ERICSSON

Configuring Port Pseudowire Connections

SYSTEM ADMINISTRATOR GUIDE

90/1543-CRA 119 1170/1 Uen D

Contents

1 Overview

Multiprotocol Label Switching (MPLS) port pseudowires (PWs) provide point-to-point connections between pairs of provider edge (PE) routers, enabling you to connect, route, and forward Layer 2 (L2) networks to Layer 3 (L3) networks. Like physical Ethernet ports in the SmartEdge router, port PWs (containing untagged Ethernet circuits) can be bound to IP interfaces in the `local` context or a VPN context (one port PW to an interface). SmartEdge OS port PWs are viewed and configured in much the same way as physical ports.

Note: Port PW connections are only supported on PPA2 and PPA3 line cards.

This document applies to both the Ericsson SmartEdge® and SM family routers. However, the software that applies to the SM family of systems is a subset of the SmartEdge OS; some of the functionality described in this document may not apply to SM family routers.

For information specific to the SM family chassis, including line cards, refer to the SM family chassis documentation.

For specific information about the differences between the SmartEdge and SM family routers, refer to the Technical Product Description *SM Family of Systems* (part number 5/221 02-CRA 119 1170/1) in the **`Product Overview`** folder of this Customer Product Information library.

In a typical deployment, port PW connections are used to transport fixed IP traffic from the ingress L2PE device (PE2) to the egress L3PE router (PE1) as in the following diagram:

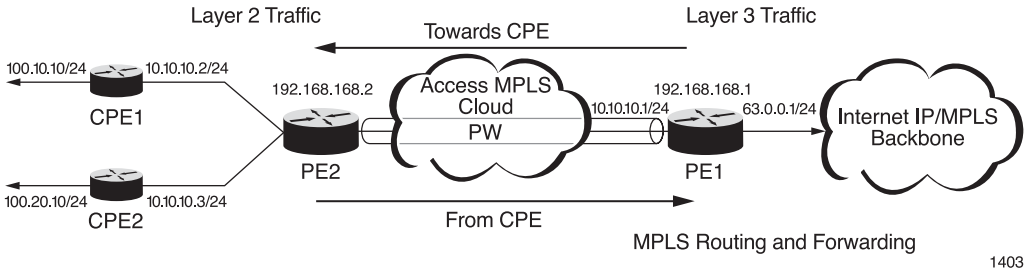

Figure 1 Typical PW Port Deployment

In this deployment, CPE1 and CPE2 are devices in the broadband access or other network topologies behind the L2PE node. Traffic from CPE1 and CPE2 is forwarded by PE2 through the port PW to PE1, where it is routed and forwarded into the L3 IP/MPLS network and the Internet.

The port PW is configured on PE1, leading to its peer, PE2, through the access MPLS cloud. The port PW interface and routing connect CPE devices behind the L2 PE to the Internet backbone or another L3 network.

PE2 is configured with an L2VPN Ethernet PW to PE1 that simply forwards L2 packets from the attachment circuit into the L2VPN PW, and back.

2 Configuration Tasks

This section describes how to configure port PW connections.

If PE2 is your router, configure L2VPN on it; see *Configuring L2VPN*.

2.1 Prerequisites

Port PWs operate over an underlying MPLS network with static routing or an Interior Gateway Protocol (IGP), either Open Shortest Path First (OSPF) or Intermediate System-to-Intermediate System (IS-IS). Configure the following prerequisite components:

2.1.1 Configuring an IGP

Before you configure a port PW, you must configure one of these IGPs; see the following documents:

- *Configuring Static Routes*
- *Configuring IS-IS*
- *Configuring OSPF*

2.1.2 Configuring MPLS

1. Configure the access MPLS network; for information on configuring MPLS, see *Configuring MPLS*.
2. Configure Label Distribution Protocol (LDP) or Resource Reservation Protocol (RSVP) signalling; see the following documents:
   - *Configuring LDP*
   - *Configuring RSVP*
3. Configure an LDP targeted neighbor session, required for PW connection; see *Configuring LDP*.

2.2 Set Up a Port PW

Configure the following components for each port PW:

2.2.1 Configuring the L2VPN Profile

Configure an L2VPN profile with the following steps:

1. Enter the *l2vpn profile* command in global configuration mode, and enter L2VPN profile configuration mode.

2. Add the peer to be connected by the port PW using the *peer* command.

2.2.2 Enabling a Control Word

Optionally, you can include a 4-byte control word in the Ethernet frame embedded between the PW label and the inner L2 header. The control word detects packet reordering and packet loss, and performs equal-cost multipath (ECMP) avoidance, and various operation, administration, and maintenance (OAM) tasks. During Ethernet PW setup, LDP advertises the control word capability to the underlying PW. If the peer does not have control word capabilities enabled, the control word is not included in the header of the packets that are sent over the PW. Therefore, the control word should be either enabled or disabled on both sides of the PW.

