Understand IOS XR L2VPN Services and Features

Contents

Introduction

Background Information

1. Point-to-Point and Multipoint Services

1.1 Point-to-Point Service

1.2 Multipoint Service

2. Attachment Circuits

2.1 ASR 9000 Ethernet Virtual Circuit

2.1.1 Incoming Interface Matching

2.1.2 VLAN Manipulation

2.2 Cisco IOS® XR Non-EVC Router Behavior (CRS and XR12000)

3. Point-to-Point Service

3.1 Local Switching

3.1.1 Main Interface

3.1.2 Subinterfaces and VLAN Manipulation

3.1.2.1 Main Interface and Dot1q Subinterface

3.1.2.2 Subinterface with Encapsulation

3.1.2.3 Ingress Direction on GigabitEthernet0/1/0/1.1

3.2 Virtual Private Wire Services

3.2.1 Overview

3.2.2 PW and AC Coupled Status

3.2.3 Type 4 and Type 5 PWs

3.2.4 Multi-segment PW

3.2.5 Redundancy

3.2.5.1 Core Redundancy

3.2.5.2 Bundle Over PWs

3.2.5.3 PW Redundancy

3.2.5.4 ASR 9000 nV Edge Cluster

3.3 CDP

3.3.1 CDP Not Enabled on Main Interface of L2VPN PE

3.3.2 CDP Enabled on Main Interface of L2VPN PE

3.4 Spanning Tree

4. Multipoint Service

4.1 Local Switching

4.2 Full MST

4.3 BVI

4.4 VPLS

4.4.1 Overview

4.4.2 PW Types and Transported Tags

4.4.3 Autodiscovery and Signaling

4.4.3.1 BGP Autodiscovery and BGP Signaling

4.4.3.2 BGP Autodiscovery and LDP Signaling

4.4.4 MAC Flushes and Withdrawals

4.4.5 H-VPLS

4.4.6 Split Horizon Groups (SHGs)
4.4.7 Redundancy
4.4.7.1 Spanning Tree
4.4.7.2 MSTAG
4.4.7.3 PVSTAG or PVRSTAG
4.4.7.4 MC-LAG
4.4.7.5 ASR 9000 nV Edge Cluster
4.4.7.6ICCP-Based Service Multi-Homing (ICCP-SM) (PMCLAG (Pseudo MCLAG) and Active/Active)
4.5 Traffic Storm Control
4.6 MAC Moves
4.7 IGMP and MLD Snooping
5. Additional L2VPN Topics
5.1 Loadbalancing
5.2 Logging
5.3 ethernet-services access-list
5.4 ethernet egress-filter

Introduction

This document describes basic Layer 2 (L2) VPN (L2VPN) topologies and presents basic examples to demonstrate design, services, and configuration.

Background Information

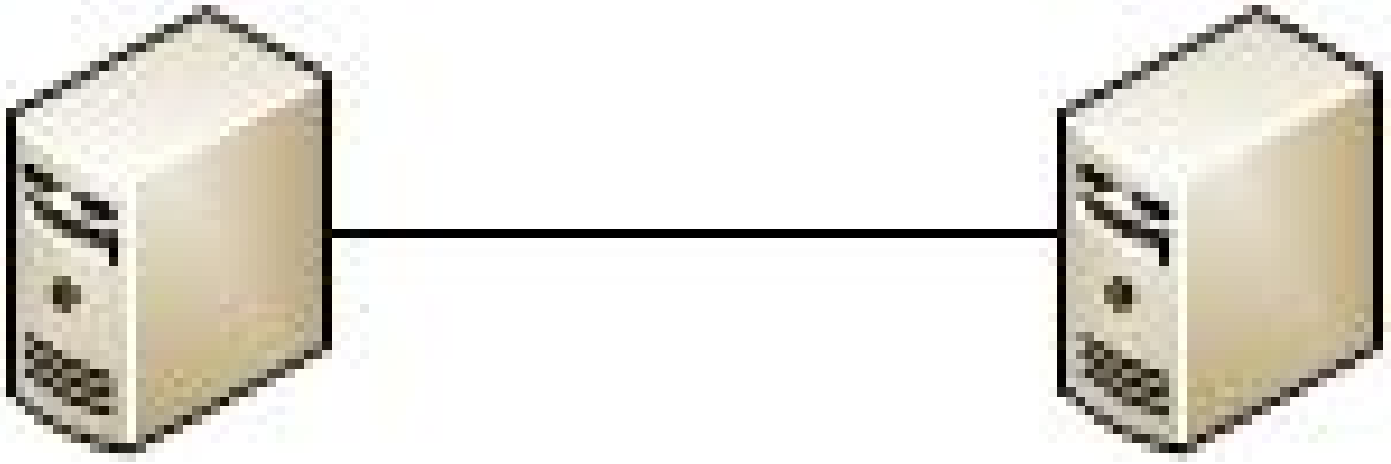
See the L2VPN and Ethernet Services Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.4.x for additional information.

1. Point-to-Point and Multipoint Services

The L2VPN feature provides the ability to provide point-to-point and multipoint services.

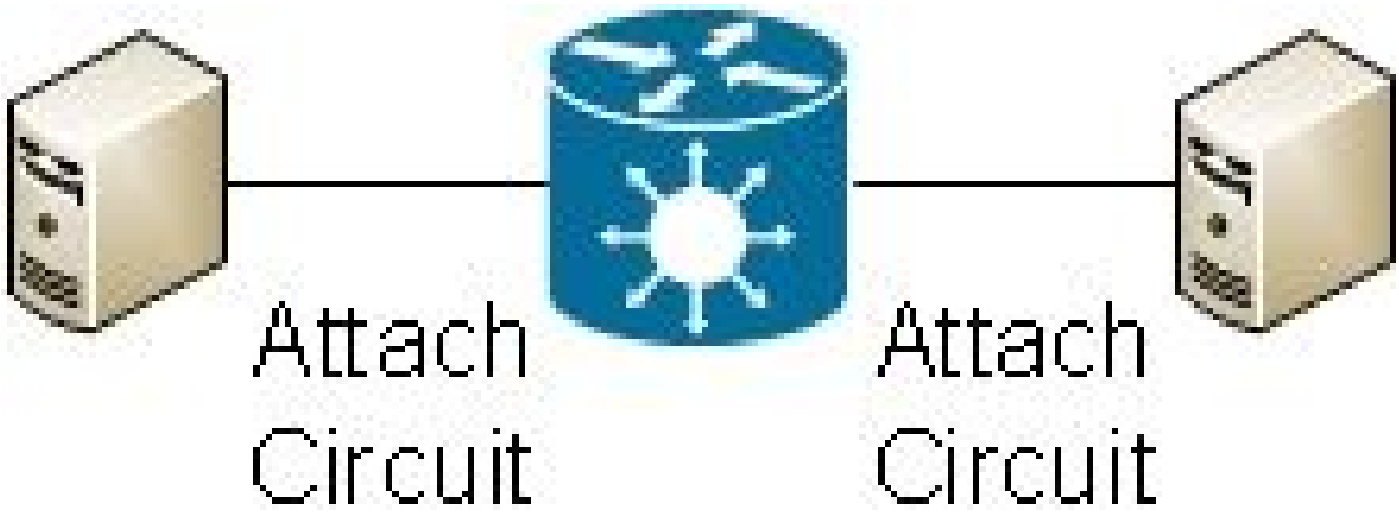
1.1 Point-to-Point Service

Point-to-point service basically emulates a transport circuit between two end nodes so the end nodes appear to be directly connected over a point-to-point link. This can be used to connect two sites.

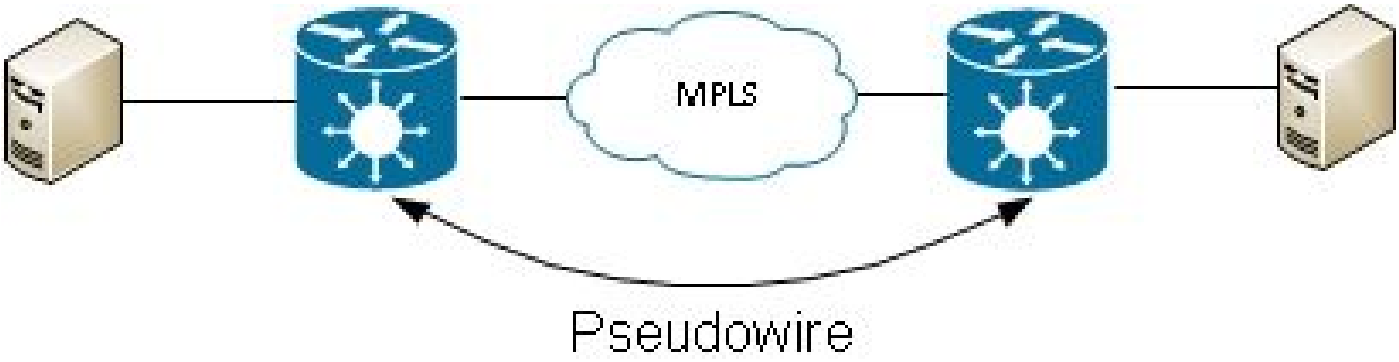


In reality, there can be multiple routers between the two end nodes, and there can be multiple designs to provide the point-to-point service.

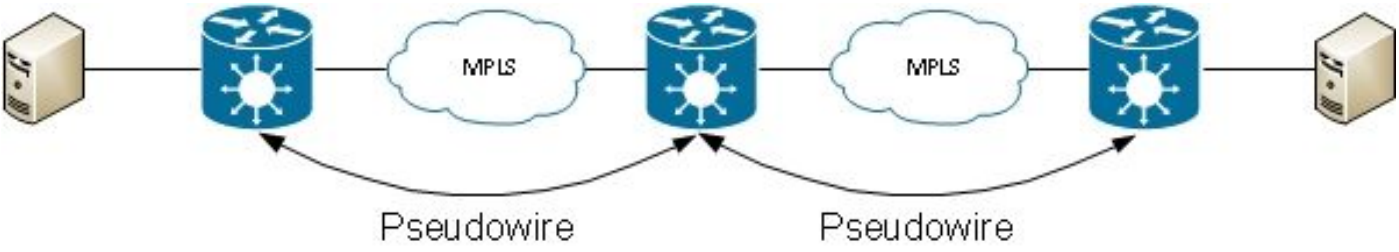
One router can do local switching between two of its interfaces:



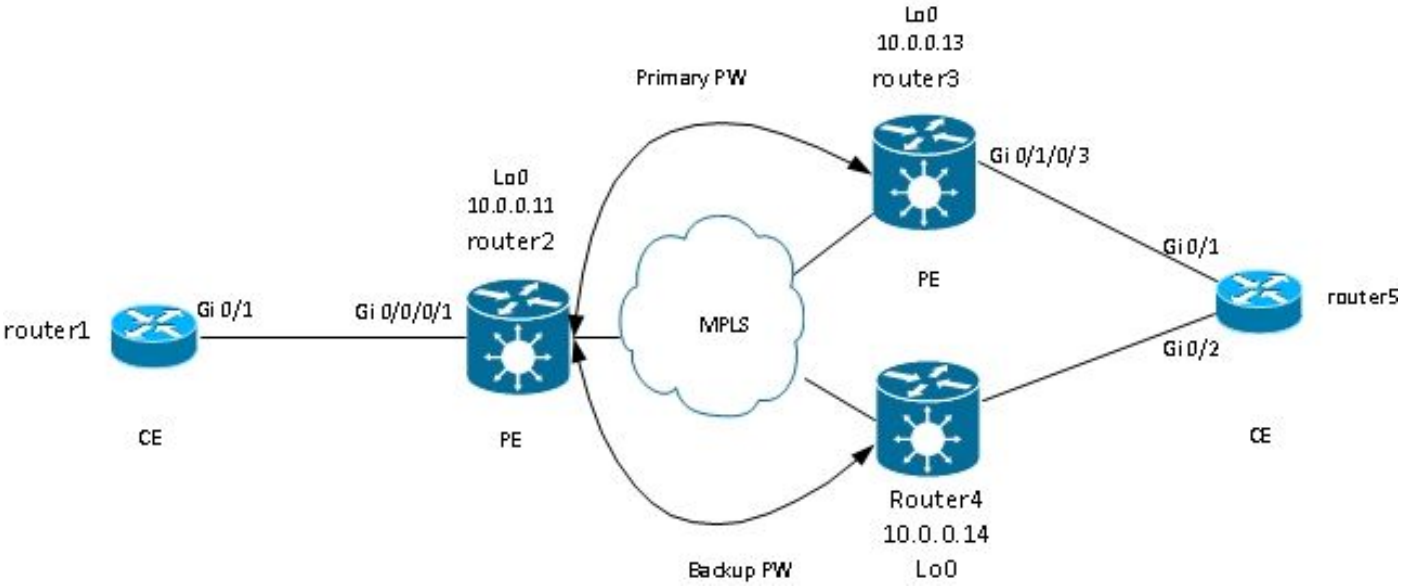
There can also be a Multiprotocol Label Switching (MPLS) pseudowire (PW) between two routers:



