Protocol-Based CLIs

---

The L2VPN Protocol-Based CLIs feature provides a set of processes and an improved infrastructure for developing and delivering Cisco IOS software on various Cisco platforms. This feature introduces new commands and modifies or replaces existing commands to achieve a consistent functionality across Cisco platforms and provide cross-Operating System (OS) support.

- [Information About L2VPN Protocol-Based CLIs, on page 1](#)
- [Additional References, on page 9](#)

## Information About L2VPN Protocol-Based CLIs

### Overview of L2VPN Protocol-Based CLIs

The L2VPN Protocol-Based CLIs feature introduces new commands and modifies or replaces existing commands to achieve a consistent functionality across Cisco platforms and provide cross-Operating System (OS) support.



---

**Note** The new, updated, and replacement commands are available in Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S. However, the legacy commands that are being replaced will be deprecated in later releases.

---

### Benefits of L2VPN Protocol-Based CLIs

The L2VPN Protocol-Based CLIs feature provides the following benefits:

- Consistent user experience across different operating systems.
- Consistent configuration for all Layer 2 VPN (L2VPN) scenarios.
- Enhanced functionality that is achieved by configuring pseudowires as virtual interfaces and monitoring the pseudowires as physical ports.
- Feature configuration such as quality of service (QoS) service policies on individual pseudowires .
- Redundant pseudowire configuration that is independent of the primary pseudowire to provide enhanced high availability.

These benefits are achieved through the following enhancements:

- New service contexts can be created for point-to-point and multipoint Layer 2 services by using the new L2VPN cross connect and L2VPN virtual forwarding interface (VFI) contexts.
  - The L2VPN cross connect context is used for configuring point-to-point pseudowires, pseudowire stitching, and local switching (hair pinning). Ethernet interfaces and subinterfaces, Ethernet Flow Points (EFP), ATM interfaces and WAN interfaces (PPP,HDLC,Serial), and pseudowire interfaces can be defined as members of an L2VPN cross connect context.
  - The L2VPN VFI context instantiates Virtual Private LAN Services (VPLS) VFI for multipoint scenarios. Pseudowires can be defined as members of an L2VPN VFI context.
  - Bridge domains or VLANs are used for multipoint scenarios. EFPs, pseudowires, or VFIs can be configured as members of a bridge domain. Pseudowires can be configured as member of a VFI. The VFI can be configured as a member of a VLANbridge domains.
- New port contexts can be created (dynamically or manually) for pseudowires by using the pseudowire interface.
- Pseudowire customization can be achieved using interface templates and pseudowire interfaces that are applied to L2VPN context members. Pseudowire customizations include following features:
  - Encapsulation type
  - Control word
  - Maximum Transmission Unit (MTU)
  - Pseudowire signaling type
  - Tunnel selection
- Interworking and redundancy group service attributes can be configured under the L2VPN service context. The redundancy groups are configured independently from the primary pseudowire, which helps achieve zero traffic interruptions while adding, modifying, or deleting backup pseudowires.

## L2VPN Protocol-Based CLI Changes

The following commands are introduced in Cisco IOS XE Release 3.7S, Cisco IOS Release 15.3(1)S, and Cisco IOS Release 15.4(1)S:

- **debug l2vpn pseudowire**
- **l2vpn**
- **l2vpn pseudowire static-oam class**
- **monitor event-trace l2vpn**
- **show interface pseudowire**
- **show l2vpn service**
- **shutdown (MPLS)**
- **vc**

The following commands are modified in Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S:

- **auto-route-target**
- **bridge-domain parameterized vlan**
- **debug condition xconnect fib**
- **debug condition xconnect interface**
- **debug condition xconnect peer**
- **debug condition xconnect segment**
- **description**
- **encapsulation (MPLS)**
- **forward permit l2protocol all**
- **interworking**
- **l2vpn subscriber authorization group**
- **l2vpn xconnect context**
- **load-balance flow**
- **monitor event-trace ac**
- **monitor event-trace atom**
- **monitor event-trace l2tp**
- **monitor peer bfd**
- **mtu**
- **preferred-path**
- **remote circuit id**
- **rd (VPLS)**
- **route-target (VPLS)**
- **sequencing**
- **status**
- **status admin-down disconnect**
- **status control-plane route-watch**
- **status decoupled**
- **status peer topology dual-homed**
- **status protocol notification static**
- **status redundancy**
- **switching tlv**

- tlv
- tlv template
- vccv
- vccv bfd status signaling
- vccv bfd template
- vpls-id
- vpn id (MPLS)

The table below lists the legacy commands that will be replaced in future releases. From Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S both new and legacy commands will coexist until the legacy commands are deprecated in future releases.

