

## Example: Configuring Inter-AS VPLS with MAC Processing at the ASBR

---

This example describes how to configure inter-AS Virtual Private LAN Service (VPLS) with MAC processing between BGP-signaled VPLS and LDP-signaled VPLS. This feature is described in RFC 4761 as multi-AS VPLS option E or method E.

This example is organized in the following sections:

- Requirements on page 1
- Overview and Topology on page 1
- Configuration on page 2
- Verification on page 26

### Requirements

To support inter-AS VPLS between BGP-signaled VPLS and LDP-signaled VPLS, your network must meet the following hardware and software requirements:

- MX Series or M320 routers for the ASBRs.
- JUNOS Software Release 9.3 or higher.
- Gigabit Ethernet or 10-Gigabit Ethernet interfaces.

### Overview and Topology

VPLS is a key enabler for delivering multipoint Ethernet service. Major service providers have implemented IP and MPLS backbones and offer VPLS services to large enterprises. Growing demand requires the VPLS network to scale to support many VPLS customers with multiple sites spread across geographically dispersed regions. BGP-signaled VPLS signaling offers scaling advantages over LDP-signaled VPLS. In some environments there is a need for BGP-signaled VPLS to interoperate with existing LDP-signaled VPLS.

This example shows one way to configure BGP-signaled VPLS interworking with an existing LDP-signaled VPLS network.

The advantages of the configuration are:

- You can interconnect customer sites that are spread across different autonomous systems (ASs).
- LDP-signaled VPLS and BGP-signaled VPLS interworking is supported.
- Because the ASBR supports MAC operations, customer sites can be connected directly to the ASBR.
- The inter-AS link is not restricted to Ethernet interfaces.
- Additional configuration for multihoming is relatively straightforward.

Traffic from the interworking virtual private LAN services is switched at the ASBR. The ASBR does all the data plane operations: flooding, MAC learning, aging, and MAC

forwarding for each AS to switch traffic among any customer facing interfaces and between the fully meshed pseudowires in the AS. A single pseudowire is created between the ASBRs across the inter-AS link and the ASBRs forward traffic from the pseudowires in each AS to the peer ASBR.

Each ASBR performs VPLS operations within its own AS and performs VPLS operations with the ASBR in the other AS. The ASBR treats the other AS as a BGP-signaled VPLS site. To establish VPLS pseudowires, VPLS NLRI messages are exchanged across the EBGP sessions on the inter-AS links between the ASBRs.

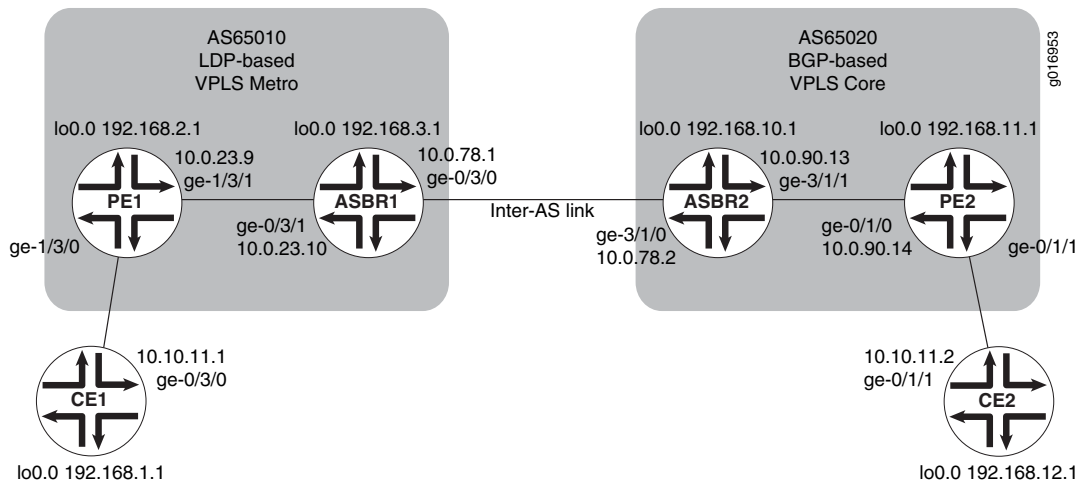
The sample metro network is configured for LDP-signaled VPLS. The core network is configured for BGP-signaled VPLS.

The first part of the example shows the basic configuration steps to configure the logical interfaces, OSPF, internal BGP, LDP, and MPLS. This part of the configuration is the same as other VPLS configurations for LDP-signaled VPLS and BGP-signaled VPLS.

The unique part of the example is configured in the VPLS routing instances, external BGP, and the policy that populates the BGP route table with routes learned from direct routes and OSPF routes. Additional details about the configuration statements are included in the step-by-step procedure.

Figure 1 shows the topology used in this example.

**Figure 1: Inter-AS VPLS with MAC Operations Example Topology**



## Configuration

To configure inter-AS VPLS between BGP-signaled VPLS and LDP-signaled VPLS, perform these tasks.



---

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

---

- Configuring Interfaces on page 3
- Configuring OSPF on page 5
- Configuring the Internal BGP Peer Group on page 6
- Configuring LDP on page 8
- Configuring MPLS on page 9
- Configuring the External BGP Peer Group Between the Loopback Interfaces on page 10
- Configuring the External BGP Peer Group Between the Inter-AS Link Interfaces on page 11
- Configuring the VPLS Routing Instances on page 15

## Configuring Interfaces

**Step-by-Step Procedure** To configure interfaces:

1. On each router, configure an IP address on the loopback logical interface 0 (lo0.0):

```
user@CE1# set interfaces lo0 unit 0 family inet address 192.168.1.1/32
primary
```

```
user@PE1# set interfaces lo0 unit 0 family inet address 192.168.2.1/32
primary
```

```
user@ASBR1# set interfaces lo0 unit 0 family inet address 192.168.3.1/32
primary
```

```
user@ASBR2# set interfaces lo0 unit 0 family inet address 192.168.10.1/32
primary
```

```
user@PE2# set interfaces lo0 unit 0 family inet address 192.168.11.1/32
primary
```

```
user@CE2# set interfaces lo0 unit 0 family inet address 192.168.12.1/32
primary
```

2. On each router, commit the configuration:

```
user@host> commit check
```

```
configuration check succeeds
```

```
user@host> commit
```

```
commit complete
```

