

# Network Configuration Example

## Interconnecting a Layer 2 Circuit with a Layer 2 VPN

Release  
**11.3**



---

Published: 2011-09-22

Juniper Networks, Inc.  
1194 North Mathilda Avenue  
Sunnyvale, California 94089  
USA  
408-745-2000  
www.juniper.net

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

*Network Configuration Example Interconnecting a Layer 2 Circuit with a Layer 2 VPN*

Release 11.3

Copyright © 2011, Juniper Networks, Inc.

All rights reserved.

Revision History

September 2011—R1 Junos OS 11.3

The information in this document is current as of the date listed in the revision history.

#### YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. The Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

## END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.



# Table of Contents

Layer 2 Circuit Overview .....	1
Layer 2 VPN Overview .....	3
Layer 2 VPN Applications .....	5
Using the Layer 2 Interworking Interface to Interconnect a Layer 2 Circuit to a Layer 2 VPN .....	7
Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN .....	9



## Layer 2 Circuit Overview

---

A Layer 2 circuit is a point-to-point Layer 2 connection transported using Multiprotocol Label Switching (MPLS) or other tunneling technology on the service provider's network. A Layer 2 circuit is similar to a circuit cross-connect (CCC), except that multiple virtual circuits (VCs) are transported over a single shared label-switched path (LSP) tunnel between two provider edge (PE) routers. In contrast, each CCC requires a separate dedicated LSP.

To establish a Layer 2 circuit, the Link Integrity Protocol (LIP) is used as the signaling protocol to advertise the ingress label to the remote PE routers. For this purpose, a targeted remote LDP neighbor session is established using the extended discovery mechanism described in LDP, and the session is brought up to the remote PE loopback IP address. Because LDP looks at the Layer 2 circuit configuration and initiates extended neighbor discovery for all the Layer 2 circuit neighbors (the remote PEs), no new configuration is necessary in LDP. Each Layer 2 circuit is represented by the logical interface connecting the local PE router to the local customer edge (CE) router. Note that LDP must be enabled on the lo0.0 interface for extended neighbor discovery to function correctly.

Packets are sent to remote CE routers over an egress VPN label advertised by the remote PE router, using a targeted LDP session. The VPN label is sent over an LDP LSP to the remote PE router connected to the remote CE router. Return traffic from the remote CE router destined to the local CE router is sent using an ingress VPN label advertised by the local PE router, which is also sent over the LDP LSP to the local PE router from the remote PE router.

### Related Documentation

- [Layer 3 VPN Overview](#)
- [Layer 2 VPN Overview on page 3](#)
- [Layer 2 VPN Applications on page 5](#)
- [Applications for Interconnecting a Layer 2 Circuit with a Layer 2 Circuit](#)
- [Applications for Interconnecting a Layer 2 Circuit with a Layer 3 VPN](#)
- [Example: Interconnecting a Layer 2 Circuit with a Layer 2 Circuit](#)
- [Example: Interconnecting a Layer 2 Circuit with a Layer 3 VPN](#)
- [Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN on page 9](#)





## Layer 2 VPN Overview

---

As the need to link different Layer 2 services to one another for expanded service offerings grows, Layer 2 MPLS VPN services are increasingly in demand.

Implementing a Layer 2 VPN on a router is similar to implementing a VPN using a Layer 2 technology, such as Asynchronous Transfer Mode (ATM). However, for a Layer 2 VPN on a router, traffic is forwarded to the router in a Layer 2 format. It is carried by Multiprotocol Label Switching (MPLS) over the service provider's network, and then converted back to Layer 2 format at the receiving site. You can configure different Layer 2 formats at the sending and receiving sites. The security and privacy of an MPLS Layer 2 VPN are equal to those of an ATM or Frame Relay VPN. The service provisioned with Layer 2 VPNs is also known as Virtual Private Wire Service (VPWS).