When the control word is present, all traffic follows a single path, because further lookups for the packet do not occur. The control word also permits the virtual circuit connectivity verification (VCCV) packet to follow the same path through the data plane that is taken by the PW data packets.

Note: This feature does not support sequencing, so no packet reordering is performed.

To enable a control word, use the *control-word* command in L2VPN profile peer configuration mode.

2.2.2.1 Using Virtual Circuit Connectivity Verification

VCCV provides a control channel on the PW that can be used for fault detection and diagnosis. VCCV verifies the connectivity of the PW using the label-switched path (LSP) or MPLS ping tool. The SmartEdge OS advertises VCCV RA + ACH (router alert + associated channel header) support in every PW establishment.

Both Type 1 and Type 2 VCCV are supported for operationally active and standby PW redundant pairs as follows:

- In-band VCCV, Type 1—Pseudowire Emulation Edge to Edge (PWE3) control word with 0001b as first nibble. Type 1 is supported only when a control word is enabled.

- Out-of-band VCCV, Type 2—MPLS router alert label. Type 2 is supported whether or not a control word is enabled.

  Note: Alternatively, a Type 2 VCCV control channel can be created by using the MPLS router alert label immediately above the PW label. However, this could result in a different ECMP hashing behavior than PW data protocol data units (PDUs), causing the VCCV control channel traffic to take a different path than the actual data traffic.

During PW setup, LDP dynamically engages in the following VCCV capability negotiation:

- If a control word is enabled on both ends and VCCV support is ACH, the PW uses Type 1.
- If a control word is enabled on both ends and VCCV support is RA, the PW uses Type 2 with ACH.
- If a control word is disabled on either end and VCCV support is RA, the PW uses Type 2 without ACH.

All other combinations are invalid, and VCCV ping will not be initiated.

To ping a particular PW to verify connectivity and display the VCCV capabilities, enter the *ping mpls pw* (**`ping mpls pw pw-id`** ***`pw-num`*** **`peer`** ***`ip-addr`*** [***`options`***]) command in exec mode.

2.2.3 Configuring QoS Propagation

Typically, on L3 circuits, when a packet arrives on a non-MPLS interface (or without labels on an MPLS interface), the IP DSCP bits are used to set up the packet descriptor (PD) Quality of Service (QoS) priority bits. If the packet arrives on an MPLS interface, the EXP bits from the first label in the packet are used to set up the PD. After the PD is set up on the ingress Packet Processing ASIC (iPPA), none of the remaining QoS bits in the packet are used to overwrite the PD value. The same default behavior is retained for traffic to the port PW. However, for inbound traffic from the port PW, the default PD propagation is from IP DSCP to PD, not EXP to PD, as in regular MPLS and virtual leased line (VLL) or Virtual Private LAN Services (VPLS) traffic.

Use the various **`propagate qos`** commands to configure QoS propagation either with or without custom class maps as described below. Three customized class maps at different levels can be configured on both inbound and outbound directions of the port PW:

- Global MPLS class map for the tunnel label. Use the *propagate qos to mpls* and *propagate qos from mpls* commands in MPLS router configuration mode.
  - PD to EXP (outbound)
  - EXP to PD (inbound)

- L2VPN class map for the PW label. There are two options:

  Global L2VPN class map. Use the *propagate qos to mpls* and *propagate qos from mpls* commands with the **`l2vpn class-map`** variable in MPLS router configuration mode.

  L2VPN class map per PW. Use the *propagate qos to mpls* and *propagate qos from mpls* commands in L2VPN peer profile configuration mode.

  - PD to EXP (outbound)
  - EXP to PD (inbound)

- IP class map at the port PW interface level. Use the *propagate qos to ip* and *propagate qos from ip* commands in interface configuration mode.

  - PD to DSCP (outbound)
  - DSCP to PD (inbound)

The SmartEdge OS supports one global MPLS class map and one global L2VPN class map under MPLS configuration mode for inbound and outbound packets. Port PWs can use the global L2VPN class map transparently. An IP class map configured at the interface level can also be used transparently by the port PW.