**Table 1: Replacement Commands Introduced in Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S**

Legacy Command	Replacement Command Introduced in Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S
backup delay	redundancy delay (under l2vpn xconnect context)
bridge-domain (service instance)	member (bridge-domain)
clear mpls l2transport fsm state transition	clear l2vpn atom fsm state transition
clear mpls l2transport fsm event	clear l2vpn atom fsm event
clear xconnect	clear l2vpn service
connect (L2VPN local switching)	l2vpn xconnect context
debug acircuit	debug l2vpn acircuit
debug mpls l2transport checkpoint	debug l2vpn atom checkpoint
debug mpls l2transport event-trace	debug l2vpn atom event-trace
debug mpls l2transport fast-failure-detect	debug l2vpn atom fast-failure-detect
debug mpls l2transport signaling	debug l2vpn atom signaling
debug mpls l2transport static-oam	debug l2vpn atom static-oam
debug mpls l2transport vc subscriber	debug l2vpn atom vc
debug mpls l2transport vc	debug l2vpn atom vc
debug mpls l2transport vc vccv bfd event	debug l2vpn atom vc vccv
debug vfi	debug l2vpn vfi
debug vfi checkpoint	debug l2vpn vfi checkpoint

<b>Legacy Command</b>	<b>Replacement Command Introduced in Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S</b>
<b>debug xconnect</b>	<b>debug l2vpn xconnect</b>
<b>debug xconnect rib</b>	<b>debug l2vpn xconnect rib</b>
<b>description (L2VFI)</b>	<b>description (L2VPN)</b>
<b>l2 pseudowire routing</b>	<b>pseudowire routing</b>
<b>l2 router-id</b>	<b>router-id</b>
<b>l2 vfi</b>	<b>l2vpn vfi context</b>
<b>l2 subscriber</b>	<b>l2vpn subscriber</b>
<b>l2 vfi autodiscovery</b>	<b>autodiscovery</b>
<b>l2 vfi point-to-point</b>	<b>l2vpn xconnect context</b>
<b>local interface</b>	<b>pseudowire type</b>
<b>monitor event-trace st-pw-oam</b>	<b>monitor event-trace pwoam</b>
<b>mpls label</b>	<b>label (pseudowire)</b>
<b>mpls control-word</b>	<b>control-word (encapsulation mpls under l2vpn connect context)</b>
<b>neighbor (l2 vfi)</b>	<b>member (l2vpn vfi)</b>
<b>protocol</b>	<b>signaling protocol</b>
<b>pseudowire-static-oam class</b>	<b>l2vpn pseudowire static-oam class</b>
<b>pseudowire tlv template</b>	<b>l2vpn pseudowire tlv template</b>
<b>pw-class keyword in the xconnect command</b>	<b>source template type pseudowire</b>
<b>remote link failure notification</b>	<b>l2vpn remote link failure notification</b>
<b>show mpls l2transport binding</b>	<b>show l2vpn atom binding</b>
<b>show mpls l2transport checkpoint</b>	<b>show l2vpn atom checkpoint</b>
<b>show mpls l2transport hw-capability</b>	<b>show l2vpn atom hw-capability</b>
<b>show mpls l2transport static-oam</b>	<b>show l2vpn atom static-oam</b>
<b>show mpls l2transport summary</b>	<b>show l2vpn atom summary</b>
<b>show mpls l2transport pwid</b>	<b>show l2vpn atom pwid</b>
<b>show mpls l2transport vc</b>	<b>show l2vpn atom vc</b>