A router can switch frames between two PWs; in this case, this is a multi-segment PW:



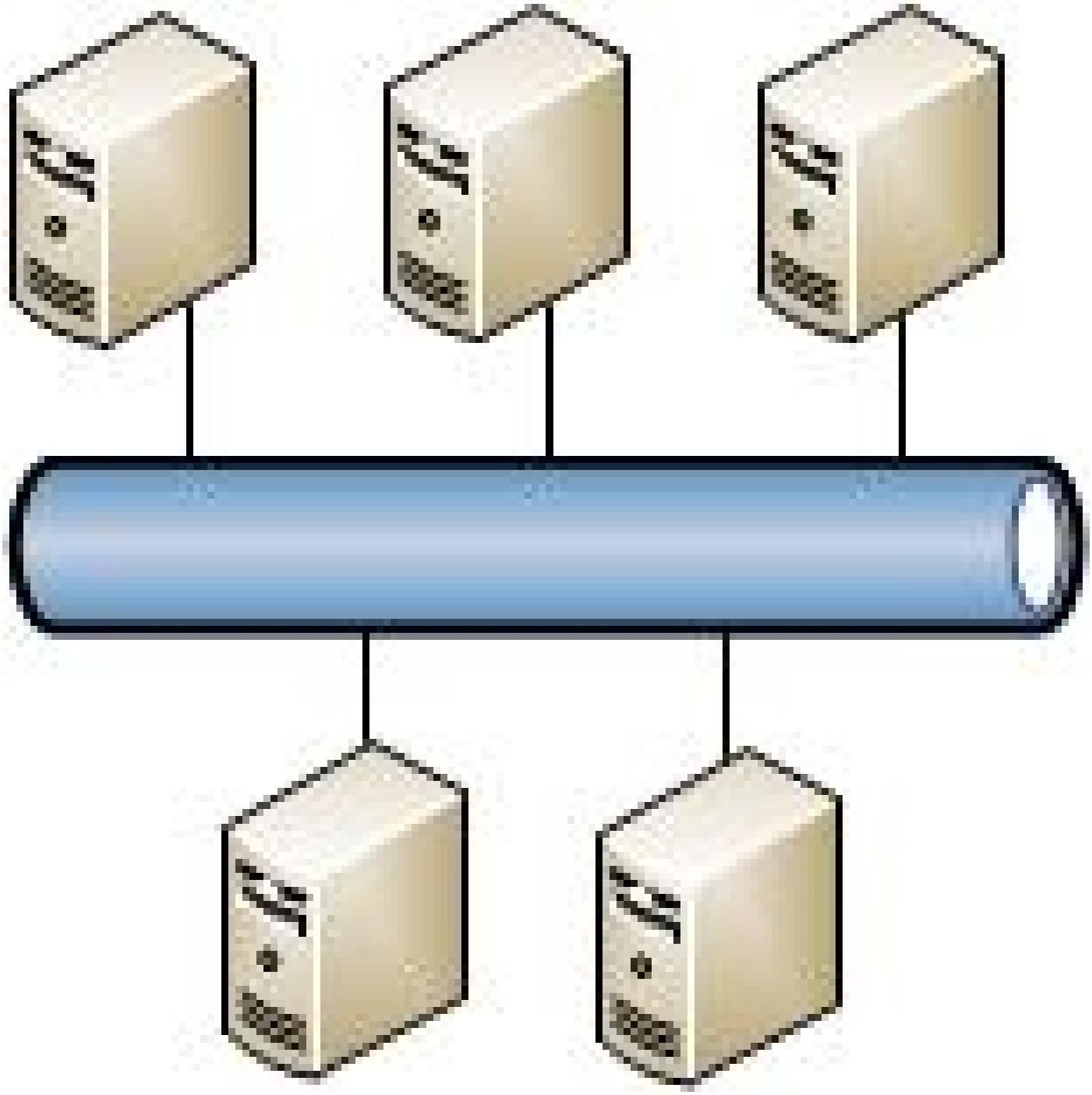
Redundancy is available through the PW redundancy feature:



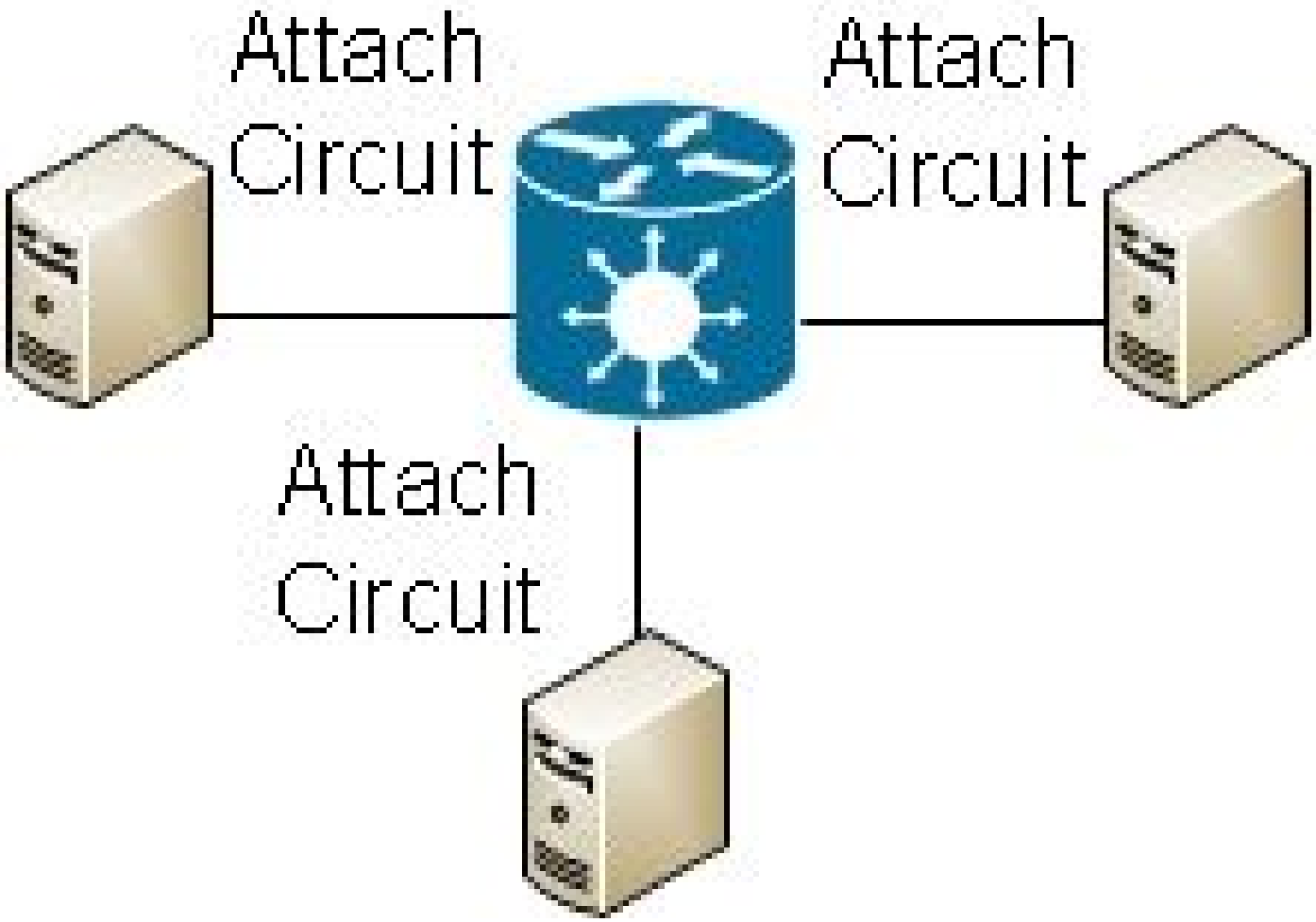
Other designs are available, but cannot all be listed here.

1.2 Multipoint Service

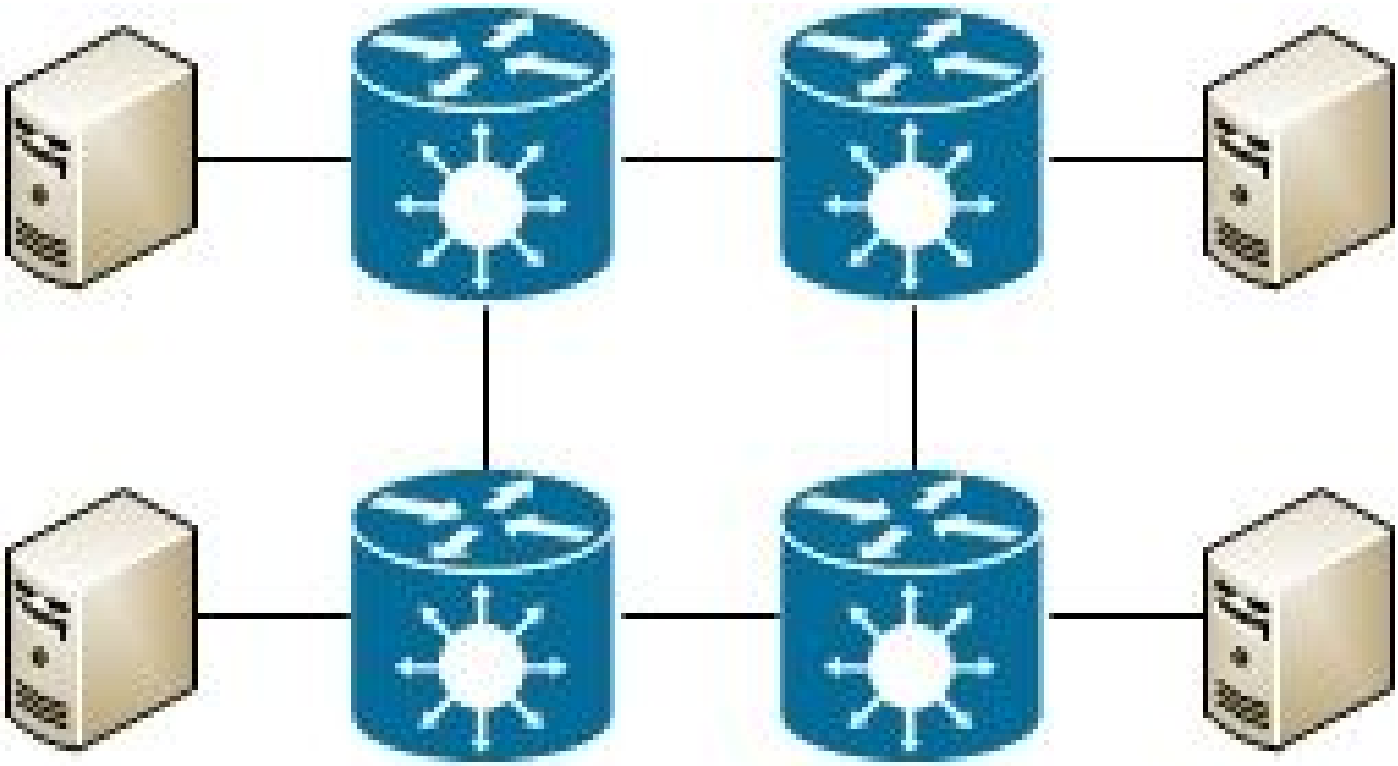
Multipoint service emulates a broadcast domain so that all hosts connected in that bridge-domain appear to be logically connected to the same Ethernet segment:



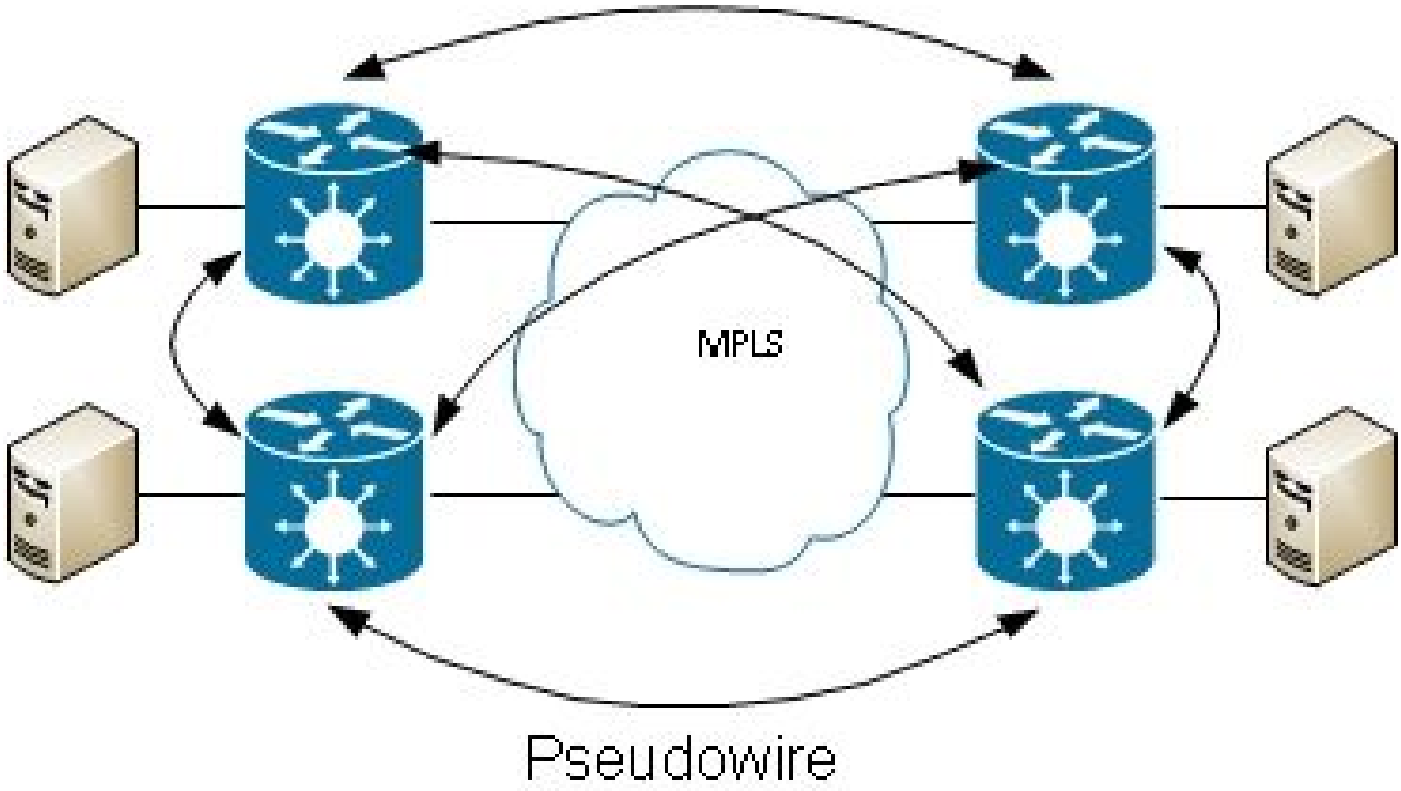
All hosts can be connected to the same router/switch:



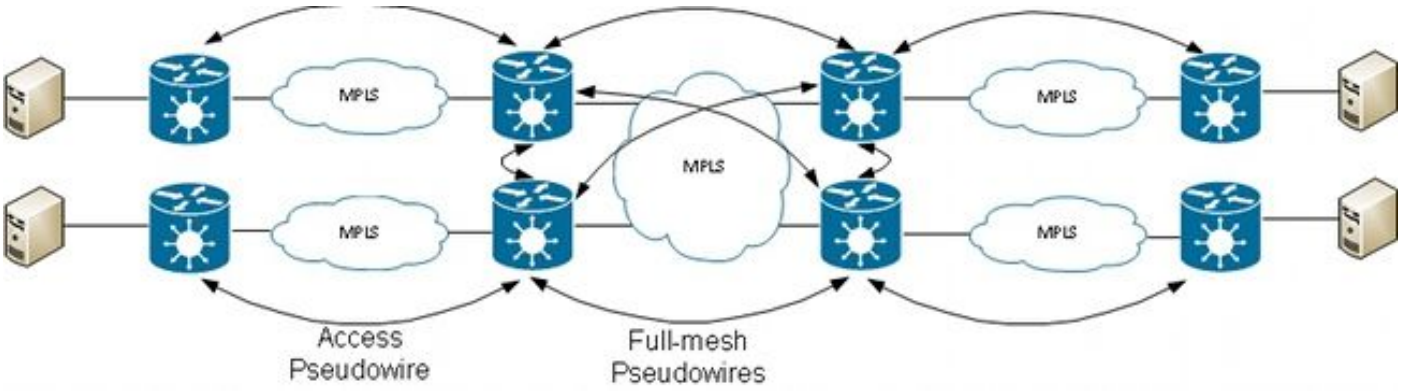
Multiple switches can do traditional Ethernet switching; spanning tree must be used in order to break loops:



Virtual Private LAN Services (VPLS) lets you extend the broadcast domain between multiple sites using MPLS PWs:



Hierarchical VPLS can be used in order to increase scalability:



2. Attachment Circuits

2.1 ASR 9000 Ethernet Virtual Circuit

2.1.1 Incoming Interface Matching

Basic rules for attachment circuits (ACs) include:

- A packet must be received on an interface configured with the l2transport keyword in order to be processed by the L2VPN feature.
- This interface can be a main interface, where the **l2transport** command is configured under the interface config mode, or a subinterface, where the l2transport keyword is configured after the subinterface number.
- A longest match lookup determines the incoming interface of the packet. The longest match lookup checks these conditions in this order to match the incoming packet to a subinterface:

1. The incoming frame has two dot1q tags and matches a subinterface configured with the same two dot1q tags (802.1Q tunneling, or QinQ). This is the longest possible match.
2. The incoming frame has two dot1q tags and matches a subinterface configured with the same dot1q first tag and any for the second tag.
3. The incoming frame has one dot1q tag and matches a subinterface configured with the same dot1q tag and the exact keyword.
4. The incoming frame has one or more dot1q tags and matches a subinterface configured with one of the dot1q tags.
5. The incoming frame has no dot1q tags and matches a subinterface configured with the **encapsulation untagged** command.
6. The incoming frame fails to match any other subinterface, so it matches a subinterface configured with the **encapsulation default** command.
7. The incoming frame fails to match any other subinterface, so it matches the main interface that is configured for l2transport.

- On traditional routers that do not use the Ethernet Virtual Connection (EVC) model, the VLAN tags configured under the subinterface are removed (popped) from the frame before they are transported by the L2VPN feature.
- On a Cisco ASR 9000 Series Aggregation Services Router that uses the EVC infrastructure, the default action is to preserve the existing tags. Use the **rewrite** command to modify the default.
- If there is a Bridge Virtual Interface (BVI) in the bridge-domain, all incoming tags can be popped because the BVI is a routed interface without any tag. See the BVI section for details.

Here are several examples that illustrate these rules:

1. A basic example is when all traffic received on a physical port must be transported, whether or not it has a VLAN tag. If you configure l2transport under the main interface, all traffic received on that physical port is transported by the L2VPN feature:

   ```
   <#root>

   interface GigabitEthernet0/0/0/2


   l2transport
   ```

2. Bundle interfaces and subinterfaces can be configured as l2transport:

   ```
   interface Bundle-Ether1
    l2transport
   ```

3. Use **encapsulation default** under an l2transport subinterface to match any tagged or untagged traffic that has not been matched by another subinterface with a longest match. (See Example 4). The l2transport keyword is configured in the subinterface name, not under the subinterface as on the main interface:

   ```
   <#root>

   interface GigabitEthernet0/1/0/3.1 l2transport


   encapsulation default
   ```

Configure **encapsulation untagged** if you want to match only untagged frames.

4. When there are multiple subinterfaces, run the longest match test on the incoming frame in order to determine the incoming interface:

```
interface GigabitEthernet0/1/0/3.1 l2transport
 encapsulation default
!
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
!
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 2 second-dot1q 3
```

   In this configuration, note that:

   - A QinQ frame with an outer VLAN tag 2 and an inner VLAN tag 3 could match the 1, 2, or 3 subinterfaces but it is assigned to the 3 subinterface because of the longest match rule. Two tags on 3 are longer than one tag on 2 and longer than no tags on 1.
   - A QinQ frame with an outer VLAN tag 2 and an inner VLAN tag 4 is assigned to the .2 subinterface because encapsulation dot1q 2 can match dot1q frames with just the VLAN tag 2 but can also match QinQ frames with an outer tag 2. Refer to Example 5 (the exact keyword) if you do not want to match the QinQ frames.
   - A QinQ frame with an outer VLAN tag 3 matches the 1 subinterface.
   - A dot1q frame with a VLAN tag 2 matches the 2 subinterface.
   - A dot1q frame with a VLAN tag 3 matches the 1 subinterface.

5. To match a dot1q frame and not a QinQ frame, use the exact keyword:

   <#root>

```
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
```

   exact

   This configuration does not match QinQ frames with an outer VLAN tag 2 because it matches only frames with exactly one VLAN tag.

6. Use the untagged keyword in order to match only untagged frames such as Cisco Discovery Protocol (CDP) packets or Multiple Spanning Tree (MST) Bridge Protocol Data Units (BPDUs):

   <#root>

```
interface GigabitEthernet0/1/0/3.1 l2transport
 encapsulation default
!
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation
```

   untagged

```
!
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 3
```

In this configuration, note that:

- Dot1q frames with a VLAN tag 3 or QinQ frames with an outer tag 3 match the 3 subinterfaces.
- All other dot1q or QinQ frames match the 1 subinterface.
- Frames without a VLAN tag match the 2 subinterface.

7. The any keyword can be used as wildcard:

<#root>

```
interface GigabitEthernet0/1/0/3.4 l2transport
 encapsulation dot1q 4 second-dot1q
```

any

```
!
interface GigabitEthernet0/1/0/3.5 l2transport
 encapsulation dot1q 4 second-dot1q 5
```

Both subinterfaces 4 and 5 could match QinQ frames with tags 4 and 5, but the frames are assigned to the 5 subinterfaces because it is more specific. This is the longest match rule.

8. Ranges of VLAN tags can be used:

```
interface GigabitEthernet0/1/0/3.6 l2transport
 encapsulation dot1q 6-10
```

9. Multiple VLAN tag values or ranges can be listed for the first or second dot1q tag:

```
interface GigabitEthernet0/1/0/3.7 l2transport
 encapsulation dot1q 6 , 7 , 8-10
!
interface GigabitEthernet0/1/0/3.11 l2transport
 encapsulation dot1q 11 second-dot1q 1 , 2 , 3 , 4-6 , 10
```

You can list a maximum of nine values. If more values are required, they must be assigned to another subinterface. Group values in a range in order to shorten the list.