3. On each router, display the interface information for lo0 and verify that the correct IP address is configured:

```
user@host> show interfaces lo0
```

```
Physical interface: lo0, Enabled, Physical link is Up
Interface index: 6, SNMP ifIndex: 6
Type: Loopback, MTU: Unlimited
Device flags   : Present Running Loopback
Interface flags: SNMP-Traps
Link flags     : None
Last flapped   : Never
  Input packets : 0
  Output packets: 0
```

```
Logical interface lo0.0 (Index 75) (SNMP ifIndex 16)
Flags: SNMP-Traps Encapsulation: Unspecified
```

```
Input packets : 0
Output packets: 0
Protocol inet, MTU: Unlimited
  Flags: None
```

```
Addresses
```

```
Local: 127.0.0.1
```

```
Addresses, Flags: Primary Is-Default Is-Primary
```

```
Local: 192.168.3.1
```

```
Logical interface lo0.16384 (Index 64) (SNMP ifIndex 21)
```

```
Flags: SNMP-Traps Encapsulation: Unspecified
```

```
Input packets : 0
```

```
Output packets: 0
```

```
Protocol inet, MTU: Unlimited
```

```
Flags: None
```

```
Addresses
```

```
Local: 127.0.0.1
```

```
Logical interface lo0.16385 (Index 65) (SNMP ifIndex 22)
```

```
Flags: SNMP-Traps Encapsulation: Unspecified
```

```
Input packets : 0
```

```
Output packets: 0
```

```
Protocol inet, MTU: Unlimited
```

```
Flags: None
```

In the example above notice that the primary lo0 local address for the inet protocol family on ASBR1 is **192:168:3:1**.

4. On each router, configure an IP address and protocol family on the Gigabit Ethernet interfaces. Specify the inet protocol family.

```
user@CE1# set interfaces ge-0/3/0 unit 0 family inet address 10.10.11.1/24
```

```
user@PE1# set interfaces ge-1/3/1 unit 0 family inet address 10.0.23.9/30
```

```
user@ASBR1# set interfaces ge-0/3/1 unit 0 family inet address
10.0.23.10/30
```

```
user@ASBR1# set interfaces ge-0/3/0 unit 0 family inet address 10.0.78.1/30
```

```
user@ASBR2# set interfaces ge-3/1/0 unit 0 family inet address 10.0.78.2/30
```

```
user@ASBR2# set interfaces ge-3/1/1 unit 0 family inet address
10.0.90.13/30
```

```
user@PE2# set interfaces ge-0/1/0 unit 0 family inet address 10.0.90.14/30
```

```
user@CE2# set interfaces ge-0/1/1 unit 0 family inet address 10.10.11.2/24
```

5. On each router, commit the configuration:

```
user@host> commit check
```

```
configuration check succeeds
```

```
user@host> commit
```

```
commit complete
```

6. Display information for Gigabit Ethernet interfaces and verify that the IP address and protocol family are configured correctly.

```
user@ASBR2> show interfaces ge-* terse
```

Interface	Admin	Link	Proto	Local	Remote
ge-3/1/0	up	up			
ge-3/1/0.0	up	up	inet	10.0.78.2/30	
			multiservice		
ge-3/1/1	up	up			
ge-3/1/1.0	up	up	inet	10.0.90.13/30	
			multiservice		
ge-3/1/2	up	down			
ge-3/1/3	up	down			

## Configuring OSPF

**Step-by-Step Procedure** To configure OSPF:

1. On the PE and ASBR routers, configure the provider instance of OSPF. Configure OSPF traffic engineering support. Specify area 0.0.0.1 in the LDP-signaled VPLS network and area 0.0.0.0 in the BGP-signaled network. Specify the Gigabit Ethernet logical interfaces between the PE and ASBR routers. Specify lo0.0 as a passive interface.

```
user@PE1# set protocols ospf traffic-engineering
```

```
user@PE1# set protocols ospf area 0.0.0.1 interface ge-1/3/1.0
```

```
user@PE1# set protocols ospf area 0.0.0.1 interface lo0.0 passive
```

```
user@ASBR1# set protocols ospf traffic-engineering
user@ASBR1# set protocols ospf area 0.0.0.1 interface ge-0/3/1.0
user@ASBR1# set protocols ospf area 0.0.0.1 interface lo0.0 passive
```

```
user@ASBR2# set protocols ospf traffic-engineering
user@ASBR2# set protocols ospf area 0.0.0.0 interface ge-3/1/1.0
user@ASBR2# set protocols ospf area 0.0.0.0 interface lo0.0 passive
```

```
user@PE2# set protocols ospf traffic-engineering
user@PE2# set protocols ospf area 0.0.0.0 interface ge-0/1/0.0
user@PE2# set protocols ospf area 0.0.0.0 interface lo0.0 passive
```

2. On each router, commit the configuration:

```
user@host> commit check

configuration check succeeds

user@host> commit

commit complete
```

3. Display OSPF neighbor information and verify that the PE routers form adjacencies with the ASBR router in the same area. Verify that the neighbor state is Full.

```
user@host> show ospf neighbor

Address          Interface      State   ID           Pri  Dead
10.0.23.10      ge-1/3/1.0    Full   192.168.3.1  128  31
```

## Configuring the Internal BGP Peer Group

### Step-by-Step Procedure

The purpose of configuring an internal BGP peer group is to create a full mesh of BGP LSPs among the PE routers in the BGP-signaled AS, including the ASBRs.

To configure the internal BGP peer group:

1. The purpose of this step is to create a full mesh of IBGP peers between the PE routers, including the ASBRs, within the BGP-signaled AS.

On ASBR2, configure internal BGP. Specify the BGP type as `internal`. Specify the local address as the local `lo0` IP address.

Specify the `inet` protocol family. Specify the `labeled-unicast` statement and the `resolve-vpn` option. The `labeled-unicast` statement causes the router to advertise labeled routes out of the IPv4 `inet.0` route table and places labeled routes into the `inet.0` route table. The `resolve-vpn` option puts labeled routes in the MPLS `inet.3` route table. The `inet.3` route table is used to resolve routes for the PE router located in the other AS.

Specify the `l2vpn` family to indicate to the router that this is a VPLS. Specify the `signaling` option to configure BGP as the signaling protocol. This enables BGP to carry Layer 2 VPLS NLRI messages for this peer group.