On a Layer 2 VPN, routing typically occurs on the customer edge (CE) router. The CE router connected to a service provider on a Layer 2 VPN must select the appropriate circuit on which to send traffic. The provider edge (PE) router receiving the traffic sends the traffic across the service provider's network to the PE router connected to the receiving site. The PE routers do not need to store or process the customer's routes; they only need to be configured to send data to the appropriate tunnel. For a Layer 2 VPN, customers need to configure their own routers to carry all Layer 3 traffic. The service provider needs to know only how much traffic the Layer 2 VPN will need to carry. The service provider's routers carry traffic between the customer's sites using Layer 2 VPN interfaces. The VPN topology is determined by policies configured on the PE routers.

Because Layer 2 VPNs use BGP as the signaling protocol, they have a simpler design and require less overhead than traditional VPNs over Layer 2 circuits. BGP signaling also enables autodiscovery of Layer 2 VPN peers. Layer 2 VPNs are similar to BGP or MPLS VPNs and VPLS in many respects; all three types of services employ BGP for signaling.

### Related Documentation

- [Layer 2 VPN Applications on page 5](#)
- [Using the Layer 2 Interworking Interface to Interconnect a Layer 2 Circuit to a Layer 2 VPN on page 7](#)
- [Using the Layer 2 Interworking Interface to Interconnect a Layer 2 VPN to a Layer 2 VPN](#)
- [Interconnecting Layer 2 VPNs with Layer 3 VPNs Overview](#)
- [Example: Interconnecting a Layer 2 VPN with a Layer 2 VPN](#)
- [Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN on page 9](#)
- [Example: Interconnecting a Layer 2 VPN with a Layer 3 VPN](#)



## Layer 2 VPN Applications

---

Implementing a Layer 2 VPN includes the following benefits:

- Terminating a Layer 2 VPN into a Layer 2 VPN using the interworking (iw0) software interface eliminates the limitation of bandwidth on the tunnel interfaces used for these configuration scenarios. Instead of using a physical Tunnel PIC for looping the packet received from the Layer 2 VPN to another Layer 2 VPN, Junos OS is used to link both the Layer 2 VPN routes.
- Layer 2 VPNs enable the sharing of a provider's core network infrastructure between IP and Layer 2 VPN services, reducing the cost of providing those services. A Layer 2 MPLS VPN allows you to provide Layer 2 VPN service over an existing IP and MPLS backbone.
- From a service provider's point of view, a Layer 2 MPLS VPN allows the use of a single Layer 3 VPN (such as RFC 2547bis), MPLS traffic engineering, and Differentiated Services (DiffServ).
- Service providers do not have to invest in separate Layer 2 equipment to provide Layer 2 VPN service. You can configure the PE router to run any Layer 3 protocol in addition to the Layer 2 protocols. Customers who prefer to maintain control over most of the administration of their own networks might want Layer 2 VPN connections with their service provider instead of a Layer 3 VPN.

### Related Documentation

- [Layer 2 VPN Overview on page 3](#)
- [Using the Layer 2 Interworking Interface to Interconnect a Layer 2 Circuit to a Layer 2 VPN on page 7](#)
- [Using the Layer 2 Interworking Interface to Interconnect a Layer 2 VPN to a Layer 2 VPN](#)
- [Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN on page 9](#)
- [Example: Interconnecting a Layer 2 VPN with a Layer 2 VPN](#)
- [Example: Interconnecting a Layer 2 VPN with a Layer 3 VPN](#)



## Using the Layer 2 Interworking Interface to Interconnect a Layer 2 Circuit to a Layer 2 VPN

---

Instead of using a physical Tunnel PIC for looping the packet received from the Layer 2 circuit, the Layer 2 interworking interface uses Junos OS to stitch together both Layer 2 VPN routes.