10. The **encapsulation dot1q second-dot1q** command uses the Ethertype 0x8100 for the outer and inner tags because this is the Cisco method to encapsulate QinQ frames. According to IEEE, however, the Ethertype 0x8100 can be reserved for 802.1q frames with one VLAN tag, and an outer tag with Ethertype 0x88a8 can be used for QinQ frames. The outer tag with Ethertype 0x88a8 can be configured with the dot1ad keyword:

<#root>

```
interface GigabitEthernet0/1/0/3.12 l2transport
 encapsulation
dot1ad
 12 dot1q 100
```

11. In order to use the old Ethertype 0x9100 or 0x9200 for the QinQ outer tags, use the **dot1q tunneling ethertype** command under the main interface of the QinQ subinterface:

```
<#root>
interface GigabitEthernet0/1/0/3
 dot1q tunneling ethertype [0x9100|0x9200]
!
interface GigabitEthernet0/1/0/3.13 l2transport
 encapsulation dot1q 13 second-dot1q 100
```

The outer tag has an Ethertype of 0x9100 or 0x9200, and the inner tag has the dot1q Ethertype 0x8100.

12. An incoming frame can be assigned to a subinterface, based on the source MAC address:

```
<#root>
interface GigabitEthernet0/1/0/3.14 l2transport
encapsulation dot1q 14 ingress source-mac 1.1.1
```

2.1.2 VLAN Manipulation

The default behavior of an EVC-based platform is to keep the VLAN tags on the incoming frame.

```
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 3
```

In this configuration, an incoming dot1q frame with a VLAN tag 3 keeps its VLAN tag 3 when the frame is forwarded. An incoming QinQ frame with an outer VLAN tag 3 and an inner tag 100 keeps both tags unchanged when the frame is forwarded.

But, the EVC infrastructure allows you to manipulate the tags with the **rewrite** command, so you can pop (remove), translate, or push (add) tags to the incoming VLAN tag stack.

Here are several examples:

- The pop keyword lets you remove a QinQ tag from an incoming dot1q frame. This example removes the outer tag 13 of the incoming QinQ frame and forwards the frame with the dot1q tag 100 on top:

  <#root>

  ```
  interface GigabitEthernet0/1/0/3.13 l2transport
   encapsulation dot1q
  ```

  13

  ```
   second-dot1q
  ```

  100

  ```
   rewrite ingress tag
  ```

  pop

  ```
   1
  ```

  symmetric

  The behavior is always symmetric, which means that the outer tag 13 is popped in the ingress direction and pushed in the egress direction.

- The translate keyword lets you replace one or two incoming tags by one or two new tags:

  <#root>

  ```
  RP/0/RSP0/CPU0:router2(config-subif)#interface GigabitEthernet0/1/0/3.3
     l2transport
  RP/0/RSP0/CPU0:router2(config-subif)# encapsulation dot1q 3
  RP/0/RSP0/CPU0:router2(config-subif)#rewrite ingress tag
  ```

  translate

  ```
   ?
    1-to-1  Replace the outermost tag with another tag
    1-to-2  Replace the outermost tag with two tags
    2-to-1  Replace the outermost two tags with one tag
    2-to-2  Replace the outermost two tags with two other tags
  RP/0/RSP0/CPU0:router2(config-subif)#rewrite ingress tag
  ```

  translate

  ```
   1-to-1 ?
    dot1ad  Push a Dot1ad tag
    dot1q   Push a Dot1Q tag
  RP/0/RSP0/CPU0:router2(config-subif)#rewrite ingress tag
  ```

  translate

  ```
   1-to-1
     dot1q 4
  RP/0/RSP0/CPU0:router2(config-subif)#show config
  Building configuration...
  ```

```
!! IOS XR Configuration 4.3.0
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag
translate
 1-to-1 dot1q 4
symmetric

!
end
```

The symmetric keyword is added automatically because it is the only supported mode.

- The push keyword lets you add a QinQ tag to an incoming dot1q frame:

```
<#root>
interface GigabitEthernet0/1/0/3.4 l2transport
 encapsulation dot1q
4

 rewrite ingress tag
push
 dot1q
100
 symmetric
```

An outer QinQ tag 100 is added to the incoming frame with a dot1q tag 4. In the egress direction, the QinQ tag is popped.

2.2 Cisco IOS® XR Non-EVC Router Behavior (CRS and XR12000)

The syntax for VLAN matching on the non-EVC platforms does not use the encapsulation keyword:

```
RP/0/RP0/CPU0:router1#config
RP/0/RP0/CPU0:router1(config)#int gig 0/0/0/2.3 l2transport
RP/0/RP0/CPU0:router1(config-subif)#dot1q ?
  vlan  Configure a VLAN ID on the subinterface
RP/0/RP0/CPU0:router1(config-subif)#dot1q vlan ?
  <1-4094>  Configure first (outer) VLAN ID on the subinterface
RP/0/RP0/CPU0:router1(config-subif)#dot1q vlan 3 ?
  <1-4094>  Configure second (inner 802.1Q) VLAN ID on the subinterface
  any       Match frames with any second 802.1Q VLAN ID
```

```
RP/0/RP0/CPU0:router1(config-subif)#dot1q vlan 3 100
```

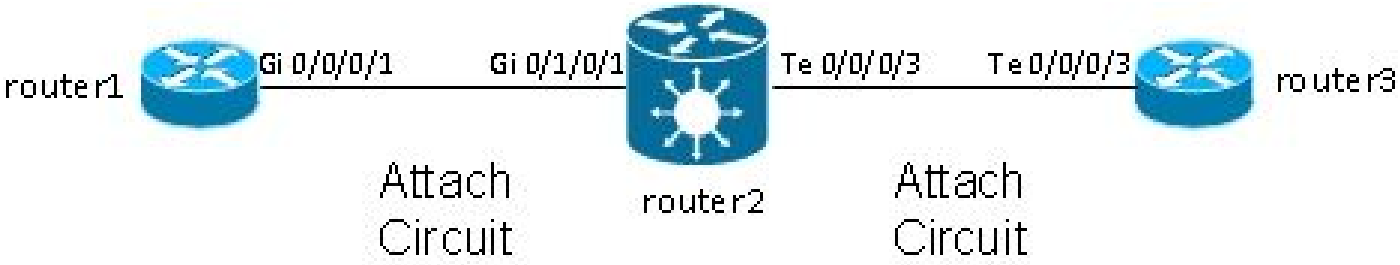
VLAN tag manipulation cannot be configured, because the only possible behavior is to pop all the tags that are specified in the **dot1q** or **dot1ad** commands. This is done by default, so there is no **rewrite** command.

3. Point-to-Point Service

3.1 Local Switching

3.1.1 Main Interface

The basic topology is a local cross connect between two main interfaces:



Router2 takes all traffic received on Gi 0/1/0/1 and forwards it to Te 0/0/0/3 and vice versa.

While router1 and router3 appear to have a direct back-to-back cable in this topology, this is not the case because router2 is actually translating between the TenGigE and GigabitEthernet interfaces. Router2 can run features on these two interfaces; an access control list (ACL), for example, can drop specific types of packets or a policy-map in order to shape or rate-limit low priority traffic.

A basic point-to-point cross connect is configured between two main interfaces that are configured as l2transport on router2:

<#root>

```
interface GigabitEthernet0/1/0/1
```

`l2transport`

```
 !
!
interface TenGigE0/0/0/3
```

`l2transport`

```
 !
```

```
!
l2vpn
 xconnect group test
  p2p p2p1
   interface TenGigE0/0/0/3
   interface GigabitEthernet0/1/0/1
  !
```

On router1 and router3, the main interfaces are configured with CDP and an IPv4 address:

<#root>

```
RP/0/RP0/CPU0:router1#sh run int Gi 0/0/0/1
interface GigabitEthernet0/0/0/1
```

cdp

ipv4

```
 address 10.1.1.1 255.255.255.0
!
```

```
RP/0/RP0/CPU0:
```

router1

```
#
RP/0/RP0/CPU0:
```

router1

```
#sh
```

cdp

```
 nei Gi 0/0/0/1
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater

Device ID       Local Intrfce    Holdtme Capability Platform  Port ID
```

router3

```
.cisco.c Gi0/0/0/1        132     R          ASR9K Ser Te0/0/0/3
RP/0/RP0/CPU0:
```

router1

```
#ping
```

10.1.1.2

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/8/32 ms
```

Router1 sees router3 as a CDP neighbor and can ping 10.1.1.2 (the interface address of router3) as if the two routers were directly connected.

Because there is no subinterface configured on router2, incoming frames with a VLAN tag are transported transparently when dot1q subinterfaces are configured on router1 and router3:

```
<#root>

RP/0/RP0/CPU0:

router1

#sh run int gig 0/0/0/1.2
interface GigabitEthernet0/0/0/1.2
 ipv4 address 10.1.2.1 255.255.255.0


dot1q

 vlan 2
!

RP/0/RP0/CPU0:

router1

#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/5 ms
```

After 10,000 pings from router1 to router3, you can use the **show interface** and **show l2vpn** commands in order to ensure that ping requests received by router2 on one AC are forwarded on the other AC and that ping replies are handled the same way in reverse.

```
<#root>

RP/0/RSP0/CPU0:router2#

sh int

 gig 0/1/0/1
GigabitEthernet0/1/0/1 is up, line protocol is up
  Interface state transitions: 1
  Hardware is GigabitEthernet, address is 0024.986c.63f1 (bia 0024.986c.63f1)
  Description: static lab connection to acdc 0/0/0/1 - dont change
  Layer 2 Transport Mode
  MTU 1514 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 1000Mb/s, SXFD, link type is force-up
  output flow control is off, input flow control is off
  loopback not set,
  Last input 00:00:00, output 00:00:00
  Last clearing of "show interface" counters 00:01:07
```

```
  5 minute input rate 28000 bits/sec, 32 packets/sec
  5 minute output rate 28000 bits/sec, 32 packets/sec
     10006 packets input, 1140592 bytes, 0 total input drops
     0 drops for unrecognized upper-level protocol
     Received 0 broadcast packets, 6 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     10007 packets output, 1140832 bytes, 0 total output drops
     Output 0 broadcast packets, 7 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions


RP/0/RSP0/CPU0:router2#
```

sh int

```
 ten 0/0/0/3
TenGigE0/0/0/3 is up, line protocol is up
  Interface state transitions: 3
  Hardware is TenGigE, address is 0024.98ea.038b (bia 0024.98ea.038b)
  Layer 1 Transport Mode is LAN
  Description: static lab connection to putin 0/0/0/3 - dont change
  Layer 2 Transport Mode
  MTU 1514 bytes, BW 10000000 Kbit (Max: 10000000 Kbit)
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 10000Mb/s, LR, link type is force-up
  output flow control is off, input flow control is off
  loopback not set,
  Last input 00:00:00, output 00:00:06
  Last clearing of "show interface" counters 00:01:15
  5 minute input rate 27000 bits/sec, 30 packets/sec
  5 minute output rate 27000 bits/sec, 30 packets/sec
     10008 packets input, 1140908 bytes, 0 total input drops
     0 drops for unrecognized upper-level protocol
     Received 0 broadcast packets, 8 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     10006 packets output, 1140592 bytes, 0 total output drops
     Output 0 broadcast packets, 6 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions


RP/0/RSP0/CPU0:router2#
```

sh l2vpn xconnect group test

```
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect               Segment 1                   Segment 2
Group      Name    ST   Description         ST       Description         ST
---------------------   -------------------------   -------------------------
test       p2p1    UP   Te0/0/0/3           UP       Gi0/1/0/1           UP
------------------------------------------------------------------------------
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect group test det

Group test, XC p2p1, state is up; Interworking none
```

```
  AC: TenGigE0/0/0/3, state is up
    Type Ethernet
    MTU 1500; XC ID 0x1080001; interworking none
    Statistics:
      packets: received 10008, sent 10006
      bytes: received 1140908, sent 1140592
  AC: GigabitEthernet0/1/0/1, state is up
    Type Ethernet
    MTU 1500; XC ID 0x1880003; interworking none
    Statistics:
      packets: received 10006, sent 10008
      bytes: received 1140592, sent 1140908


RP/0/RSP0/CPU0:router2#
```

sh l2vpn forwarding interface gigabitEthernet 0/1/0/1

hardware ingress detail location 0/1/CPU0