Specify the `lo0` interface IP address of the PE as the neighbor. Configure an autonomous system identifier.

```
user@ASBR2# set protocols bgp group core-ibgp type internal
user@ASBR2# set protocols bgp group core-ibgp local-address 192.168.10.1
user@ASBR2# set protocols bgp group core-ibgp family inet labeled-unicast
resolve-vpn
user@ASBR2# set protocols bgp group core-ibgp family l2vpn signaling
user@ASBR2# set protocols bgp group core-ibgp neighbor 192.168.11.1
user@ASBR2# set routing-options autonomous-system 0.65020
```

2. On PE2, configure internal BGP. Specify the BGP type as internal. Specify the local address as the local `lo0` IP address.

Specify the `l2vpn` family to indicate this is a VPLS. Specify the `signaling` option to configure BGP as the signaling protocol. This enables BGP to carry Layer 2 VPLS NLRI messages.

Specify the `lo0` interface IP address of ASBR2 as the neighbor. Configure an autonomous system identifier.

```
user@PE2# set protocols bgp group core-ibgp type internal
user@PE2# set protocols bgp group core-ibgp local-address 192.168.11.1
user@PE2# set protocols bgp group core-ibgp family l2vpn signaling
user@PE2# set protocols bgp group core-ibgp neighbor 192.168.10.1
user@PE2# set routing-options autonomous-system 0.65020
```

3. On each router, commit the configuration:

```
user@host> commit check

configuration check succeeds

user@host> commit

commit complete
```

4. On PE2 and ASBR2, display BGP neighbor information and verify that the peer connection state is `Established`.

```
user@ASBR2> show bgp neighbor

Peer: 192.168.11.1+49443 AS 65020 Local: 192.168.10.1+179 AS 65020
Type: Internal State: Established Flags: ImportEval Sync
Last State: OpenConfirm Last Event: RecvKeepAlive
Last Error: None
Options: Preference LocalAddress AddressFamily Rib-group Refresh
Address families configured: l2vpn-signaling inet-labeled-unicast
Local Address: 192.168.10.1 Holdtime: 90 Preference: 170
Number of flaps: 0
Peer ID: 192.168.11.1 Local ID: 192.168.10.1 Active Holdtime:
```

```
90
  Keepalive Interval: 30      Peer index: 0
...

```

## Configuring LDP

**Step-by-Step Procedure** To configure LDP:

1. On the PE and ASBR routers, configure LDP with the Gigabit Ethernet interfaces between the PE and ASBR routers, and between the two ASBRs. To support LDP-signaled VPLS, additionally configure LDP with the lo0.0 interface on PE1 and ASBR1:

```
user@PE1# set protocols ldp interface ge-1/3/1.0
user@PE1# set protocols ldp interface lo0.0

```

```
user@ASBR1# set protocols ldp interface ge-0/3/1.0
user@ASBR1# set protocols ldp interface ge-0/3/0.0
user@ASBR1# set protocols ldp interface lo0.0

```

```
user@ASBR2# set protocols ldp interface ge-3/1/0.0
user@ASBR2# set protocols ldp interface ge-3/1/1.0

```

```
user@PE2# set protocols ldp interface ge-0/1/0.0

```



**NOTE:** The configuration of LDP signaling between the ASBRs is not required for Inter-AS VPLS. It is included here for reference only and might be used in LDP environments.

---

2. On each router, commit the configuration:

```
user@host> commit check

configuration check succeeds

user@host> commit

commit complete

```

3. Display LDP configuration information and verify that the correct interfaces are configured. LDP operation can be verified after MPLS is configured.

```
user@ASBR1> show configuration protocols ldp

interface ge-0/3/0.0;
interface ge-0/3/1.0;
interface lo0.0;

```



The preceding example is from ASBR1.

## Configuring MPLS

**Step-by-Step Procedure** To configure MPLS:

1. On the PE and ASBR routers, configure MPLS. Enable MPLS on the logical interfaces. Add the Gigabit Ethernet interfaces to the MPLS protocol. This adds entries to the MPLS forwarding table.

```
user@pe1# set protocols mpls interface ge-1/3/1.0
user@pe1# set interfaces ge-1/3/1 unit 0 family mpls
```

```
user@ASBR1# set protocols mpls interface ge-0/3/1.0
user@ASBR1# set protocols mpls interface ge-0/3/0.0
user@ASBR1# set interfaces ge-0/3/1 unit 0 family mpls
user@ASBR1# set interfaces ge-0/3/0 unit 0 family mpls
```

```
user@ASBR2# set protocols mpls interface ge-3/1/0.0
user@ASBR2# set protocols mpls interface ge-3/1/1.0
user@ASBR2# set interfaces ge-3/1/0 unit 0 family mpls
user@ASBR2# set interfaces ge-3/1/1 unit 0 family mpls
```

```
user@pe2# set protocols mpls interface ge-0/1/0.0
user@pe2# set interfaces ge-0/1/0 unit 0 family mpls
```

2. On each router, commit the configuration:

```
user@host> commit check

configuration check succeeds

user@host> commit

commit complete
```

3. On the PE and ASBR routers, display LDP neighbor information and verify that the directly connected LDP neighbors are listed:

```
user@ASBR1> show ldp neighbor

Address          Interface          Label space ID    Hold time
192.168.2.1      lo0.0              192.168.2.1:0     44
10.0.78.2        ge-0/3/0.0         192.168.10.1:0    13
10.0.23.9        ge-0/3/1.0         192.168.2.1:0     11
```

The preceding example is from ASBR1.

## Configuring the External BGP Peer Group Between the Loopback Interfaces

**Step-by-Step Procedure** To configure the external BGP (EBGP) peer group between the loopback interfaces:

1. On ASBR1 and PE1, configure an autonomous system identifier:

```
user@PE1# set routing-options autonomous-system 0.65010
```

```
user@ASBR1# set routing-options autonomous-system 0.65010
```

2. On ASBR1, configure an external BGP peer group for the loopback interfaces. Specify the external BGP group type. Include the multihop statement. Specify the local address as the local lo0 IP address. Configure the I2vpn family for BGP signaling. Configure the peer AS as the core AS number. Specify the lo0 IP address of ASBR2 as the neighbor.

```
user@ASBR1# set protocols bgp group vpls-core type external
user@ASBR1# set protocols bgp group vpls-core multihop
user@ASBR1# set protocols bgp group vpls-core local-address 192.168.3.1
user@ASBR1# set protocols bgp group vpls-core family I2vpn signaling
user@ASBR1# set protocols bgp group vpls-core peer-as 65020
user@ASBR1# set protocols bgp group vpls-core neighbor 192.168.10.1
```