Note: The SmartEdge OS also supports a static EXP setting in the L2VPN profile for VLL PW, but this is not supported on port PW.

The per-PW class map is associated with the port PW using the L2VPN profile. Class map grid values are programmed to the PPA on the port PW L0 circuit. All PWs that use the same L2VPN profile are associated with the same class map grid. The global L2VPN and per-PW class maps can coexist on the same system. When both are present, the per-PW class map takes precedence over the global class map in both the inbound and outbound directions.

The PD specified by the ingress card (backbone interface) is used to populate the QoS bits of the packet headers going out on the PW. Optionally, all outbound propagate commands (**`propagate qos to`**) can be issued with or without a custom class map. The custom class map is not applied to the LSP label; PW label (inner label) EXP bits are copied into the LSP label (outer label) EXP bits.

QoS propagation for ingress traffic from the port is IP DSCP to PD (unlike MPLS and VLL or VPLS traffic, which is EXP to PD). Optionally, all inbound propagate commands (**`propagate qos from`**) can be issued with or without a custom class map to populate the PD value. When more than one class map is present, precedence is as follows:

1. LSP label
2. PW label
3. Inner IP DSCP

When no class maps are present for port PW label traffic, the IP DSCP value is used instead of EXP bits to populate the PD.

2.2.4 Configuring a Port PW

To configure a port PW perform the following steps:

1. Enter the *port pseudowire* command in global configuration mode (and enter port configuration mode).

   By default, the port has Ethernet encapsulation.

2. Enter the *no shutdown* command.

3. Assign a VC ID and L2VPN profile to the port PW, using the *vc-id (Port PW)* command (**vc-id *vc-id* profile *prof-name***).

   The value of the ***vc-id*** variable must be unique per router.

4. Optional. Assign other port PW details with the following commands:

   - Assign a description with the *description* command.
   - Assign a MAC address with the *mac-address* command.
   - Assign the maximum transmission unit for packets in the port PW with the *mtu* command.

2.2.5 Binding an Interface to the Port PW

When an interface is bound to a port PW, the SmartEdge OS treats it as an interface bound to a regular physical Ethernet port. It adds the routes to the routing table and detects that the next-hop is on the port PW by checking the port PW circuit that is bound to the interface.

You can bind a port PW to an interface in the `local` context or any VPN context.

For more information about creating interfaces, see *Configuring Contexts and Interfaces*.

1. For each port PW connection, create an interface leading to the L2 network; for example as in the diagram, 10.10.10.1/24.

2. Bind the port PW to the IP interface leading to the L2 network with the *bind interface* (**bind interface *int-name ctx-name***) command.

There are two ways you can configure IPoE customers over a port PW connection:

- The IPoE service is explicitly addressed over the port PW as a point-to-point service
- The IPoE service is an element on a LAN connected over the port PW, with communication by a broadcast service

2.2.6 Configuring IP Routing over the Port PW

Port PW connections support static routing with optional dynamically verified static routing (DVSR), Routing Information Protocol (RIP), Border Gateway Protocol (BGP), IS-IS, and OSPF protocols. By enabling routing on port PWs, enterprise, L3VPN, and other networking services are delivered using port PWs to customers connected to the L2VPN network.

Configure one of the following routing protocols:

- Static routing with optional DVSR (see *Configuring Static Routes*)
- RIP (see *Configuring RIP*)
- BGP (see *Configuring BGP*)
- IS-IS (see *Configuring IS-IS*)
- OSPF (see *Configuring OSPF*)

For L3VPN services, routing on a port PW interface in a VPN context emulates a provider edge to customer edge (PE-CE) connection.

2.3 Configuring Multicast over Port PW Connections

Multicast traffic forwarding is supported over PW connections in regular and L3VPN routing contexts. Use this feature to support customers that have dedicated links to a PE router that terminates in an L3VPN context. Internet Group Management Protocol (IGMP) is used on the CE-to-PE connection to join groups connected to CE routers that do not support Protocol Independent Multicast (PIM). PIM works over the port PW to enable the CE routers that do support it to join the multicast tree. Multicast traffic between PE routers is forwarded over the multicast distribution tree (MDT).

To enable multicast for port PW connections, configure PIM on each interface used to bind a port PW. Use the *pim sparse-mode* command in interface configuration mode. Whenever you enable PIM on an interface, IGMP also runs (IGMPv2, by default). If you use the **`pim sparse-mode passive`** command, only IGMP runs (not PIM).

3 Operations Tasks

This section describes the commands used to verify, manage, and troubleshoot port PW connections.