```
Local interface: GigabitEthernet0/1/0/1, Xconnect id: 0x1880003, Status: up
  Segment 1
    AC, GigabitEthernet0/1/0/1, Ethernet port mode, status: Bound
    Statistics:
      packets: received 10022, sent 10023
      bytes: received 1142216, sent 1142489
      packets dropped: PLU 0, tail 0
      bytes dropped: PLU 0, tail 0
  Segment 2
    AC, TenGigE0/0/0/3, Ethernet port mode, status: Bound

  Platform AC context:
  Ingress AC: Local Switch, State: Bound
    Flags: Remote is Simple AC
  XID: 0x00580003, SHG: None
  Ingress uIDB: 0x0003, Egress uIDB: 0x0003, NP: 3, Port Learn Key: 0
  NP3
    Ingress uIDB:
      Flags: L2, Status
      Stats Ptr: 0x0d842c, uIDB index: 0x0003, Wire Exp Tag: 0
      BVI Bridge Domain: 0, BVI Source XID: 0x01000000
      VLAN1: 0, VLAN1 etype: 0x0000, VLAN2: 0, VLAN2 etype: 0x0000
      L2 ACL Format: 0, L2 ACL ID: 0, IPV4 ACL ID: 0, IPV6 ACL ID: 0
      QOS ID: 0, QOS Format ID: 0
      Local Switch dest XID: 0x00000001
      UIDB IF Handle: 0x00000000, Source Port: 1, Num VLANs: 0
    Xconnect ID: 0x00580003, NP: 3
      Type: AC, Remote type: AC
      Flags: Learn enable
      uIDB Index: 0x0003, LAG pointer: 0x0000
      Split Horizon Group: None

RP/0/RSP0/CPU0:router2#
```

sh l2vpn forwarding interface Te 0/0/0/3 hardware egress

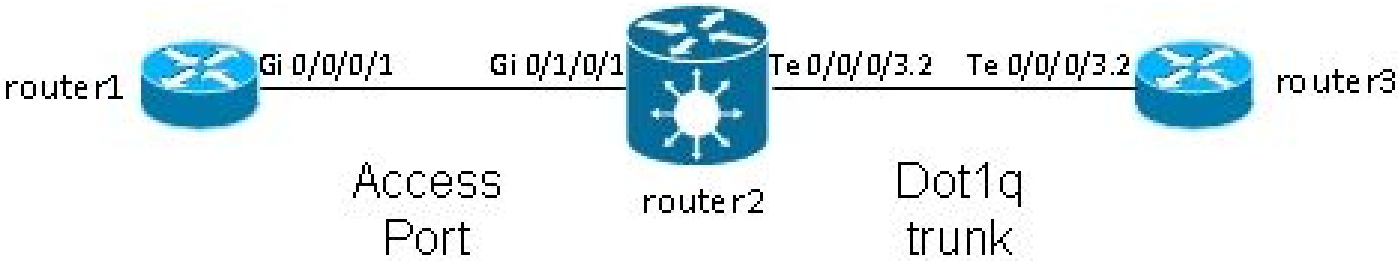
detail location 0/0/CPU0

```
Local interface: TenGigE0/0/0/3, Xconnect id: 0x1080001, Status: up
  Segment 1
    AC, TenGigE0/0/0/3, Ethernet port mode, status: Bound
    Statistics:
      packets: received 10028, sent 10027
      bytes: received 1143016, sent 1142732
      packets dropped: PLU 0, tail 0
      bytes dropped: PLU 0, tail 0
  Segment 2
    AC, GigabitEthernet0/1/0/1, Ethernet port mode, status: Bound

  Platform AC context:
  Egress AC: Local Switch, State: Bound
    Flags: Remote is Simple AC
  XID: 0x00000001, SHG: None
  Ingress uIDB: 0x0007, Egress uIDB: 0x0007, NP: 0, Port Learn Key: 0
  NP0
    Egress uIDB:
      Flags: L2, Status, Done
      Stats ptr: 0x000000
      VPLS SHG: None
      L2 ACL Format: 0, L2 ACL ID: 0, IPV4 ACL ID: 0, IPV6 ACL ID: 0
      VLAN1: 0, VLAN1 etype: 0x0000, VLAN2: 0, VLAN2 etype: 0x0000
      UIDB IF Handle: 0x04000240, Search VLAN Vector: 0
      QOS ID: 0, QOS format: 0
    Xconnect ID: 0x00000001, NP: 0
      Type: AC, Remote type: AC
      Flags: Learn enable
      uIDB Index: 0x0007, LAG pointer: 0x0000
      Split Horizon Group: None
```

3.1.2 Subinterfaces and VLAN Manipulation

In Cisco IOS® software terminology, this example has one AC that is like a switchport mode access interface and a dot1q subinterface that is like a trunk:



Typically this topology uses a bridge-domain because there are usually more than two ports in the VLAN, although you can use a point-to-point cross connect if there are only two ports. This section describes how flexible rewrite capabilities give you multiple ways to manipulate the VLAN.

3.1.2.1 Main Interface and Dot1q Subinterface

In this example, the main interface is on one side, and the dot1q subinterface is on the other side:

This is the main interface on router1:

```
RP/0/RP0/CPU0:router1#sh run int gig 0/0/0/1
interface GigabitEthernet0/0/0/1
 description static lab connection to router2 0/1/0/1
 cdp
 ipv4 address 10.1.1.1 255.255.255.0
!
```

This is the dot1q subinterface on router2:

```
RP/0/RSP0/CPU0:router2#sh run int gig 0/1/0/1
interface GigabitEthernet0/1/0/1
 description static lab connection to router1 0/0/0/1
 l2transport

RP/0/RSP0/CPU0:router2#sh run int ten 0/0/0/3.2
interface TenGigE0/0/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric

RP/0/RSP0/CPU0:router2#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p2
   interface TenGigE0/0/0/3.2
   interface GigabitEthernet0/1/0/1
```

There is now an l2transport keyword in the subinterface name of TenGigE0/0/0/3.2. Router3 sends dot1q frames with tag 2, which match the TenGigE0/0/0/3.2 subinterface on router2.

The incoming tag 2 is removed in the ingress direction by the **rewrite ingress tag pop 1 symmetric** command. Since the tag has been removed in the ingress direction on the TenGigE0/0/0/3.2, the packets are sent untagged in the egress direction on GigabitEthernet0/1/0/1.

Router1 sends untagged frames, which match the main interface GigabitEthernet0/1/0/1.

There is no **rewrite** command on GigabitEthernet0/1/0/1, so no tag is popped, pushed, or translated.

When packets have to be forwarded out of TenGigE0/0/0/3.2, the dot1q tag 2 is pushed due to the symmetric keyword in the **rewrite ingress tag pop 1** command. The command pops one tag in the ingress direction but symmetrically pushes one tag in the egress direction. This is an example on router3:

```
RP/0/RSP0/CPU0:router3#sh run int ten 0/0/0/3.2
interface TenGigE0/0/0/3.2
 ipv4 address 10.1.1.2 255.255.255.0
 encapsulation dot1q 2
```

Monitor the subinterface counters with the same **show interface** and **show l2vpn** commands:

<#root>

```
RP/0/RSP0/CPU0:router2#clear counters
Clear "show interface" counters on all interfaces [confirm]
RP/0/RSP0/CPU0:router2#clear l2vpn forwarding counters
RP/0/RSP0/CPU0:router2#
RP/0/RSP0/CPU0:router2#
RP/0/RSP0/CPU0:router2#sh int TenGigE0/0/0/3.2
TenGigE0/0/0/3.2 is up, line protocol is up
  Interface state transitions: 1
  Hardware is VLAN sub-interface(s), address is 0024.98ea.038b
  Layer 2 Transport Mode
  MTU 1518 bytes, BW 10000000 Kbit (Max: 10000000 Kbit)
     reliability Unknown, txload Unknown, rxload Unknown
  Encapsulation 802.1Q Virtual LAN,
    Outer Match: Dot1Q VLAN 2
    Ethertype Any, MAC Match src any, dest any
  loopback not set,
  Last input 00:00:00, output 00:00:00
  Last clearing of "show interface" counters 00:00:27
```

1000 packets input

```
, 122000 bytes
     0 input drops, 0 queue drops, 0 input errors
```

1002 packets output

```
, 122326 bytes
     0 output drops, 0 queue drops, 0 output errors
```

```
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect detail

Group test, XC p2p2, state is up; Interworking none
  AC: TenGigE0/0/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1500; XC ID 0x1080001; interworking none
    Statistics:
```

packets: received 1001

```
,
```

sent 1002

```
      bytes: received 118080, sent 118318
```

```
      drops: illegal VLAN 0, illegal length 0
  AC: GigabitEthernet0/1/0/1, state is up
    Type Ethernet
    MTU 1500; XC ID 0x1880003; interworking none
    Statistics:

packets: received 1002, sent 1001

      bytes: received 114310, sent 114076
```

As expected, the number of packets received on TenGigE0/0/0/3.2 matches the number of packets sent on GigabitEthernet0/1/0/1 and vice versa.

3.1.2.2 Subinterface with Encapsulation

Instead of the main interface on GigabitEthernet0/1/0/1, you can use a subinterface with encapsulation default in order to catch all frames or with encapsulation untagged in order to match only untagged frames:

<#root>

```
RP/0/RSP0/CPU0:router2#sh run interface GigabitEthernet0/1/0/1.1
interface GigabitEthernet0/1/0/1.1 l2transport

encapsulation untagged
```

```
RP/0/RSP0/CPU0:router2#sh run int TenGigE0/0/0/3.2
interface TenGigE0/0/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric

RP/0/RSP0/CPU0:router2#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p3
   interface TenGigE0/0/0/3.2
   interface GigabitEthernet0/1/0/1.1
```

3.1.2.3 Ingress Direction on GigabitEthernet0/1/0/1.1

Rather than pop tag 2 in the ingress direction on TenGigE0/0/0/3.2, you can push tag 2 in the ingress direction on GigabitEthernet0/1/0/1.1 and not do anything on TenGigE0/0/0/3.2:

<#root>

```
RP/0/RSP0/CPU0:router2#sh run int  TenGigE0/0/0/3.2
interface TenGigE0/0/0/3.2 l2transport
 encapsulation dot1q 2

RP/0/RSP0/CPU0:router2#sh run interface GigabitEthernet0/1/0/1.1
interface GigabitEthernet0/1/0/1.1 l2transport
 encapsulation untagged


rewrite ingress tag push dot1q 2 symmetric



RP/0/RSP0/CPU0:router2#sh run int TenGigE0/0/0/3.2
interface TenGigE0/0/0/3.2 l2transport
 encapsulation dot1q 2