To configure the interworking interface, include the **iw0** statement. The **iw0** statement is configured at the **[edit interfaces]** hierarchy level. This specifies the peering between two logical interfaces. This configuration is similar to the configuration for a logical tunnel interface. The logical Interfaces must be associated with the endpoints of a Layer 2 circuit and Layer 2 VPN connections.

```
[edit interfaces]
iw0 {
  unit 0 {
    peer-unit 1;
  }
  unit 1 {
    peer-unit 0;
  }
}
```

Configure the Layer 2 circuit protocol by including the **l2circuit** statement at the **[edit protocols]** hierarchy level and specifying the **neighbor** and **iw0** interface.

```
[edit protocols]
l2circuit {
  neighbor 1.2.3.4 {
    interface iw0.0;
  }
}
```

Configure the Layer 2 VPN connection, by including the **routing-instance-name** statement at the **[edit routing-instances]** hierarchy level and specifying the **instance-type l2vpn** option.

```
[edit routing-instances]
routing-instance-name {
  instance-type l2vpn;
  interface iw0.1;
  ...
  protocols {
    l2vpn {
      <l2vpn configuration>;
    }
  }
}
```

In addition to the **iw0** interface configuration, Layer 2 interworking **l2iw** protocols must be enabled. Without the **l2iw** configuration, the **l2iw** routes will not be formed, regardless of whether any **iw** interfaces are present. Within the **l2iw** protocols, only trace options can be configured in the standard fashion. The minimum configuration necessary for the feature to work is shown below:

```
[edit]
```

```
protocols {  
  l2iw;  
}
```

**Related  
Documentation**

- [Layer 2 Circuit Overview on page 1](#)
- [Layer 2 VPN Overview on page 3](#)
- [Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN on page 9](#)

## Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN

This example provides a step-by-step procedure and commands for configuring and verifying a Layer 2 circuit to a Layer 2 VPN. It contains the following sections:

- [Requirements on page 9](#)
- [Overview and Topology on page 9](#)
- [Configuration on page 10](#)

### Requirements

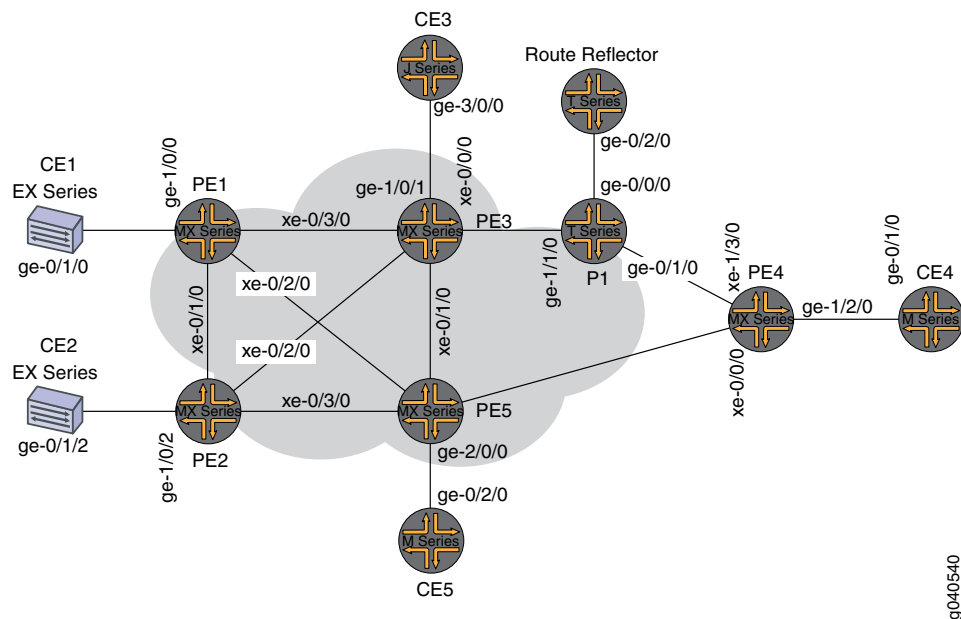
This example uses the following hardware and software components:

- Junos OS Release 9.3 or later
- 2 MX Series routers
- 2 M Series routers
- 1 T Series router
- 1 EX Series router
- 1 J Series router

### Overview and Topology

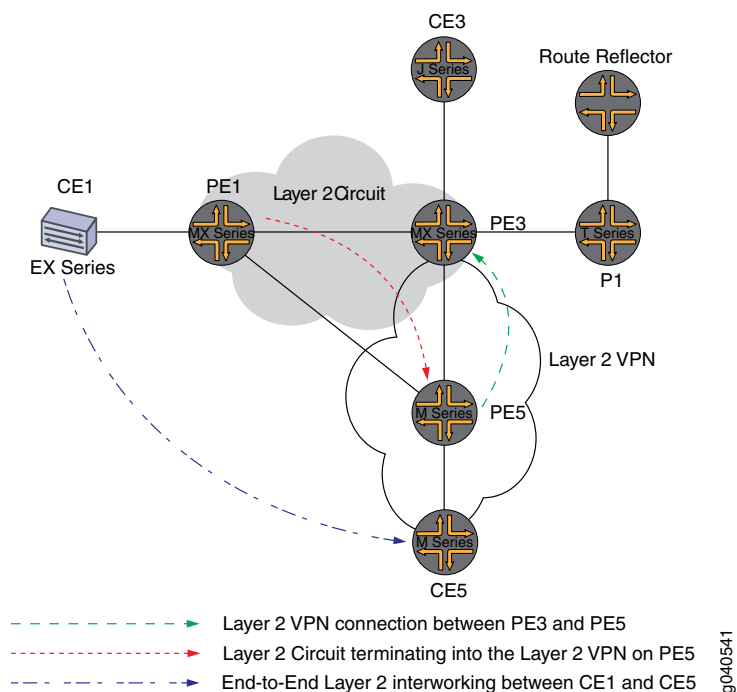
The physical topology of a Layer 2 circuit to a Layer 2 VPN connection is shown in [Figure 1 on page 9](#).

**Figure 1: Physical Topology of a Layer 2 Circuit to a Layer 2 VPN Connection**



The logical topology of a Layer 2 circuit to a Layer 2 VPN connection is shown in [Figure 2 on page 10](#).

**Figure 2: Logical Topology of a Layer 2 Circuit to a Layer 2 VPN Connection**



## Configuration



**NOTE:** In any configuration session, it is good practice to verify periodically that the configuration can be committed using the `commit check` command.

In this example, the router being configured is identified using the following command prompts:

- **CE1** identifies the customer edge 1 (CE1) router
- **PE1** identifies the provider edge 1 (PE1) router
- **CE3** identifies the customer edge 3 (CE3) router
- **PE3** identifies the provider edge 3 (PE3) router
- **CE5** identifies the customer edge 5 (CE5) router
- **PE5** identifies the provider edge 5 (PE5) router

This example is organized in the following sections:

- [Configuring Protocols on the PE and P Routers on page 11](#)
- [Verification on page 15](#)



## Configuring Protocols on the PE and P Routers

### Step-by-Step Procedure

#### Base Configuration

All of the PE routers and P routers are configured with OSPF as the IGP protocol. The MPLS, LDP, and BGP protocols are enabled on all of the interfaces except **fxp0.0**. Core-facing interfaces are enabled with the MPLS address and inet address.