3.1 Verifying and Managing Port PW Connections

Use the following commands to verify that a port PW connection is up and passing traffic (see Figure 1 for the components named).

1. To check the status of port PW connections, examine log messages for port PW UP and DOWN messages.
2. To list the port PWs on the router, use the *show port pseudowire* command. Add the ***pw-name*** variable to show basic information about one specified port PW. You can add the **counters** keyword to display counters for the port PW.

   Alternatively, you can add the **detail** keyword for detailed information; the output displays the circuit handle, operational status, line state, administrative state, encapsulation type, MTU size, and MAC address for the port PW).
3. To verify the status of the port PW, use the *show port pseudowire* command for Ethernet port status or the *show mpls port pseudowire* command for MPLS details.
4. To verify IP connectivity with a device behind the L2 network, use the *ping* command; for example, to verify the connection from PE1 to CPE1, enter the **ping 10.10.10.2/24** command on PE1.
5. To verify traffic flow, use the *show circuit counters port-pseudowire* command with the ***pw-name*** argument or the **live**, or **detail** keywords.
6. To verify connected routes, use the *show arp-cache* command or the *show ip route* command with the **detail** or **next-hop** keywords.
7. To verify multicast over port PWs, use the following commands:
   - *show igmp group*
   - *show igmp circuit*
   - *show ip mfib*
   - *show ip mroute*
   - *show pim circuit*

3.2 Troubleshooting Port PW Connections

1. To check MPLS signalling, use one of the following commands:
   - *show mpls label-mapping*
   - *show rsvp lsp*
   - *show mpls lsp*
2. To check port PW neighbors, use the *show ldp neighbor* command with the **`detail`** keyword.
3. If there is a problem with BGP peering between PE1 and one of the CPEs, check BGP functions with the following commands:
   - *debug bgp rib*
   - *show bgp neighbor*
   - *show bgp route*
   - *show bgp notification*
4. If there is a problem with RIP peering between PE1 and one of the CPEs, check RIP functions with the following commands:
   - *debug rip interface*
   - *debug rip global-rib*
   - *show rip route*
   - *show ip route*
   - *show rip interface*
5. To investigate the L3VPN side, use the following commands:
   - *show bgp route ipv4 vpn*
   - *show ip route*
6. To investigate the L2PE node, use the following commands:
   - *show ldp l2vpn fec detail*
   - *show xc l2vpn*
7. To investigate the CPEs behind the L2 network, use the following commands:
   - *show rip route*
   - *show bgp route*

- *show arp-cache*

4 Configuration Examples

This section provides examples of configuring the port PW components and the complete configuration of the node, PE1.

The first example shows how to configure an L2VPN profile, `l2_profile_CPE1` (in global configuration mode), which associates PW ports with the peer `192.168.168.2` and enters L2VPN profile configuration mode. This example assumes that the underlying MPLS network and IGP are already configured; see the complete configuration for PE1 below.

```
[local]Redback(config)#l2vpn profile l2_profile_CPE1
 [local]Redback(config-l2vpn-xc-profile)#peer 192.168.168.2
```

The following example in the `l3vpn-1` context configures the `to-CPE1` interface to be used by the port PW.

For a diagram of the connections, see Figure 1.

```
[local]Redback(config)#context l3vpn-1
[local]Redback(config-ctx)#interface to-CPE1
[local]Redback(config-if)#ip address 10.10.10.1/24
```

The following example configures the port PW in global configuration mode (and enters port configuration mode), adds an optional description, configures the port PW to be in the Up state, binds it to the previously configured interface, assigns a VC-ID and associates it with the previously configured L2VPN profile.

```
[local]Redback(config)#port pseudowire l2-net
[local]Redback(config-port)#description Connects to the L2 network of CPE1.
[local]Redback(config-port)#no shutdown
[local]Redback(config-port)#bind interface to-CPE1 l3vpn-1
[local]Redback(config-port)#vc-id 100 profile l2_profile_CPE1
```