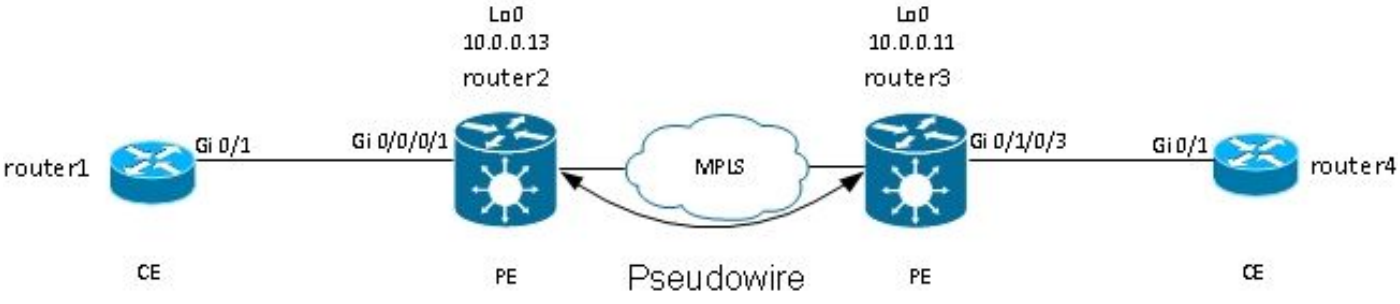
RP/0/RSP0/CPU0:router2#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p3
   interface TenGigE0/0/0/3.2
   interface GigabitEthernet0/1/0/1.1
```

Thus, you can see that the EVC model with the **encapsulation** and **rewrite** commands gives you great flexibility to match and manipulate VLAN tags.

3.2 Virtual Private Wire Services

3.2.1 Overview

Virtual Private Wire Services (VPWS), also known as Ethernet over MPLS (EoMPLS), allow two L2VPN Provider Edge (PE) devices to tunnel the L2VPN traffic over an MPLS cloud. The two L2VPN PEs are typically connected at two different sites with an MPLS core between them. The two ACs connected at each L2VPN PE are linked by a PW over the MPLS network, which is the MPLS PW.



Each PE needs to have an MPLS label in order to reach the loopback of the remote PE. This label, usually called the Interior Gateway Protocol (IGP) label, can be learned through the MPLS Label Distribution Protocol (LDP) or MPLS Traffic Engineering (TE).

The two PEs establish a targeted MPLS LDP session between themselves so they can establish and control the status of the PW. One PE advertises to the other PE the MPLS label for PW identification.

Note: While BGP can be used for signaling, it is not covered in this document.

The traffic received by router2 on its local AC is encapsulated in an MPLS label stack:

- The outer MPLS label is the IGP label to reach the loopback of router3. This could be the implicit-null label if the labels are directly connected; this means that no IGP label would be appended.
- The inner MPLS label is the PW label advertised by router3 through the targeted LDP session.
- There can be a PW control word after the MPLS labels, depending on the configuration and the type of encapsulation. The control word is not used by default on Ethernet interfaces and must be explicitly configured when needed.
- The transported L2 frame follows in the packet.
- Some VLAN tags are transported over the PW, depending on the configuration and the PW type.

The penultimate hop, just before router3 in the MPLS core, pops the IGP label or replaces it with an explicit null label. Thus, the top meaningful label on the frame received by router3 is the PW label that router3 signaled to router2 for the PW. So, router3 knows that traffic received with that MPLS label can be switched to the AC connected to router4.

In the previous example, you can first check whether each L2VPN has an MPLS label for the loopback of the remote PE. This is an example of how to check labels on router2:

```
RP/0/RSP1/CPU0:router2#sh mpls forwarding prefix 10.0.0.11/32
Local  Outgoing    Prefix             Outgoing     Next Hop        Bytes
Label  Label       or ID              Interface                    Switched
------ ----------- ------------------ ------------ --------------- ------------
16008  16009       10.0.0.11/32       Te0/0/0/1    10.0.23.2       681260
```

The AC configuration is still the same:

```
RP/0/RSP1/CPU0:router2#sh run int gig 0/0/0/1.2
Wed May  1 13:56:07.668 CEST
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
```

Because there is no **rewrite ingress pop** command, the incoming VLAN tag 2 is transported over the PW. See Type 4 and 5 PWs for details.

The L2VPN configuration specifies the local AC and the remote L2VPN PE with a PW ID that must match on each side and must be unique for each neighbor:

```
RP/0/RSP1/CPU0:router2#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p4
   interface GigabitEthernet0/0/0/1.2
```

neighbor 10.0.0.11 pw-id 222

The corresponding configuration on router3 is:

<#root>

```
RP/0/RSP0/CPU0:router3#sh run int gig 0/1/0/3.2
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
!

RP/0/RSP0/CPU0:router3#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p4
   interface GigabitEthernet0/1/0/3.2
```

neighbor 10.0.0.13 pw-id 222

Use the **show l2vpn xconnect detail** command in order to view details on the cross connect:

<#root>

```
RP/0/RSP1/CPU0:router2#
```

sh l2vpn xconnect group test xc-name p2p4 detail

```
Group test, XC p2p4, state is up; Interworking none
```

AC: GigabitEthernet0/0/0/1.2

```
, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
```

MTU 1504

```
; XC ID 0x840006; interworking none
    Statistics:
      packets: received
```

186

```
, sent
```

38448

```
      bytes: received 12644, sent 2614356
      drops: illegal VLAN 0, illegal length 0
  PW: neighbor 10.0.0.11, PW ID 222, state is up ( established )
    PW class not set, XC ID 0xc0000004
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.13
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ ------------------------------
      Label
```

16026

```
                        16031
      Group ID     0x4000280                      0x6000180
      Interface
```

GigabitEthernet0/0/0/1.2

```
       GigabitEthernet0/1/0/3.2
      MTU          1504                           1504
      Control word disabled                       disabled
      PW type      Ethernet                       Ethernet
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x6                            0x6
                   (router alert label)           (router alert label)
                   (TTL expiry)                   (TTL expiry)
      ------------ ------------------------------ ------------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Outgoing Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    MIB cpwVcIndex: 3221225476
    Create time: 30/04/2013 16:30:58 (21:31:00 ago)
    Last time status changed: 30/04/2013 16:36:42 (21:25:16 ago)
    Statistics:
      packets: received
```

38448

```
, sent
```

186

```

      bytes: received 2614356, sent 12644
```

In this configuration, note that:

- The maximum transmission unit (MTU) of the AC is 1504 because the incoming tag on the AC is not popped. The MTU must match on each side, or the PW does not come up.
- 186 packets were received on the AC and were sent on the PW as expected.

- 38448 packets were received on the PW and were sent on the AC as expected.
- The local label on router2 is 16026 and is the label that router3 uses as the inner label. The packets are received on router2 with that MPLS label as the top label because the IGP label has been popped by the penultimate MPLS hop. Router2 knows that incoming frames with that PW label can be switched to the AC Gi 0/0/0/1.2:

<#root>

```
RP/0/RSP1/CPU0:router2#sh mpls forwarding labels 16026
Local  Outgoing    Prefix             Outgoing     Next Hop        Bytes
Label  Label       or ID              Interface                    Switched
------ ----------- ------------------ ------------ --------------- ------------

16026

  Pop         PW(10.0.0.11:222)

Gi0/0/0/1.2

  point2point     2620952
```

3.2.2 PW and AC Coupled Status

In a point-to-point cross connect, the AC and the PW are coupled. So, if the AC goes down, the L2VPN PE signals via LDP to the remote PE that the PW status can be down. This triggers convergence when PW redundancy is configured. See the Redundancy section for details.

In this example, the AC is down on router2 and is sending the AC Down PW status to router3:

<#root>

```
RP/0/RSP1/CPU0:router2#sh l2vpn xconnect group test xc-name p2p4 detail
Wed May  1 23:38:55.542 CEST

Group test, XC p2p4, state is down; Interworking none
  AC: GigabitEthernet0/0/0/1.2, state is down
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1504; XC ID 0x840006; interworking none
    Statistics:
      packets: received 186, sent 38544
      bytes: received 12644, sent 2620884
      drops: illegal VLAN 0, illegal length 0
  PW: neighbor 10.0.0.11, PW ID 222, state is down ( remote standby )
    PW class not set, XC ID 0xc0000004
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.13
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
```

```
      ------------ ------------------------------ ------------------------------
      Label        16026                          16031
      Group ID     0x4000280                      0x6000180
      Interface    GigabitEthernet0/0/0/1.2       GigabitEthernet0/1/0/3.2
      MTU          1504                           1504
      Control word disabled                       disabled
      PW type      Ethernet                       Ethernet
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x6                            0x6
                   (router alert label)           (router alert label)
                   (TTL expiry)                   (TTL expiry)
      ------------ ------------------------------ ------------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Outgoing Status (PW Status TLV):
      Status code: 0x6 (
```

AC Down

```
) in Notification message
    MIB cpwVcIndex: 3221225476
    Create time: 30/04/2013 16:30:58 (1d07h ago)
    Last time status changed: 01/05/2013 14:05:07 (09:33:47 ago)
    Statistics:
      packets: received 38544, sent 186
      bytes: received 2620884, sent 12644
```

Router3 knows that the PW can be down because the remote AC is down:

<#root>

```
RP/0/RSP0/CPU0:router3#sh l2vpn xconnect group test xc-name p2p4 detail

Group test, XC p2p4, state is down; Interworking none
  AC: GigabitEthernet0/1/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1504; XC ID 0xc40003; interworking none
    Statistics:
      packets: received 38545, sent 186
      bytes: received 2620952, sent 12644
      drops: illegal VLAN 0, illegal length 0
```

PW: neighbor 10.0.0.13, PW ID 222, state is down (local ready)

```
    PW class not set, XC ID 0xc0000005
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.11
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ ------------------------------
      Label        16031                          16026
```

```
  Group ID     0x6000180                      0x4000280
  Interface    GigabitEthernet0/1/0/3.2       GigabitEthernet0/0/0/1.2
  MTU          1504                           1504
  Control word disabled                       disabled
  PW type      Ethernet                       Ethernet
  VCCV CV type 0x2                            0x2
               (LSP ping verification)        (LSP ping verification)
  VCCV CC type 0x6                            0x6
               (router alert label)           (router alert label)
               (TTL expiry)                   (TTL expiry)
  ------------ ------------------------------ ------------------------------
Incoming Status (PW Status TLV):
```

Status code: 0x6 (AC Down) in Notification message

```
Outgoing Status (PW Status TLV):
  Status code: 0x0 (Up) in Notification message
MIB cpwVcIndex: 3221225477
Create time: 30/04/2013 16:37:57 (1d07h ago)
Last time status changed: 01/05/2013 14:11:33 (09:35:50 ago)
Statistics:
  packets: received 186, sent 38545
  bytes: received 12644, sent 2620952
```

3.2.3 Type 4 and Type 5 PWs

Two types of PWs can be used - type 4 and type 5.

- A type 4 PW is known as a VLAN-based PW. The ingress PE is not supposed to remove the incoming VLAN tags that are to be transported over the PW.

  On the EVC-based platforms such as the ASR 9000, the problem is that the incoming ACs can have a **rewrite** command that pops the incoming VLAN tags, so there can not be any VLAN tag to be transported over the PW. In order to address this possibility, the EVC platforms insert a dummy VLAN tag 0 on top of the frame for type 4 PWs. Type 4 PWs are configured with the **transport-mode vlan** command. The remote PE can be EVC-based and can understand that the top VLAN tag is the dummy tag to be stripped.

  However, if you use a type 4 PW between an EVC platform and a non-EVC platform, this can lead to interoperability problems. The non-EVC platform does not consider the top VLAN tag as the dummy VLAN tag and instead forwards the frame with the dummy VLAN tag 0 as the outer tag. The EVC platforms have the ability to manipulate the VLAN tags received on the incoming frame with the **rewrite** command. The results of that VLAN manipulation are transported over the type 4 PW with the extra dummy tag 0 on top.

  Recent Cisco IOS XR software releases offer the ability to use a type 4 PW without use of the dummy tag 0 with the **transport-mode vlan passthrough** command. The VLAN tag manipulation on the Ethernet Flow Point (EFP) must ensure that at least one tag remains because there must be a VLAN tag transported on a type 4 PW and because, in this case, there is no dummy tag that meets that requirement. The tags that remain on the frame after the incoming interface tag rewrite are transported transparently through the PW.

- A type 5 PW is known as an Ethernet port-based PW. The ingress PE transports frames received on a main interface or after the subinterface tags have been removed when the packet is received on a subinterface. There is no requirement to send a tagged frame over a type 5 PW, and no dummy tag is added by the EVC-based platforms. The EVC-based platforms have the ability to manipulate the VLAN tags received on the incoming frame with the **rewrite** command. The results of that VLAN manipulation are transported over the type 5 PW, whether tagged or untagged.

By default, the L2VPN PEs try to negotiate a type 5 PW, as seen in this example:

<#root>

```
RP/0/RSP1/CPU0:router2#sh l2vpn xconnect group test det | i " PW type"
```

PW type Ethernet

```
, control word disabled, interworking none
      PW type      Ethernet                        Ethernet
```

The PW type Ethernet indicates a type 5 PW.

This is a sniffer capture of an ARP request sent by router1 and encapsulated by router2 over the PW to router3:

<#root>

```
Frame 38: 82 bytes on wire (656 bits), 82 bytes captured (656 bits)
Ethernet II, Src: Cisco_2f:dc:04 (00:0b:60:2f:dc:04), Dst: Cisco_1e:93:50
   (00:24:f7:1e:93:50)
MultiProtocol Label Switching Header, Label:
```

16031