3. On ASBR2, configure an external BGP peer group for the loopback interfaces. Specify the external BGP group type. Include the multihop statement. The multihop statement is needed because the EBGP neighbors are in different ASs. Specify the local address as the local lo0 IP address. Configure the I2vpn family for BGP signaling. Configure the peer AS as the metro AS number. Specify the lo0 IP address of ASBR1 as the neighbor.

```
user@ASBR2# set protocols bgp group vpls-metro type external
user@ASBR2# set protocols bgp group vpls-metro multihop
user@ASBR2# set protocols bgp group vpls-metro local-address 192.168.10.1
user@ASBR2# set protocols bgp group vpls-metro family I2vpn signaling
user@ASBR2# set protocols bgp group vpls-metro peer-as 65010
user@ASBR2# set protocols bgp group vpls-metro neighbor 192.168.3.1
```

4. On each router, commit the configuration:

```
user@host> commit
```

## Configuring the External BGP Peer Group Between the Inter-AS Link Interfaces

### Step-by-Step Procedure

The purpose of configuring external BGP peer groups between the inter-AS link interfaces is to create a full mesh of BGP LSPs among the ASBRs. To configure the external BGP peer group between the inter-AS link interfaces:

1. On ASBR1, configure a policy to export OSPF and direct routes, including the lo0 address of the PE routers, into BGP for the establishment of label-switched paths (LSPs):

```
user@ASBR1# set policy-options policy-statement loopback term term1 from
protocol ospf
user@ASBR1# set policy-options policy-statement loopback term term1 from
protocol direct
user@ASBR1# set policy-options policy-statement loopback term term1 from
route-filter 192.168.0.0/16 longer
user@ASBR1# set policy-options policy-statement loopback term term1 then
accept
```

2. On ASBR1, configure an external BGP peer group for the inter-AS link. Specify the external BGP group type. Specify the local inter-AS link IP address as the local address. Configure the inet family and include the labeled-unicast and resolve-vpn statements. The labeled-unicast statement advertises labeled routes out of the IPv4 inet.0 route table and places labeled routes into the inet.0 route table. The resolve-vpn option stores labeled routes in the MPLS inet.3 route table.

Include the export statement and specify the policy you created. Configure the peer AS as the core AS number. Specify the inter-AS link IP address of ASBR2 as the neighbor.

```
user@ASBR1# set protocols bgp group metro-core type external
user@ASBR1# set protocols bgp group metro-core local-address 10.0.78.1
user@ASBR1# set protocols bgp group metro-core family inet labeled-unicast
resolve-vpn
user@ASBR1# set protocols bgp group metro-core export loopback
user@ASBR1# set protocols bgp group metro-core peer-as 65020
user@ASBR1# set protocols bgp group metro-core neighbor 10.0.78.2
```

3. On ASBR2, configure a policy to export OSPF and direct routes, including the lo0 address, into BGP for the establishment of LSPs:

```
user@ASBR2# set policy-options policy-statement loopback term term1 from
protocol ospf
user@ASBR2# set policy-options policy-statement loopback term term1 from
protocol direct
user@ASBR2# set policy-options policy-statement loopback term term1 from
route-filter 192.168.0.0/16 longer
user@ASBR2# set policy-options policy-statement loopback term term1 then
accept
```

4. On ASBR2, configure an external BGP peer group for the inter-AS link. Specify the external BGP group type. Specify the local inter-AS link IP address as the

local address. Configure the `inet` family and include the `labeled-unicast` and `resolve-vpn` statements. Include the `export` statement and specify the policy you created. Configure the peer AS as the core AS number. Specify the inter-AS link IP address of ASBR1 as the neighbor.

```
user@ASBR2# set protocols bgp group core-metro type external
user@ASBR2# set protocols bgp group core-metro local-address 10.0.78.2
user@ASBR2# set protocols bgp group core-metro family inet labeled-unicast
resolve-vpn
user@ASBR2# set protocols bgp group core-metro export loopback
user@ASBR2# set protocols bgp group core-metro peer-as 65010
user@ASBR2# set protocols bgp group core-metro neighbor 10.0.78.1
```

5. On each router, commit the configuration:

```
user@host> commit check

configuration check succeeds

user@host> commit

commit complete
```