The following example shows these configurations in the end-to-end configuration of the node, PE1 in Figure 1. Although in this example the port PW is bound to a VPN context, they can be bound to any interface leading to the l2 network.

```
service multiple-contexts
!
!
!
!
! l2vpn profile configuration
l2vpn profile l2_profile_CPE1
  peer 192.168.168.2
!
```

```
context local
!
 no ip domain-lookup
!
router-id 192.168.168.1
!
 interface loopback loopback
  ip address 192.168.168.1/32
!
 interface mgmt
  ip address 10.18.20.116/24
!
 interface to-MPLS-BACKBONE
  ip address 172.172.200.1/24
!
 interface to-PE2
  ip address 172.172.100.1/24
 logging console
!
 router ospf 1
  fast-convergence
  area 0.0.0.0
   interface to-MPLS-BACKBONE
   interface to-PE2
   interface loopback
!
 router mpls
  interface to-MPLS-BACKBONE
  interface to-PE2
  interface loopback
!
 router ldp
  neighbor 192.168.168.2 targeted
  interface to-MPLS-BACKBONE
  interface to-PE2
  interface loopback
!
 router bgp 65535
!
  neighbor 192.168.168.3 internal
    update-source loopback
   address-family ipv4 unicast
   address-family ipv4 vpn

context l3vpn-1 vpn-rd 65535:1
!
 no ip domain-lookup
!
 interface to-CPE1
  ip address 10.10.10.1/24
 no logging console
!
 router bgp vpn
  address-family ipv4 unicast
   export route-target 65535:1
   import route-target 65535:1
   redistribute connected
!
!
card ge4-20-port 9
!
port ethernet 9/4
 no shutdown
 bind interface to-MPLS-BACKBONE local
!
card 10ge-4-port 10
!
port ethernet 10/2
 no shutdown
 bind interface to-PE2 local
!
 !
port pseudowire l2-net
 description connects to l2 network of CPE1
 no shutdown
```

```
  bind interface to-CPE1 l3vpn-1
 vc-id 100 profile l2_profile_CPE1
!
 system hostname PE1
!
 boot configuration /flash/admin.cfg
 no timeout session idle
!
```

The following example shows how to configure three port PWs (with multicast enabled) on the interfaces to which they are bound (`pwif1` and `pwif2` in the `local` context and `pwif3` in the `ctx2` context). It also shows how to enable multicast (and IGMP) on a subscriber interface leading to a multicast source:

```
!
!
!
!
!
service multiple-contexts
!
!
!
!
! l2vpn profile configuration
l2vpn profile prof1
  peer 1.1.1.1
!
!
!
!
!
!
!
context local
!
!
 interface lo1 loopback
  ip address 2.2.2.2/32
!
 interface mgmt
  ip address 172.31.23.121/24
!
 interface pwif1<<<<<<< port PW pw1 bound to this interface.
  ip address 10.1.1.2/24
  pim sparse-mode<<<<<<< PIM-SM enabled for this interface.
!
 interface pwif2<<<<<<< port PW pw2 bound to this interface.
  ip address 10.1.2.2/24
  pim sparse-mode<<<<<<< PIM-SM enabled for this interface.
!
 interface to-dev1
  ip address 20.1.1.2/24
```

```
!
 interface to-dst
  ip address 30.1.1.1/24
  pim sparse-mode
 logging console
!
 router ospf 1
  fast-convergence
  area 0.0.0.1
   interface lo1
   interface to-dev1
!
!
 router rip rip01
  redistribute connected
  interface pwif1
  interface pwif2
  interface to-dst
!
 router mpls
  interface to-dev1
!
 router ldp
  interface lo1
  interface to-dev1
!
!
!
!
context ctx2
!
!
 interface to-host
  ip address 40.1.1.1/24
  pim sparse-mode passive
!
 interface pwif3<<<<<<< port PW pw3 bound to this interface.
  ip address 10.1.3.2/24
  pim sparse-mode<<<<<<< PIM-SM enabled for this interface.
 no logging console
!
 router rip rip02
  redistribute connected
  interface pwif3
  interface to-host
!
!
!
!
! ** End Context **
!
```

```
!
!
!
!
!
card ge-20-port 2
!
port ethernet 2/10
 no shutdown
 bind interface to-host ctx2
!
port ethernet 2/11
 no shutdown
 bind interface to-dst local
!
card ge-10-port 3
!
port ethernet 3/10
 no shutdown
 bind interface to-dev1 local
!
!
!
 !
port pseudowire pw1
 no shutdown
   bind interface pwif1 local
 vc-id 100 profile prof1
 !
port pseudowire pw2
 no shutdown
   bind interface pwif2 local
 vc-id 200 profile prof1
 !
port pseudowire pw3
 no shutdown
   bind interface pwif3 ctx2
 vc-id 300 profile prof1
```

5 Operations Examples

This section shows how to verify the port PW.