1. Configure all the PE and P routers with OSPF as the IGP. Enable the MPLS, LDP, and BGP protocols on all interfaces except **fxp0.0**. LDP is used as the signaling protocol on Router PE1 for the Layer 2 circuit. The following configuration snippet shows the protocol configuration for Router PE1:

```
[edit]
protocols {
  mpls {
    interface all;
    interface fxp0.0 {
      disable;
    }
  }
  bgp {
    group RR {
      type internal;
      local-address 1.1.1.1;
      family l2vpn {
        signaling;
      }
      neighbor 7.7.7.7;
    }
  }
  ospf {
    traffic-engineering;
    area 0.0.0.0 {
      interface all;
      interface fxp0.0 {
        disable;
      }
    }
  }
  ldp {
    interface all;
    interface fxp0.0 {
      disable;
    }
  }
}
```

2. Configure the PE and P routers with OSPF as the IGP. Enable the MPLS, LDP, and BGP protocols on all interfaces except **fxp0.0**. BGP is used as the signaling protocol on Router PE3 for the Layer 2 VPN. The following configuration snippet shows the protocol configuration for Router PE3:

```
[edit]
protocols {
```

```
mpls {
  interface all;
  interface fxp0.0 {
    disable;
  }
}
bgp {
  group RR {
    type internal;
    local-address 3.3.3.3;
    family l2vpn {
      signaling;
    }
    neighbor 7.7.7.7;
  }
}
ospf {
  traffic-engineering;
  area 0.0.0.0 {
    interface all;
    interface fxp0.0 {
      disable;
    }
  }
}
ldp {
  interface all;
  interface fxp0.0 {
    disable;
  }
}
```

**Step-by-Step  
Procedure****Configuring Interfaces**

1. On Router PE1, configure the **ge-1/0/0** interface encapsulation. To configure the interface encapsulation, include the **encapsulation** statement and specify the **ethernet-ccc** option (vlan-ccc encapsulation is also supported). Configure the **ge-1/0/0.0** logical interface family for circuit cross-connect functionality. To configure the logical interface family, include the **family** statement and specify the **ccc** option. The encapsulation should be configured the same way for all routers in the Layer 2 circuit domain.

```
[edit interfaces]
ge-1/0/0 {
  encapsulation ethernet-ccc;
  unit 0 {
    family ccc;
  }
}
lo0 {
  unit 0 {
    family inet {
      address 1.1.1.1/32;
    }
  }
}
```

```
}
}
```

- Router PE5 is the router that is *stitching* the Layer 2 circuit to the Layer 2 VPN using the interworking interface. The configuration of the peer unit interfaces is what makes the interconnection.

On Router PE5, configure the **iw0** interface with two logical interfaces. To configure the **iw0** interface, include the **interfaces** statement and specify **iw0** as the interface name. For the unit 0 logical interface, include the **peer-unit** statement and specify the logical interface **unit 1** as the peer l interface. For the unit 1 logical interface, include the **peer-unit** statement and specify the logical interface **unit 0** as the peer interface.

```
[edit interfaces]
iw0 {
  unit 0 {
    encapsulation ethernet-ccc;
    peer-unit 1;
  }
  unit 1 {
    encapsulation ethernet-ccc;
    peer-unit 0;
  }
}
```

- On Router PE5, configure the logical loopback interface. The loopback interface is used to establish the targeted LDP sessions to Routers PE1 and PE5.

```
[edit interfaces]
lo0 {
  unit 0 {
    family inet {
      address 5.5.5.5/32;
    }
  }
}
```

### Step-by-Step Procedure

#### Configuring the Layer 2 circuit protocol

- On Router PE1, configure the IP address of the remote PE router with the **neighbor** statement. The loopback address and router ID of the PE neighbor is commonly the neighbor's IP address. To allow a Layer 2 circuit to be established even though the maximum transmission unit (MTU) configured on the PE router does not match the MTU configured on the remote PE router, include the **ignore-mtu-mismatch** statement.

```
[edit]
protocols {
  l2circuit {
    neighbor 5.5.5.5 {
      interface ge-1/0/0.0 {
        virtual-circuit-id 100;
        no-control-word;
        ignore-mtu-mismatch;
      }
    }
  }
}
```

```
    }
  }
}
```