6. On ASBR1, display the BGP neighbors. Verify that the first peer is the IP address of the Gigabit Ethernet interface of ASBR2. Verify that the second peer is the IP address of the lo0 interface of ASBR2. Also verify that the state of each peer is `Established`. Notice that on ASBR1 the NLRI advertised by ASBR2 the inter-AS link peer is `inet-labeled-unicast` and the NLRI advertised by ASBR2 the loopback interface peer is `l2vpn-signaling`.

```
user@ASBR1> show bgp neighbor

Peer: 10.0.78.2+65473 AS 65020 Local: 10.0.78.1+179 AS 65010
  Type: External   State: Established   Flags: Sync
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: Cease
  Export: [ loopback ]
  Options: Preference LocalAddress AddressFamily PeerAS Rib-group Refresh

Address families configured: inet-labeled-unicast
Local Address: 10.0.78.1 Holdtime: 90 Preference: 170
Number of flaps: 3
Last flap event: Stop
Error: 'Cease' Sent: 1 Recv: 2
Peer ID: 192.168.10.1      Local ID: 192.168.3.1      Active Holdtime:
90
Keepalive Interval: 30      Peer index: 0
BFD: disabled, down
Local Interface: ge-0/3/0.0
NLRI for restart configured on peer: inet-labeled-unicast
NLRI advertised by peer: inet-labeled-unicast
NLRI for this session: inet-labeled-unicast
Peer supports Refresh capability (2)
Restart time configured on the peer: 120
Stale routes from peer are kept for: 300
```

```

Restart time requested by this peer: 120
NLRI that peer supports restart for: inet-labeled-unicast
NLRI that restart is negotiated for: inet-labeled-unicast
NLRI of received end-of-rib markers: inet-labeled-unicast
NLRI of all end-of-rib markers sent: inet-labeled-unicast
Peer supports 4 byte AS extension (peer-as 65020)
Table inet.0 Bit: 10000
  RIB State: BGP restart is complete
  Send state: in sync
  Active prefixes:          2
  Received prefixes:       3
  Accepted prefixes:       3
  Suppressed due to damping: 0
  Advertised prefixes:     3
  Last traffic (seconds): Received 8   Sent 3   Checked 60
  Input messages: Total 8713   Updates 3   Refreshes 0   Octets
165688
  Output messages: Total 8745   Updates 2   Refreshes 0   Octets
166315
  Output Queue[0]: 0

Peer: 192.168.10.1+51234 AS 65020 Local: 192.168.3.1+179 AS 65010
  Type: External   State: Established   Flags: Sync
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: Cease
  Options: Multihop Preference LocalAddress AddressFamily PeerAS Rib-group
Refresh
  Address families configured: 12vpn-signaling
  Local Address: 192.168.3.1 Holdtime: 90 Preference: 170
  Number of flaps: 3
  Last flap event: Stop
  Error: 'Cease' Sent: 1 Recv: 2
  Peer ID: 192.168.10.1   Local ID: 192.168.3.1   Active Holdtime:
90
  Keepalive Interval: 30   Peer index: 0
  BFD: disabled, down
  NLRI for restart configured on peer: 12vpn-signaling
  NLRI advertised by peer: 12vpn-signaling
  NLRI for this session: 12vpn-signaling
  Peer supports Refresh capability (2)
  Restart time configured on the peer: 120
  Stale routes from peer are kept for: 300
  Restart time requested by this peer: 120
  NLRI that peer supports restart for: 12vpn-signaling
  NLRI that restart is negotiated for: 12vpn-signaling
  NLRI of received end-of-rib markers: 12vpn-signaling
  NLRI of all end-of-rib markers sent: 12vpn-signaling
  Peer supports 4 byte AS extension (peer-as 65020)
Table bgp.12vpn.0 Bit: 20000
  RIB State: BGP restart is complete
  RIB State: VPN restart is complete
  Send state: in sync
  Active prefixes:          1
  Received prefixes:       1
  Accepted prefixes:       1
  Suppressed due to damping: 0
  Advertised prefixes:     1
Table inter-as.12vpn.0
  RIB State: BGP restart is complete
  RIB State: VPN restart is complete

```

```

Send state: not advertising
Active prefixes:          1
Received prefixes:       1
Accepted prefixes:       1
Suppressed due to damping: 0
Last traffic (seconds): Received 19 Sent 18 Checked 42
Input messages: Total 8712 Updates 3 Refreshes 0 Octets
165715
Output messages: Total 8744 Updates 2 Refreshes 0 Octets
166342
Output Queue[1]: 0
Output Queue[2]: 0

```

7. On ASBR2, display the BGP summary. Notice that the first peer is the IP address of the Gigabit Ethernet interface of ASBR1, the second peer is the IP address of the lo0 interface of ASBR1, and the third peer is the lo0 interface of PE2. Verify that the state of each peer is **Established**.

```

user@ASBR2> show bgp summary

Groups: 3 Peers: 3 Down peers: 0
Table          Tot Paths  Act Paths Suppressed  History Damp State
Pending
inet.0         3          2          0          0          0
0
bgp.l2vpn.0    2          2          0          0          0
0
Peer           AS         InPkt   OutPkt   OutQ   Flaps Last
Up/Dwn State|#Active/Received/Accepted/Damped...
10.0.78.1      65010     8781    8748     0      2 2d
17:54:56 Establ
inet.0: 2/3/3/0
192.168.3.1    65010     8780    8747     0      2 2d
17:54:54 Establ
bgp.l2vpn.0: 1/1/1/0
inter-as.l2vpn.0: 1/1/1/0
192.168.11.1   65020     8809    8763     0      1 2d
17:59:22 Establ
bgp.l2vpn.0: 1/1/1/0
inter-as.l2vpn.0: 1/1/1/0

```

8. On PE2, display the BGP group. Verify that the peer is the IP address of the lo0 interface of ASBR2. Verify that the number of established peer sessions is 1.

```

user@PE1> show bgp group

Group Type: Internal AS: 65020 Local AS: 65020
Name: core-ibgp Index: 1 Flags: Export Eval
Holdtime: 0
Total peers: 1 Established: 1
192.168.10.1+179
bgp.l2vpn.0: 1/1/1/0
inter-as.l2vpn.0: 1/1/1/0

Groups: 1 Peers: 1 External: 0 Internal: 1 Down peers: 0 Flaps:
7
Table          Tot Paths  Act Paths Suppressed  History Damp State
Pending
bgp.l2vpn.0    1          1          0          0          0

```

```

0
inte.12vpn.0      1      1      0      0      0
0

```

## Configuring the VPLS Routing Instances

**Step-by-Step Procedure** To configure the VPLS routing instances:

1. On PE1, configure the VPLS routing instance. To enable a VPLS instance, specify the **vpls** instance type. Configure VPLS on the CE-facing Gigabit Ethernet interface. Configure the CE-facing interface to use **ethernet-vpls** encapsulation.

```

user@PE1# set routing-instances metro instance-type vpls
user@PE1# set routing-instances metro interface ge-1/3/0.0

```

2. On PE1, configure the VPLS protocol within the routing instance. To uniquely identify the virtual circuit, configure the VPLS identifier. The VPLS identifier uniquely identifies each VPLS in the router. Configure the same VPLS ID on all the routers for a given VPLS.

Specify the IP address of the lo0 interface on ASBR2 as the neighbor.

Configure the CE-facing interface to use **ethernet-vpls** encapsulation and the **vpls** protocol family.

```

user@PE1# set routing-instances metro protocols vpls vpls-id 101
user@PE1# set routing-instances metro protocols vpls neighbor 192.168.3.1
user@PE1# set interfaces ge-1/3/0 encapsulation ethernet-vpls
user@PE1# set interfaces ge-1/3/0 unit 0 family vpls

```

3. On ASBR1, configure the VPLS routing instance. To enable a VPLS instance, specify the **vpls** instance type. Configure a route distinguisher and a VRF target. The **vrf-target** statement causes default VRF import and export policies to be generated that accept and tag routes with the specified target community.



**NOTE:** A route distinguisher allows the router to distinguish between two identical IP prefixes used as VPN routes. Configure a different route distinguisher on each ASBR router.