The following example confirms that OSPF is up between PE1 and PE2:

```
[local]PE1#show ospf neighbor

  --- OSPF Neighbors for Instance 1/Router ID 192.168.168.1 ---

NeighborID      NeighborAddress Pri State    DR-State IntfAddress     TimeLeft
192.168.168.2   172.172.100.2   1   Full     DR       172.172.100.1   37
192.168.168.3   172.172.200.2   1   Full     DR       172.172.200.1   32


[local]PE1#show ldp neighbor
PeerFlags: A - LocalActiveOpen, D - Deleted, R - Reseting, E - OpenExtraDelay
           N - OpenNoDelay, P - SetMD5Passwd, T - RetainRoute, F - FlushState
           X - ExplicitNullEnabled, C - ExplicitNullStatusChanging
           G - Graceful Restart Supported, L - Session Life Extended
           V - Reachable Via RSVP-TE LSP
SHld - Session Holdtime Left, HHld - Hello Holdtime Left

NeighborAddr    LDP Identifier        State   Flag SHld HHld Interface
192.168.168.2   192.168.168.2:0       Oper    G    80   12   to-PE2
                                                        35   none - remote
192.168.168.3   192.168.168.3:0       Oper    G    66   10   to-MPLS-BACKBONE
```

The following example confirms that the LDP neighbors are UP between PE1 and PE2:

```
[local]PE1#show ldp neighbor
PeerFlags: A - LocalActiveOpen, D - Deleted, R - Reseting, E - OpenExtraDelay
           N - OpenNoDelay, P - SetMD5Passwd, T - RetainRoute, F - FlushState
           X - ExplicitNullEnabled, C - ExplicitNullStatusChanging
           G - Graceful Restart Supported, L - Session Life Extended
           V - Reachable Via RSVP-TE LSP
SHld - Session Holdtime Left, HHld - Hello Holdtime Left

NeighborAddr    LDP Identifier        State   Flag SHld HHld Interface
192.168.168.2   192.168.168.2:0       Oper    G    80   12   to-PE2
                                                        35   none - remote
192.168.168.3   192.168.168.3:0       Oper    G    66   10   to-MPLS-BACKBONE
```

The following examples confirm that the port PW l2-net is UP:

```
[local]PE1#show port pseudowire
Name                            CCT                  State
l2-net                          255/25:1:1/1/0/9     Up

[local]PE1#show port pseudowire detail

l2-net 255/25:1:1/1/0/9 state is Up
Description               : connects to l2 network of CPE1

Line state                : Up
Admin state               : Up
Encapsulation             : ethernet
MTU size                  : 1500 Bytes
MAC address               : 00:30:88:02:52:3d


[local]PE1#show mpls port pseudowire detail

Name            : l2-net
Oper State      : Up              Admin State      : Enable
Peer            : 192.168.168.2   VC ID            : 100
Current State   : UP              Prev State       : DOWN
Last Event      : XC_UP           Event Flags      : 0x0000012f
L0 CCT          : 255/25:1:1/1/0/9
L1 CCT          : 255/25:1:1/1/1/10
PW in label     : 131072          PW out label     : 131072
PW local MTU    : 1500            PW remote MTU    : 1500
Profile name    : l2_profile_CPE1 Context          : Local
Local VC Type   : Ethernet        Remote VC Type   : Ethernet
LSP Configured  :                 LSP Used         :
```

The following example confirms that the IP interface is in the UP state for the port-PW:

```
[local]PE1#context l3vpn-1
[l3vpn-1]PE1#show ip interface brief

Fri Nov 19 21:25:16 2010
Name              Address                  MTU   State    Bindings
to-CPE1           10.10.10.1/24            1500  Up       ethernet PORT PW 0
[l3vpn-1]PE1#
```

The following example confirms that ARP is UP for the l2-net port PW:

```
[l3vpn-1]PE1#show arp-cache
Total number of arp entries in cache: 2
  Resolved entry   : 2
  Incomplete entry : 0

Host              Hardware address    Ttl    Type  Circuit
10.10.10.1        00:30:88:02:52:3d   -      ARPA  PORT PW 0
10.10.10.2        00:30:88:14:3d:94   2793   ARPA  PORT PW 0
```

The following example pings a remote CPE through the l2-net port PW:

```
[l3vpn-1]PE1#ping 10.10.10.2
PING 10.10.10.2 (10.10.10.2): source 10.10.10.1, 36 data bytes,
timeout is 1 second
!!!!!

----10.10.10.2 PING Statistics----
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.716/1.892/2.142/0.160 ms
[l3vpn-1]PE1#
```