2. On Router PE5, configure the IP address of the remote PE router. To configure the IP address of the remote PE router, include the **neighbor** statement and specify the IP address of the loopback interface on Router PE1. Configure the virtual circuit ID to be the same as the virtual circuit ID on the neighbor router. To allow a Layer 2 circuit to be established even though the MTU configured on the local PE router does not match the MTU configured on the remote PE router, include the **ignore-mtu-mismatch** statement. Also disable the use of the control word for demultiplexing by including the **no-control-word** statement.

```
[edit protocols]
l2circuit {
  neighbor 1.1.1.1 {
    interface iw0.0 {
      virtual-circuit-id 100;
      no-control-word;
      ignore-mtu-mismatch;
    }
  }
}
```

3. On Router PE5, configure the Layer 2 VPN protocols by including the **l2vpn** statement at the **[edit routing-instances routing-instances-name protocols]** hierarchy level. To configure the **iw0** interface, include the **interfaces** statement and specify **iw0** as the interface name. The **iw0** interface is configured under the Layer 2 VPN protocols to receive the looped packet from the **iw0.1**. Protocols **l2vpn** is configured on Router PE5 with a site CE5, which is configured in the BGP L2VPN. The CE1 should have communication to CE5, through the Layer 2 interworking configuration on Router PE5.

```
[edit]
routing-instances {
  L2VPN {
    instance-type l2vpn;
    interface ge-2/0/0.0;
    interface iw0.1;
    route-distinguisher 65000:5;
    vrf-target target:65000:2;
    protocols {
      l2vpn {
        encapsulation-type ethernet;
        site CE5 {
          site-identifier 5;
          interface ge-2/0/0.0 {
            remote-site-id 3;
          }
        }
        site l2-circuit {
          site-identifier 6;
          interface iw0.1 {
            remote-site-id 3;
          }
        }
      }
    }
  }
}
```

```

    }
  }
}

```

4. In addition to the **iw0** interface configuration, Layer 2 interworking **l2iw** protocol must be configured. Without the **l2iw** configuration, the **l2iw** routes are not formed, regardless of whether any **iw** interfaces are present.

On Router PE5, configure the **l2iw** protocol. To configure the protocol, include the **l2iw** statement at the **[edit protocols]** hierarchy level.

```

[edit]
protocols {
  l2iw;
}

```

## Verification

### Step-by-Step Procedure Verifying the Layer 2 Circuit Connection on Router PE1

1. On Router PE1, use the **show l2circuit connections** command to verify that the Layer 2 Circuit from Router PE1 to Router PE5 is **Up**.

```
user@PE1> show l2circuit connections
```

Layer-2 Circuit Connections:  
Legend for connection status (St)

EI -- encapsulation invalid	NP -- interface h/w not present
MM -- mtu mismatch	Dn -- down
EM -- encapsulation mismatch	VC-Dn -- Virtual circuit Down
CM -- control-word mismatch	Up -- operational
VM -- vlan id mismatch	CF -- Call admission control failure
OL -- no outgoing label	IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC	TM -- TDM misconfiguration
BK -- Backup Connection	ST -- Standby Connection
CB -- rcvd cell-bundle size bad	XX -- unknown
SP -- Static Pseudowire	

Legend for interface status

Up -- operational

Dn -- down

Neighbor: 5.5.5.5

Interface	Type	St	Time last up	# Up trans
ge-1/0/0.0(vc 100)	rmt	Up	Jan 3 22:00:49 2010	1

Remote PE: 5.5.5.5, Negotiated control-word: No  
Incoming label: 301328, Outgoing label: 300192  
Local interface: ge-1/0/0.0, Status: Up, Encapsulation: ETHERNET

2. On Router PE5, use the **show l2vpn connections** command to verify that the Layer 2 VPN connection is **Up** using the **iw0** peer interface of the Layer 2 circuit.

```
user@PE5> show l2vpn connections
```