---



**NOTE:** You must configure the same VRF target on both ASBR routers.

---

```

user@ASBR1# set routing-instances inter-as instance-type vpls
user@ASBR1# set routing-instances inter-as route-distinguisher 65010:1
user@ASBR1# set routing-instances inter-as vrf-target target:2:1

```

4. On ASBR1, configure the VPLS protocol within the routing instance. Configure the VPLS identifier. Specify the IP address of the lo0 interface on PE1 as the neighbor.

```
user@ASBR1# set routing-instances inter-as protocols vpls vpls-id 101
user@ASBR1# set routing-instances inter-as protocols vpls neighbor
192.168.2.1
```



**NOTE:** The VPLS identifier uniquely identifies each LDP-signaled VPLS in the router. Configure the same VPLS ID on PE1 and ASBR1.

5. On ASBR1, configure the VPLS site within the routing instance. Configure the site identifier as required by the protocol to establish the EBGP pseudowire. As a best practice for more complex topologies involving multihoming, configure a site preference.

```
user@ASBR1# set routing-instances inter-as protocols vpls site ASBR-metro
site-identifier 1
user@ASBR1# set routing-instances inter-as protocols vpls site ASBR-metro
site-preference 1000
```

6. On ASBR1, configure the VPLS mesh group `peer-as` statement within the routing instance to specify which ASs belong to this AS mesh group. Configure the peer AS for the mesh group as all.

This statement enables the router to establish a single pseudowire between the ASBRs. VPLS NLRI messages are exchanged across the EBGP sessions on the inter-AS links between the ASBRs. All autonomous systems are in one mesh group.

```
user@ASBR1# set routing-instances inter-as protocols vpls mesh-group metro
peer-as all
```

7. On ASBR2, configure the VPLS routing instance. To enable a VPLS instance, specify the `vpls` instance type. Configure a route distinguisher and a VRF target. The `vrf-target` statement causes default VRF import and export policies to be generated that accept and tag routes with the specified target community.



**NOTE:** A route distinguisher allows the router to distinguish between two identical IP prefixes used as VPN routes. Configure a different route distinguisher on each ASBR router.



**NOTE:** You must configure the same VRF target community on both ASBR routers.

```
user@ASBR2# set routing-instances inter-as instance-type vpls
user@ASBR2# set routing-instances inter-as route-distinguisher 65020:1
user@ASBR2# set routing-instances inter-as vrf-target target:2:1
```

8. On ASBR2, configure the VPLS site within the routing instance. Configure the site identifier as required by the protocol.



```
user@ASBR2# set routing-instances inter-as protocols vpls site ASBR-core
site-identifier 2
```

9. On ASBR2, configure the VPLS mesh group within the routing instance to specify which VPLS PEs belong to this AS mesh group. Configure the peer AS for the mesh group as all.

This statement enables the router to establish a single pseudowire between the ASBRs. VPLS NLRI messages are exchanged across the EBGP sessions on the inter-AS links between the ASBRs. All autonomous systems are in one mesh group.

```
user@ASBR1# set routing-instances inter-as protocols vpls mesh-group core
peer-as all
```

10. On PE2, configure the VPLS routing instance. To enable a VPLS instance, specify the vpls instance type. Configure VPLS on the CE-facing Gigabit Ethernet interface. Configure a route distinguisher and a VRF target.

```
user@PE2# set routing-instances inter-as instance-type vpls
user@PE2# set routing-instances inter-as interface ge-0/1/1.0
user@PE2# set routing-instances inter-as route-distinguisher 65020:1
user@PE2# set routing-instances inter-as vrf-target target:2:1
```

11. On PE2, configure the VPLS site within the routing instance. Configure the site identifier as required by the protocol.

Configure the CE-facing interface to use ethernet-vpls encapsulation and the vpls protocol family.

```
user@PE2# set routing-instances inter-as protocols vpls site PE2 site-identifier
3
user@PE2# set interfaces ge-0/1/1 encapsulation ethernet-vpls
user@PE2# set interfaces ge-0/1/1 unit 0 family vpls
```

12. On each router, commit the configuration:

```
user@host> commit check

configuration check succeeds

user@host> commit

commit complete
```

13. On the PE routers, display the CE-facing Gigabit Ethernet interface information and verify that the encapsulation is configured correctly:

```
user@host> show interfaces ge-1/3/0

Address          Interface      Label space ID  Hold time
10.0.23.10       ge-1/3/1.0    192.168.3.1:0   11

Physical interface: ge-1/3/0, Enabled, Physical link is Up
```

```

Interface index: 147, SNMP ifIndex: 145
Link-level type: Ethernet, MTU: 1514, Speed: 1000Mbps, MAC-REWRITE Error:
None,
Loopback: Disabled, Source filtering: Disabled, Flow control: Enabled,
Auto-negotiation: Enabled, Remote fault: Online
Device flags   : Present Running
Interface flags: SNMP-Traps Internal: 0x4000
Link flags     : None
CoS queues     : 4 supported, 4 maximum usable queues
Schedulers    : 256
Current address: 00:12:1e:ee:34:db, Hardware address: 00:12:1e:ee:34:db

Last flapped   : 2008-08-27 19:02:52 PDT (5d 22:32 ago)
Input rate     : 0 bps (0 pps)
Output rate    : 0 bps (0 pps)
Ingress rate at Packet Forwarding Engine      : 0 bps (0 pps)
Ingress drop rate at Packet Forwarding Engine : 0 bps (0 pps)
Active alarms  : None
Active defects : None

Logical interface ge-1/3/0.0 (Index 84) (SNMP ifIndex 146)
Flags: SNMP-Traps Encapsulation: ENET2
Input packets : 0
Output packets: 1
Protocol inet, MTU: 1500
Flags: None
Addresses, Flags: Is-Preferred Is-Primary
Destination: 10.10.11/24, Local: 10.10.11.11, Broadcast:
10.10.11.255

```

**Results** The relevant sample configuration for the CE1 router follows.

```

Router CE1 interfaces {
  lo0 {
    unit 0 {
      family inet {
        address 192.168.1.1/32 {
          primary;
        }
        address 127.0.0.1/32;
      }
    }
  }
  ge-0/3/0 {
    unit 0 {
      family inet {
        address 10.10.11.1/24;
      }
    }
  }
}

```

The relevant sample configuration for the PE1 router follows.

```

Router PE1 interfaces {