Legacy Command	Replacement Command Introduced in Cisco IOS XE Release 3.7S and Cisco IOS Release 15.3(1)S
show xconnect pwmib	show l2vpn pwmib
show xconnect rib	show l2vpn rib
show xconnect	show l2vpn service
show vfi	show l2vpn vfi
xconnect	l2vpn xconnect context and member
xconnect logging pseudowire status global	logging pseudowire status
xconnect logging redundancy global	logging redundancy
xconnect peer-ip vc-id	neighbor peer-ip vc-id (xconnect context)

## MPLS L2VPN Protocol-Based CLI: Examples

The examples in this section provide the new configurations that are introduced by the MPLS L2VPN Protocol-Based CLIs feature that replace the existing (legacy) MPLS L2VPN CLIs.

### MPLS L2VPN VPWS Configuration Using Replacement (or New) Commands

The following example shows the configuration for Virtual Private Wired Service (VPWS)—Ethernet over Multiprotocol Label Switching (EoMPLS). In this example, L2VPN members point to peer ID or virtual circuit (VC) ID. This configuration is used in most cases except when features like quality of service (QoS), need to be applied at the pseudowire level.

```
l2vpn xconnect context foo
  member GigabitEthernet2/1/1 service-instance 300
  member 10.0.0.1 888 encapsulation mpls
!
interface GigabitEthernet2/1/1
  service instance 300 GigabitEthernetEthernet
  encapsulation dot1q 30
  rewrite ingress tag pop 1 symmetric
!
  service instance 400 GigabitEthernetEthernet
  encapsulation dot1q 40
  rewrite ingress tag pop 1 symmetric

l2vpn xconnect context faa
  member GigabitEthernet2/1/1 service-instance 400
  member 10.0.0.1 999 encapsulation mpls
!
```

### MPLS L2VPN Pseudowire Configuration Using Replacement (or New) Commands

In the following example, L2VPN members point to a pseudowire interface. The pseudowire interface is manually configured and includes peer ID and VC ID. This configuration is used in most cases except when features like quality of service (QoS), need to be applied at the pseudowire level.

```
l2vpn xconnect context foo
  member GigabitEthernet2/1/1 service-instance 300
  member Pseudowire888
!
```

```

interface Pseudowire 888
  encapsulation mpls
  neighbor 10.0.0.1 888
!
interface Pseudowire 999
  encapsulation mpls
  neighbor 10.0.0.1 999
!
interface GigabitEthernet2/1/1
  service instance 300 GigabitEthernetEthernet
  encapsulation dot1q 30
  rewrite ingress tag pop 1 symmetric
!
  service instance 400 GigabitEthernetEthernet
  encapsulation dot1q 40
  rewrite ingress tag pop 1 symmetric

l2vpn xconnect context faa
  member GigabitEthernet2/1/1 service-instance 400
  member Pseudowire 999
!

```

### MPLS L2VPN Pseudowire Redundancy Configuration Using Replacement (or New) Commands

The following example shows the configuration for pseudowire redundancy. The new configuration shows concise pseudowire redundancy with no submodes or separate groups. This configuration allows the addition of redundant members to a service without service disruption. This configuration also allows modifying or deleting redundant service configurations without service disruption.

```

l2vpn xconnect context sample-pw-redundancy
  member Ethernet2/1GigabitEthernet2/1/1 service-instance 200
  member 10.1.1.1 180 encap mpls group Denver
  member 2.2.2.2 180180 encap mpls group Denver priority 1
  member 3.3.3.3 180181 encap mpls group Denver priority 2
  redundancy delay 1 20 group Denver
!
interface GigabitEthernet2/1/1
  service instance 200 GigabitEthernetEthernet
  encapsulation dot1q 100
  rewrite ingress tag pop 1 symmetric