Instance: L2VPN

Local site: CE5 (5)

connection-site	Type	St	Time last up	# Up trans
l2-circuit (6)	loc	OR		

```

3                               rmt    Up
Jan  3 22:51:12 2010           1
Remote PE: 3.3.3.3, Negotiated control-word: Yes (Null)
Incoming label: 800258, Outgoing label: 800000
Local interface: ge-2/0/0.0, Status: Up, Encapsulation: ETHERNET
Local site: l2-circuit (6)
connection-site                Type St      Time last up    # Up trans
CE5 (5)                        loc   OR
3                               rmt    Up      Jan  3 22:56:38 2010    1
Remote PE: 3.3.3.3, Negotiated control-word: Yes (Null)
Incoming label: 800262, Outgoing label: 800001
Local interface: iwo.1, Status: Up, Encapsulation: ETHERNET

```

### Step-by-Step Procedure

Verifying the Layer 2 Circuit terminating into Layer 2 VPN connection

1. On Router PE 5, use the **show l2circuit connections** command to verify that the Layer 2 circuit is **Up** using the **iwo** interface. This will be looped through the **iwo.1** interface to the Layer 2 VPN.

```
user@PE5> show l2circuit connections
```

Layer-2 Circuit Connections:

```

Neighbor: 1.1.1.1
Interface                Type St      Time last up    # Up trans
iwo.0(vc 100)            rmt  Up      Jan  3 21:59:07 2010    1
Remote PE: 1.1.1.1, Negotiated control-word: No
Incoming label: 300192, Outgoing label: 301328

```

2. On Router PE 5, use the **show route table mpls.0** command to verify the Layer 2 circuit and Layer 2 VPN routes. In the example below, the Layer 2 circuit is associated with LDP label **301328** and the Layer 2 VPN is associated with LDP label **800001**. Notice the two **iwo** interfaces that are used for the Layer 2 interworking route.

```
user@PE5>show route table mpls.0
```

```

mpls.0: 18 destinations, 20 routes (18 active, 2 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

0                *[MPLS/0] 5d 20:07:31, metric 1
                  Receive
1                *[MPLS/0] 5d 20:07:31, metric 1
                  Receive
2                *[MPLS/0] 5d 20:07:31, metric 1
                  Receive
299776           *[LDP/9] 2d 03:00:51, metric 1
300048           *[LDP/9] 2d 03:00:49, metric 1
                  > to 10.10.6.1 via xe-0/1/0.0, Pop
300048(S=0)      *[LDP/9] 2d 03:00:49, metric 1
                  > to 10.10.6.1 via xe-0/1/0.0, Pop
300192           *[L2IW/6] 19:11:05, metric2 1
                  > to 10.10.6.1 via xe-0/1/0.0, Swap 800001
                  [L2CKT/7] 20:08:36
                  > via iwo.0, Pop
800258           *[L2VPN/7] 19:16:31
                  > via ge-2/0/0.0, Pop          Offset: 4
800262          *[L2IW/6] 19:11:05, metric2 1

```

```
> to 10.10.3.1 via xe-1/1/0.0, Swap 301328
[L2VPN/7] 19:11:05
> via iw0.1, Pop Offset: 4
ge-2/0/0.0      *[L2VPN/7] 19:16:31, metric2 1
                 > to 10.10.6.1 via xe-0/1/0.0, Push 800000 Offset: -4
iw0.0           *[L2CKT/7] 20:08:36, metric2 1
                 > to 10.10.3.1 via xe-1/1/0.0, Push 301328
iw0.1           *[L2VPN/7] 19:11:05, metric2 1
                 > to 10.10.6.1 via xe-0/1/0.0, Push 800001 Offset: -4
```

- Related Documentation**
- [Layer 2 Circuit Overview on page 1](#)
  - [Layer 2 VPN Overview on page 3](#)
  - [Layer 2 VPN Applications on page 5](#)
  - [Using the Layer 2 Interworking Interface to Interconnect a Layer 2 Circuit to a Layer 2 VPN on page 7](#)