```

```

lo0 {
  unit 0 {
    family inet {
      address 192.168.2.1/32 {
        primary;
      }
      address 127.0.0.1/32;
    }
  }
}
ge-1/3/0 {
  encapsulation ethernet-vpls;
  unit 0 {
    family vpls;
  }
}
ge-1/3/1 {
  unit 0 {
    family inet {
      address 10.0.23.9/30;
    }
    family mpls;
  }
}
routing-options {
  autonomous-system 0.65010;
}
protocols {
  mpls {
    interface ge-1/3/1.0;
  }
  ospf {
    traffic-engineering;
    area 0.0.0.1 {
      interface ge-1/3/1.0;
      interface lo0.0 {
        passive;
      }
    }
  }
}
ldp {
  interface ge-1/3/1.0;
  interface lo0.0;
}
routing-instances {
  metro {
    instance-type vpls;
    interface ge-1/3/0.0;
    protocols {
      vpls {
        vpls-id 101;
        neighbor 192.168.3.1;
      }
    }
  }
}

```

```
}  
}
```

The relevant sample configuration for the ASBR1 router follows.

```
Router ASBR1 interfaces {  
  lo0 {  
    unit 0 {  
      family inet {  
        address 192.168.3.1/32 {  
          primary;  
        }  
        address 127.0.0.1/32;  
      }  
    }  
  }  
  }  
  }  
  ge-0/3/0 {  
    unit 0 {  
      family inet {  
        address 10.0.78.1/30;  
      }  
      family mpls;  
    }  
  }  
  ge-0/3/1 {  
    unit 0 {  
      family inet {  
        address 10.0.23.10/30;  
      }  
      family mpls;  
    }  
  }  
  }  
  routing-options {  
    autonomous-system 0.65010;  
  }  
  protocols {  
    mpls {  
      interface ge-0/3/1.0;  
      interface ge-0/3/0.0;  
    }  
    bgp {  
      group vpls-core {  
        type external;  
        multihop;  
        local-address 192.168.3.1;  
        family l2vpn {  
          signaling;  
        }  
        peer-as 65020;  
        neighbor 192.168.10.1;  
      }  
      group metro-core {  
        type external;  
      }  
    }  
  }  
}
```

```

        local-address 10.0.78.1;
        family inet {
            labeled-unicast {
                resolve-vpn;
            }
        }
        export loopback;
        peer-as 65020;
        neighbor 10.0.78.2;
    }
}
ospf {
    traffic-engineering;
    area 0.0.0.1 {
        interface ge-0/3/1.0;
        interface lo0.0 {
            passive;
        }
    }
}
ldp {
    interface ge-0/3/0.0;
    interface ge-0/3/1.0;
    interface lo0.0;
}
}
policy-options {
    policy-statement loopback {
        term term1 {
            from {
                protocol [ ospf direct ];
                inactive: route-filter 10.0.0.0/8 longer;
                route-filter 192.168.0.0/16 longer;
            }
            then accept;
        }
    }
}
}
routing-instances {
    inter-as {
        instance-type vpls;
        route-distinguisher 65010:1;
        vrf-target target:2:1;
        protocols {
            vpls {
                site ASBR-metro {
                    site-identifier 1;
                    site-preference 10000;
                }
            }
            vpls-id 101;
            neighbor 192.168.2.1;
            mesh-group metro {
                peer-as {
                    all;
                }
            }
        }
    }
}
}

```

```

    }
  }
}

```

The relevant sample configuration for the ASBR2 router follows.

```

Router ASBR2 interfaces {
  lo0 {
    unit 0 {
      family inet {
        address 192.168.10.1/32 {
          primary;
        }
        address 127.0.0.1/32;
      }
    }
  }
  ge-3/1/0 {
    unit 0 {
      family inet {
        address 10.0.78.2/30;
      }
      family mpls;
    }
  }
  ge-3/1/1 {
    unit 0 {
      family inet {
        address 10.0.90.13/30;
      }
      family mpls;
    }
  }
}
routing-options {
  autonomous-system 0.65020;
}
protocols {
  mpls {
    interface ge-3/1/0.0;
    interface ge-3/1/1.0;
  }
  bgp {
    group core-ibgp {
      type internal;
      local-address 192.168.10.1;
      family inet {
        labeled-unicast {
          resolve-vpn;
        }
      }
      family l2vpn {
        signaling;
      }
    }
  }
}

```

```

        neighbor 192.168.11.1;
    }
    group vpls-metro {
        type external;
        multihop;
        local-address 192.168.10.1;
        family l2vpn {
            signaling;
        }
        peer-as 65010;
        neighbor 192.168.3.1;
    }
    group core-metro {
        type external;
        local-address 10.0.78.2;
        family inet {
            labeled-unicast {
                resolve-vpn;
            }
        }
        export loopback;
        peer-as 65010;
        neighbor 10.0.78.1;
    }
}
ospf {
    traffic-engineering;
    area 0.0.0.0 {
        interface ge-3/1/1.0;
        interface lo0.0 {
            passive;
        }
    }
}
ldp {
    interface ge-3/1/0.0;
    interface ge-3/1/1.0;
}
}
policy-options {
    policy-statement loopback {
        term term1 {
            from {
                protocol [ ospf direct ];
                route-filter 192.168.0.0/16 longer;
            }
            then accept;
        }
    }
}
}
routing-instances {
    inter-as {
        instance-type vpls;
        route-distinguisher 65020:1;
        vrf-target target:2:1;
        protocols {

```

```

    vpls {
      site ASBR-core {
        site-identifier 2;
      }
      mesh-group core {
        peer-as {
          all;
        }
      }
    }
  }
}

```

The relevant sample configuration for the PE2 router follows.

```

Router PE2 interfaces {
  lo0 {
    unit 0 {
      family inet {
        address 192.168.11.1/32 {
          primary;
        }
        address 127.0.0.1/32;
      }
    }
  }
  ge-0/1/0 {
    unit 0 {
      family inet {
        address 10.0.90.14/30;
      }
      family mpls;
    }
  }
  ge-0/1/1 {
    encapsulation ethernet-vpls;
    unit 0 {
      family vpls;
    }
  }
}
routing-options {
  autonomous-system 0.65020;
}
protocols {
  mpls {
    interface ge-0/1/0.0;
  }
  bgp {
    group core-ibgp {
      type internal;
      local-address 192.168.11.1;
      family l2vpn {
        signaling;
      }
    }
  }
}