```

### MPLS L2VPN Static Pseudowire Configuration Using Replacement (or New) Commands



**Note** The following configuration is shown for the Provider Edge (PE) 1 router in a network scheme where Customer Edge (CE) 1 and PE 1 and PE 2 and CE 2 traverse through a Provider core (P) router (CE 1—PE 1—P—PE 2—CE 2).

```

interface g2/1/1
  service instance 300 ethernet
  encapsulation dot1q 300
  no shutdown
!
interface pseudowire 100
  neighbor 10.4.4.4 121
  encapsulation mpls
  label 200 300
  signaling protocol none
  no shutdown
!
l2vpn xconnect context foo

```

```

member GigabitEthernet2/1/1 service-instance 300
member pseudowire 100

```

### MPLS L2VPN Static Pseudowire Template Configuration Using Replacement (or New) Commands



**Note** The following configuration is shown for the Provider Edge (PE) 1 router in a network scheme where Customer Edge (CE) 1 and PE 1 and PE 2 and CE 2 traverse through a Provider core (P) router (CE 1—PE 1—P—PE 2—CE 2).

```

template type pseudowire test
encapsulation mpls
signaling protocol none
!
interface g2/1/1
service instance 300 ethernet
encapsulation dot1q 300
no shutdown
!
interface pseudowire 100
neighbor 10.4.4.4 121
source template type pseudowire test
label 200 300
no shutdown
!
l2vpn xconnect context foo
member GigabitEthernet2/1/1 service-instance 300
member pseudowire 100

```

### MPLS L2VPN Dynamic Pseudowire Template Configuration Using Replacement (or New) Commands



**Note** The following configuration is shown for the Provider Edge (PE) 1 router in a network scheme where Customer Edge (CE) 1 and PE 1 and PE 2 and CE 2 traverse through a Provider core (P) router (CE 1—PE 1—P—PE 2—CE 2).

```

template type pseudowire test
encapsulation mpls
signaling protocol ldp
!
!
interface g2/1/1
service instance 300 ethernet
encapsulation dot1q 300
no shutdown
!
interface pseudowire 100
neighbor 10.4.4.4 121
source template type pseudowire test
no shutdown
!
l2vpn xconnect context foo
member GigabitEthernet2/1/1 service-instance 300
member pseudowire 100

```

### MPLS L2VPN Multi-segment Static-Dynamic Pseudowire Template Configuration Using Replacement (or New) Commands

The following PE router configuration is for a multi-segment static-dynamic pseudowire:



```

l2vpn pseudowire tlv template TLV
  tlv mtu 1 4 dec 1500
!
interface pseudowire401
  source template type pseudowire staticTempl
encapsulation mpls
neighbor 10.4.4.4 101
signaling protocol none
label 4401 4301
pseudowire type 4
  tlv template TLV
  tlv 1 4 dec 1500
  tlv vccv-flags C 4 hexstr 0110
!
interface pseudowire501
  source template type pseudowire dynTempl
encapsulation mpls
neighbor 10.2.2.2 101
signaling protocol ldp

```

### Displaying MPLS L2VPN Pseudowire Template Configuration Using Replacement (or New) Commands

The following example displays output from the **show interface pseudowire** command:

```

PE1#show interface pseudowire 100
pseudowire100 is up
  Description: Pseudowire Interface
  MTU 1500 bytes, BW 10000000 Kbit
  Encapsulation mpls
  Peer IP 10.4.4.4, VC ID 121
  RX
    21 packets 2623 bytes 0 drops
  TX
    20 packets 2746 bytes 0 drops

```

The following example displays output from the **show template** command:

```

PE1#show template

Template      class/type      Component(s)
ABC           owner           interface pseudowire
  BOUND: pw1

```

### Sourcing a Template Under an Interface Pseudowire Using Replacement (or New) Commands

The following example configures the interface pseudowire to inherit all attributes defined from a template on the PE 2 router.

```

PE2(config-subif)#interface pseudowire 100
PE2(config-if)#source template type pseudowire test
PE2(config-if)#neighbor 10.4.4.4 121
PE2(config-if)#no shutdown

```

## Additional References

### Related Documents

Related Topic	Document Title
Cisco IOS commands	<a href="#">Cisco IOS Master Command List, All Releases</a>

Related Topic	Document Title
MPLS commands	<a href="#">Multiprotocol Label Switching Command Reference</a>

**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.	<a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a>