```
, Exp: 0, S: 1, TTL: 251
Ethernet II, Src: Cisco_03:1f:46 (00:1d:46:03:1f:46), Dst: Broadcast
   (ff:ff:ff:ff:ff:ff)
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 2
Address Resolution Protocol (request)
```

The MPLS label 16031 is the PW label advertised by router3. The sniffer capture has been taken between the penultimate hop and router3, so there is no IGP label.

The encapsulated Ethernet frame starts immediately after the PW label. There can be a PW control word, but it is not configured in this example.

Even if it is a type 5 PW, the incoming VLAN tag 2 received on the AC by router2 is transported because there is no **rewrite** command that pops it on the AC. The results that come from the AC after the rewrite processing are transported because there is no automatic tag popping on the EVC-based platforms. Notice

that there is no dummy VLAN tag 0 with a type 5 PW.

If you configured with the **rewrite ingress tag pop 1 symmetric** command, there would be no VLAN tag transported over the PW.

Here is an example of a type 4 PW with configuration of a pw-class on router2 and router3.

Note: If you configure a type 4 on one side only, the PW stays down and reports **Error: PW type mismatched**.

<#root>

```
l2vpn
```

pw-class

```
 VLAN
  encapsulation mpls
   transport-mode vlan
  !
 !
 xconnect group test
  p2p p2p4
   neighbor 10.0.0.11 pw-id 222
```

pw-class

```
 VLAN
   !
  !
 !
!
```

The PW type Ethernet VLAN indicates a type 4 PW.

<#root>

```
RP/0/RSP1/CPU0:router2#sh l2vpn xconnect group test det | i " PW type"
```

PW type Ethernet VLAN

```
, control word disabled, interworking none
      PW type      Ethernet VLAN                  Ethernet VLAN
```

There is now a dummy tag 0 inserted on top of the frame being transported:

<#root>

```
Frame 15: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)
Ethernet II, Src: Cisco_2f:dc:04 (00:0b:60:2f:dc:04), Dst: Cisco_1e:93:50
   (00:24:f7:1e:93:50)
MultiProtocol Label Switching Header, Label: 16031, Exp: 0, S: 1, TTL: 251
Ethernet II, Src: Cisco_03:1f:46 (00:1d:46:03:1f:46), Dst: Broadcast
   (ff:ff:ff:ff:ff:ff)
```

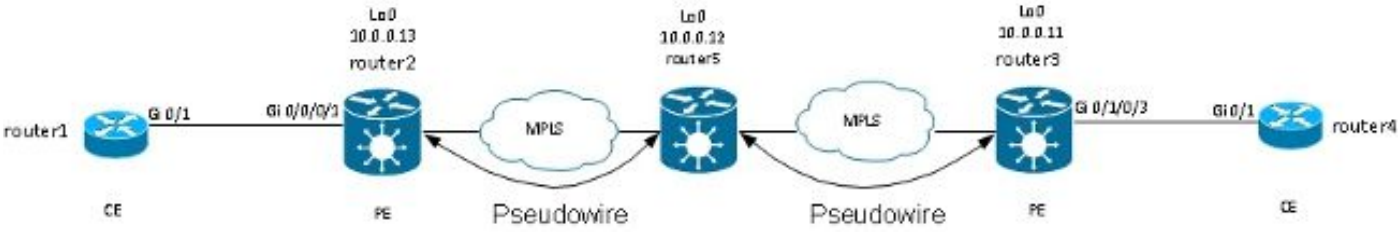
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 0

```
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 2
Address Resolution Protocol (request)
```

The egress EVC-based PE removes the dummy tag and forwards the frame with the tag 2 on its local AC. The egress PE applies the local tag manipulation configured on its AC on the frame received on the PW. If its local AC is configured as **rewrite ingress tag pop 1 symmetric**, the configured tag must be pushed in the egress direction, so a new tag is pushed on top of the tag 2 received on the PW. The rewrite command is very flexible but you can carefully evaluate what you want to achieve at each side of the PW.

3.2.4 Multi-segment PW

It is possible to have an L2VPN PE that has a PW, instead of a physical interface, as an AC:



Router5 receives packets on the PW from router2 and switches the packets on its other PW to router3. So router5 is switching between PWs in order to create a multisegment PW between router2 and router3.

The configuration on router2 now points at router5 as the remote PE:

```
RP/0/RSP1/CPU0:router2#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p5
   interface GigabitEthernet0/0/0/1.2
   neighbor 10.0.0.12 pw-id 222
   !
  !
 !
!
```

The configuration on router5 is basic:

<#root>

```
RP/0/RSP0/CPU0:router5#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p5
   neighbor 10.0.0.11 pw-id 223
   !
   neighbor 10.0.0.13 pw-id 222
   !
```

description R2-R5-R3

```
  !
 !
!
```

The **description** command is optional and is inserted in a PW switching Type Length Value (TLV) that is sent by router5 to each remote PE (router2 and router3). The **description** is useful when you need to troubleshoot a PW problem when there is a router in the middle that does PW switching.

Enter the **sh l2vpn xconnect** command in order to review the PW switching TLV:

<#root>

```
RP/0/RSP0/CPU0:router5#
```

sh l2vpn xconnect group test det

```
Group test, XC p2p5, state is down; Interworking none
  Description: R2-R5-R3
  PW: neighbor 10.0.0.11, PW ID 223, state is down ( provisioned )
    PW class not set, XC ID 0xc0000002
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.12
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ ------------------------------
      Label        16042                          unknown
      Group ID     0x4000280                      0x0
      Interface    GigabitEthernet0/0/0/1.2       unknown
      MTU          1504                           unknown
      Control word disabled                       unknown
      PW type      Ethernet                       unknown
      VCCV CV type 0x2                            0x0
                                                  (none)
                   (LSP ping verification)
      VCCV CC type 0x4                            0x0
                                                  (none)
                   (TTL expiry)
```

```
      ------------ ------------------------------ ------------------------------
    Outgoing PW Switching TLVs (Label Mapping message):
      Local IP Address: 10.0.0.12, Remote IP Address: 10.0.0.13, PW ID: 222
```

Description: R1-R5-R3

```
    Outgoing Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Statistics for MS-PW:
      packets: received 0
      bytes: received 0
    MIB cpwVcIndex: 3221225474
    Create time: 02/05/2013 15:37:53 (00:34:43 ago)
    Last time status changed: 02/05/2013 16:12:30 (00:00:06 ago)
    Last time PW went down: 02/05/2013 16:12:30 (00:00:06 ago)
  PW: neighbor 10.0.0.13, PW ID 222, state is up ( established )
    PW class not set, XC ID 0xc0000001
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.12
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ ------------------------------
      Label        16043                          16056
      Group ID     0x6000180                      0x4000280
      Interface    GigabitEthernet0/1/0/3.2       GigabitEthernet0/0/0/1.2
      MTU          1504                           1504
      Control word disabled                       disabled
      PW type      Ethernet                       Ethernet
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x4                            0x6
                                                  (router alert label)
                   (TTL expiry)                   (TTL expiry)
      ------------ ------------------------------ ------------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Outgoing PW Switching TLVs (Label Mapping message):
      Local IP Address: 10.0.0.12, Remote IP Address: 10.0.0.11, PW ID: 223
```

Description: R2-R5-R3

```
    Outgoing Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Statistics for MS-PW:
      packets: received 0
      bytes: received 0
    MIB cpwVcIndex: 0
    Create time: 02/05/2013 15:37:53 (00:34:43 ago)
    Last time status changed: 02/05/2013 16:12:35 (00:00:01 ago)
    Last time PW went down: 02/05/2013 16:12:30 (00:00:06 ago)
```

Router5 sends a PW switching TLV to router3 with the details of its PW to router2 and sends a PW switching TLV to router2 with the details of its PW to router3.

3.2.5 Redundancy

A point-to-point PW can be used to connect two sites, but these two sites can remain connected in case of a PE or AC failure.

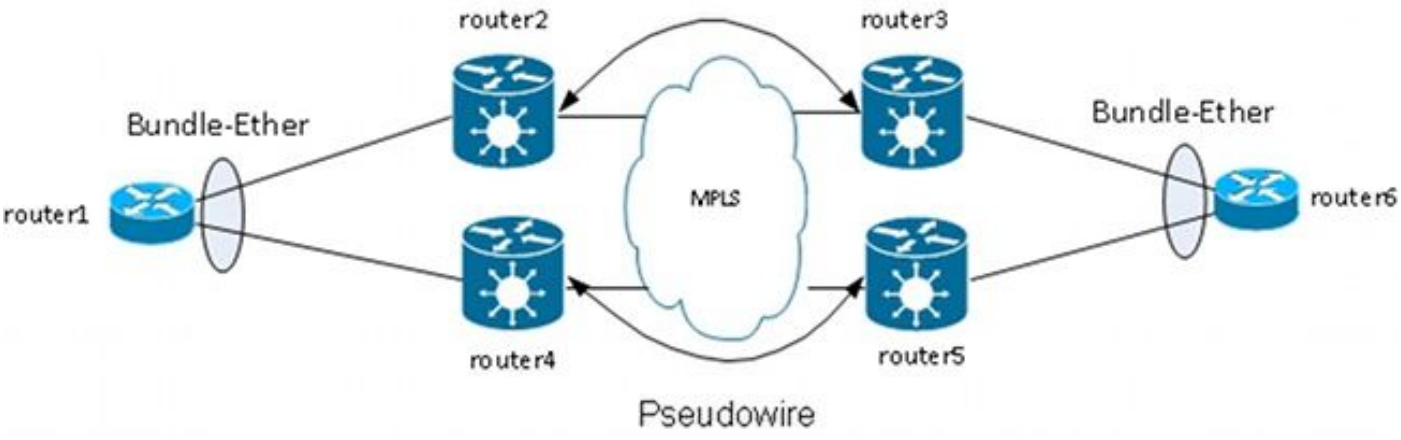
3.2.5.1 Core Redundancy

If you make any topology change that affects rerouting in the MPLS core, the MPLS PW inherits the new path immediately.

3.2.5.2 Bundle Over PWs

A Customer Edge (CE) device can be connected to the PE through an Ethernet bundle in order to provide link redundancy if there is a bundle member link failure between the CE and the PE. The bundle remains up even if one bundle link member goes down. This does not provide PE redundancy because a PE failure brings the entire bundle down.

One method for redundancy is to have multiple circuits transported by point-to-point PWs. Each circuit is a member of an Ethernet bundle between two CEs:



The PE does not terminate the bundle and instead transports frames transparently over the PW, including the Link Aggregation Control Protocol (LACP) frames that CEs exchange between them.

With this design, the loss of an AC or a PE causes a bundle member goes down, but the bundle remains up.

Note: LACP BPDUs were not transported over L2VPN by the ASR 9000 in releases earlier than Cisco IOS XR Software Release 4.2.1.

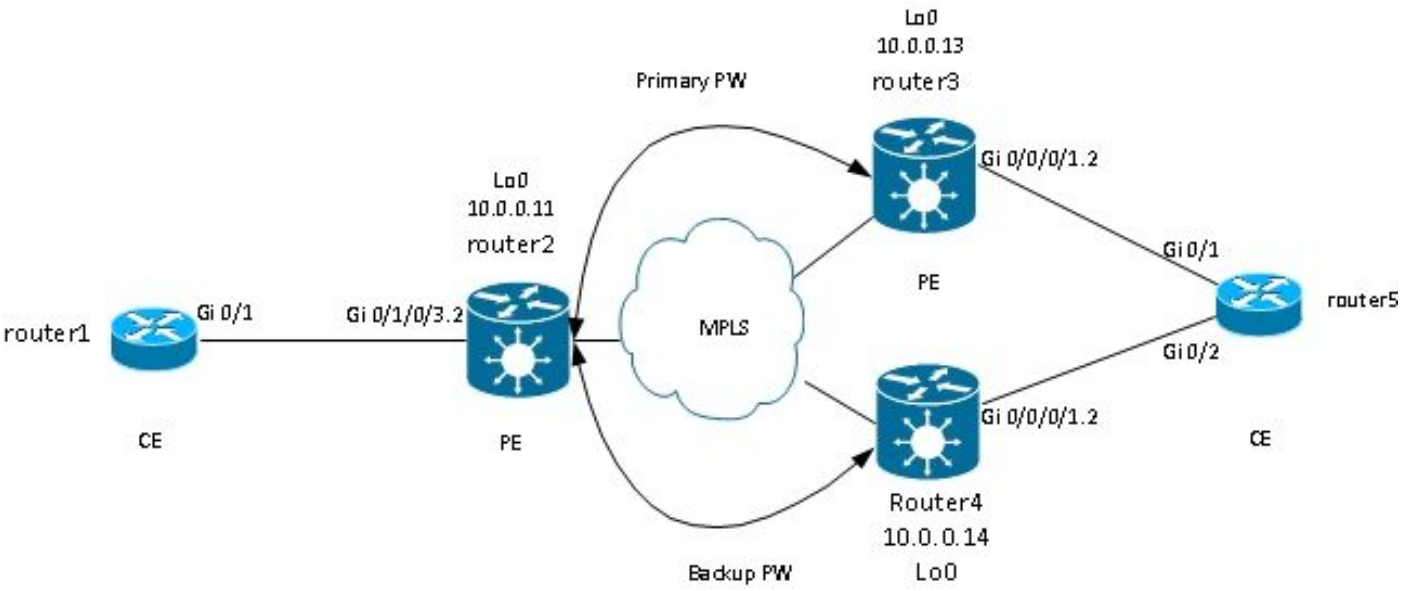
The CE is still a single point of failure in this design. Other redundancy features that can be used on the CE include:

- Multichassis Link Aggregation Group (MC-LAG)
- ASR 9000 Network Virtualization (nV) clustering
- Virtual Switching System (VSS) on Cisco IOS switches
- Virtual Port Channel (vPC) on Cisco Nexus Switches

From the perspective of the PE, there is a simple point-to-point connection between an AC and an MPLS PW.

3.2.5.3 PW Redundancy

PEs can also provide redundancy with a feature called PW Redundancy.



Router2 has a primary PW to router3. Traffic from router1 to router6 flows over that primary PW under normal circumstances. Router2 also has a backup PW to router4 in hot standby but, under normal circumstances, no traffic flows over that PW.

If there is a problem with the primary PW, with the remote PE of the primary PW (router3), or with the AC on the remote PE (router3), router2 immediately activates the backup PW, and the traffic starts flowing through it. Traffic moves back to primary PW when the problem is resolved.

The configuration on router2 is:

<#root>