```



```

    }
    neighbor 192.168.10.1;
  }
}
ospf {
  traffic-engineering;
  area 0.0.0.0 {
    interface ge-0/1/0.0;
    interface lo0.0 {
      passive;
    }
  }
}
ldp {
  interface ge-0/1/0.0;
}
}
routing-instances {
  inter-as {
    instance-type vpls;
    interface ge-0/1/1.0;
    route-distinguisher 65020:1;
    vrf-target target:2:1;
    protocols {
      vpls {
        site PE2 {
          site-identifier 3;
        }
      }
    }
  }
}
}
}

```

The relevant sample configuration for the CE2 router follows.

```

Router CE2 interfaces {
  lo0 {
    unit 0 {
      family inet {
        address 192.168.12.1/32 {
          primary;
        }
        address 127.0.0.1/32;
      }
    }
  }
  ge-0/1/1 {
    unit 0 {
      family inet {
        address 10.10.11.2/24;
      }
    }
  }
}
}

```

## Verification

To confirm that the complete configuration is working properly, perform these tasks:

- Verifying VPLS Connections on page 26
- Verifying End-to-End Traffic Flow on page 27

### Verifying VPLS Connections

**Purpose** To verify the VPLS connections have been established, enter the following command on the ASBR and PE routers.

**Action** user@PE1> **show vpls connections**  
Layer-2 VPN connections:

Legend for connection status (St)

EI -- encapsulation invalid	NC -- interface encapsulation not CCC/TCC/VPLS
EM -- encapsulation mismatch	WE -- interface and instance encaps not same
VC-Dn -- Virtual circuit down	NP -- interface hardware not present
CM -- control-word mismatch	-> -- only outbound connection is up
CN -- circuit not provisioned	<- -- only inbound connection is up
OR -- out of range	Up -- operational
OL -- no outgoing label	Dn -- down
LD -- local site signaled down	CF -- call admission control failure
RD -- remote site signaled down	SC -- local and remote site ID collision
LN -- local site not designated	LM -- local site ID not minimum designated
RN -- remote site not designated	RM -- remote site ID not minimum designated
XX -- unknown connection status	IL -- no incoming label
MM -- MTU mismatch	MI -- Mesh-Group ID not available
BK -- Backup connection	ST -- Standby connection

Legend for interface status

Up -- operational  
Dn -- down

Instance: metro

VPLS-id: 101

Neighbor	Type	St	Time last up	# Up trans
192.168.3.1(vpls-id 101)	rmt	Up	Sep 9 14:05:18 2008	1

Remote PE: 192.168.3.1, Negotiated control-word: No

Incoming label: 800001, Outgoing label: 800000

Local interface: vt-1/2/0.1048576, Status: Up, Encapsulation: ETHERNET

Description: Intf - vpls metro neighbor 192.168.3.1 vpls-id 101

user@ASBR1> **show vpls connections**

...

Instance: inter-as

BGP-VPLS State

Mesh-group connections: metro

Neighbor	Local-site	Remote-site	St	Time last up
192.168.10.1	1	2	Up	Sep 8 20:16:28 2008

Incoming label: 800257, Outgoing label: 800000

Local interface: vt-1/2/0.1049088, Status: Up, Encapsulation: VPLS

LDP-VPLS State

VPLS-id: 101

Mesh-group connections: \_\_ves\_\_

Neighbor	Type	St	Time last up	# Up trans
----------	------	----	--------------	------------

```

192.168.2.1(vpls-id 101) rmt Up Sep 9 14:05:22 2008 1
Remote PE: 192.168.2.1, Negotiated control-word: No
Incoming label: 800000, Outgoing label: 800001
Local interface: vt-0/1/0.1049089, Status: Up, Encapsulation: ETHERNET
Description: Intf - vpls inter-as neighbor 192.168.2.1 vpls-id 101

```

```
user@ASBR2> show vpls connections
```

```

...
Instance: inter-as
BGP-VPLS State
Mesh-group connections: __ves__
Neighbor Local-site Remote-site St Time last up
192.168.11.1 2 3 Up Sep 11 15:18:23 2008
Incoming label: 800002, Outgoing label: 800001
Local interface: vt-4/0/0.1048839, Status: Up, Encapsulation: VPLS
Mesh-group connections: core
Neighbor Local-site Remote-site St Time last up
192.168.3.1 2 1 Up Sep 8 20:16:28 2008
Incoming label: 800000, Outgoing label: 800257
Local interface: vt-4/0/0.1048834, Status: Up, Encapsulation: VPLS

```

```
user@PE2> show vpls connections
```

```

...
Instance: inter-as
Local site: PE2 (3)
connection-site Type St Time last up # Up trans
2 rmt Up Sep 8 20:16:28 2008 1
Remote PE: 192.168.10.1, Negotiated control-word: No
Incoming label: 800001, Outgoing label: 800002
Local interface: vt-0/3/0.1048832, Status: Up, Encapsulation: VPLS
Description: Intf - vpls inter-as local site 3 remote site 2

```

**Meaning** In the display from PE1, notice that the neighbor is the lo0 address of ASBR1 and that the status is up.

In the display from ASBR1, notice that the neighbor is the lo0 address of PE1 and that the status is up.

In the display from ASBR2, notice that the neighbor is the lo0 address of PE2 and that the status is up.

In the display from PE2, notice that the neighbor is the lo0 address of ASBR2 and that the status is up.

## Verifying End-to-End Traffic Flow

**Purpose** To verify that the CEs can send and receive traffic across the VPLS, use the ping command.

```

Action user@CE1> ping 10.10.11.2
PING 10.10.11.2 (10.10.11.2): 56 data bytes
64 bytes from 10.10.11.2: icmp_seq=0 ttl=64 time=1.369 ms
64 bytes from 10.10.11.2: icmp_seq=1 ttl=64 time=1.360 ms
64 bytes from 10.10.11.2: icmp_seq=2 ttl=64 time=1.333 ms
^C

```

```
user@CE2> ping 10.10.11.1
```

```
PING 10.10.11.1 (10.10.11.1): 56 data bytes
64 bytes from 10.10.11.1: icmp_seq=0 ttl=64 time=6.209 ms
64 bytes from 10.10.11.1: icmp_seq=1 ttl=64 time=1.347 ms
64 bytes from 10.10.11.1: icmp_seq=2 ttl=64 time=1.324 ms
^C
```

**Meaning** If CE1 can send and receive traffic from CE2 and CE2 can send and receive traffic from CE1, the VPLS is performing correctly.

---

Published: 2010-04-15