```
RP/0/RSP0/CPU0:router2#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p6
   interface GigabitEthernet0/1/0/3.2
   neighbor 10.0.0.13 pw-id 222
```

backup neighbor 10.0.0.14 pw-id 222

```
    !
   !
  !
 !
!
```

The standard configuration on router3 and router4 is:

```
RP/0/RSP1/CPU0:router3#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p6
   interface GigabitEthernet0/0/0/1.2
   neighbor 10.0.0.11 pw-id 222
   !
  !
 !
!
```

Under stable conditions, the PW to router3 is active, and the PW to router4 is in a standby state:

```
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                Segment 1              Segment 2
Group      Name    ST   Description   ST     Description           ST
---------------------   ------------------   --------------------------
test       p2p6    UP   Gi0/1/0/3.2   UP     10.0.0.13     222     UP
                                              Backup
                                              10.0.0.14     222     SB
------------------------------------------------------------------------
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect group test det

Group test, XC p2p6, state is up; Interworking none
  AC: GigabitEthernet0/1/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1504; XC ID 0xc40003; interworking none
    Statistics:
      packets: received 51412, sent 25628
      bytes: received 3729012, sent 1742974
      drops: illegal VLAN 0, illegal length 0
  PW: neighbor 10.0.0.13, PW ID 222, state is up ( established )
    PW class not set, XC ID 0xc0000005
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.11
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set
```

```
    PW Status TLV in use
      MPLS         Local                        Remote
      ------------ ---------------------------- --------------------------
      Label        16049                        16059
      Group ID     0x6000180                    0x4000280
      Interface    GigabitEthernet0/1/0/3.2     GigabitEthernet0/0/0/1.2
      MTU          1504                         1504
      Control word disabled                     disabled
      PW type      Ethernet                     Ethernet
      VCCV CV type 0x2                          0x2
                   (LSP ping verification)      (LSP ping verification)
      VCCV CC type 0x6                          0x6
                   (router alert label)         (router alert label)
                   (TTL expiry)                 (TTL expiry)
      ------------ ---------------------------- --------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Outgoing Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    MIB cpwVcIndex: 3221225477
    Create time: 03/05/2013 15:04:03 (00:21:26 ago)
    Last time status changed: 03/05/2013 15:17:34 (00:07:55 ago)
    MAC withdraw message: send 0 receive 0
    Statistics:
      packets: received 25628, sent 51412
      bytes: received 1742974, sent 3729012

  Backup PW:
  PW: neighbor 10.0.0.14, PW ID 222, state is standby ( all ready )
    Backup for neighbor 10.0.0.13 PW ID 222 ( inactive )
    PW class not set, XC ID 0xc0000006
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.11
    PW type Ethernet, control word disabled, interworking none
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ --------------------------
      Label        16050                          289971
      Group ID     0x6000180                      0x4000100
      Interface    GigabitEthernet0/1/0/3.2       GigabitEthernet0/0/0/1.2
      MTU          1504                           1504
      Control word disabled                       disabled
      PW type      Ethernet                       Ethernet
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x6                            0x6
                   (router alert label)           (router alert label)
                   (TTL expiry)                   (TTL expiry)
      ------------ ------------------------------ --------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Outgoing Status (PW Status TLV):
      Status code: 0x20 (Standby) in Notification message
    MIB cpwVcIndex: 3221225478
    Create time: 03/05/2013 15:04:03 (00:21:26 ago)
    Last time status changed: 03/05/2013 15:17:34 (00:07:55 ago)
    MAC withdraw message: send 0 receive 0
RP/0/RSP0/CPU0:router2#
```

Because the AC status and the PW status are coupled, router3 signals AC down to router2 when the AC on router3 goes down. Router2 brings its primary PW down and activates the backup PW:

<#root>

```
RP/0/RSP0/CPU0:May  3 15:34:08.772 : l2vpn_mgr[1121]: %L2-L2VPN_PW-3-UPDOWN :
   Pseudowire with address 10.0.0.13, id  222, state is Down
RP/0/RSP0/CPU0:May  3 15:34:08.772 : l2vpn_mgr[1121]: %L2-L2VPN_PW-3-UPDOWN :
   Pseudowire with address 10.0.0.14, id  222, state is Up

RP/0/RSP0/CPU0:router2#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                    Segment 1               Segment 2
Group      Name       ST    Description    ST       Description            ST
------------------------   ----------------------   --------------------------
test       p2p6       UP    Gi0/1/0/3.2    UP       10.0.0.13       222    DN
                                                     Backup
                                                     10.0.0.14       222    UP
------------------------------------------------------------------------------
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect group test det

Group test, XC p2p6, state is up; Interworking none
  AC: GigabitEthernet0/1/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1504; XC ID 0xc40003; interworking none
    Statistics:
      packets: received 51735, sent 25632
      bytes: received 3752406, sent 1743230
      drops: illegal VLAN 0, illegal length 0
  PW: neighbor 10.0.0.13, PW ID 222, state is down ( local ready )
    PW class not set, XC ID 0xc0000005
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.11
    PW type Ethernet, control word disabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ --------------------------
      Label        16049                          16059
      Group ID     0x6000180                      0x4000280
      Interface    GigabitEthernet0/1/0/3.2       GigabitEthernet0/0/0/1.2
      MTU          1504                           1504
      Control word disabled                       disabled
      PW type      Ethernet                       Ethernet
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x6                            0x6
                   (router alert label)           (router alert label)
                   (TTL expiry)                   (TTL expiry)
      ------------ ------------------------------ --------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x6 (
```

AC Down

```
) in Notification message
    Outgoing Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    MIB cpwVcIndex: 3221225477
    Create time: 03/05/2013 15:04:03 (00:30:14 ago)
    Last time status changed: 03/05/2013 15:34:08 (00:00:09 ago)
    MAC withdraw message: send 0 receive 0

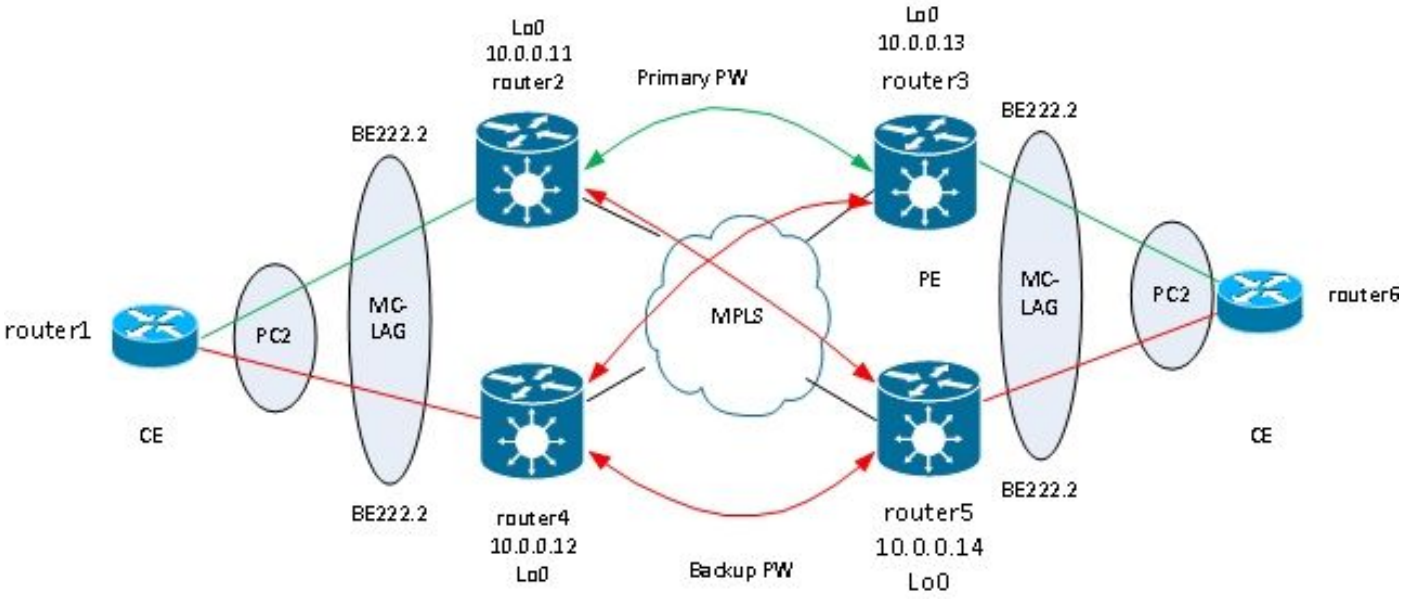
  Backup PW:
  PW: neighbor 10.0.0.14, PW ID 222, state is up ( established )
    Backup for neighbor 10.0.0.13 PW ID 222 ( active )
    PW class not set, XC ID 0xc0000006
    Encapsulation MPLS, protocol LDP
    Source address 10.0.0.11
    PW type Ethernet, control word disabled, interworking none
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                     Remote
      ------------ ------------------------- ------------------------------
      Label        16050                     289971
      Group ID     0x6000180                 0x4000100
      Interface    GigabitEthernet0/1/0/3.2  GigabitEthernet0/0/0/1.2
      MTU          1504                      1504
      Control word disabled                  disabled
      PW type      Ethernet                  Ethernet
      VCCV CV type 0x2                       0x2
                   (LSP ping verification)   (LSP ping verification)
      VCCV CC type 0x6                       0x6
                   (router alert label)      (router alert label)
                   (TTL expiry)              (TTL expiry)
      ------------ ------------------------- ------------------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    Outgoing Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    MIB cpwVcIndex: 3221225478
    Create time: 03/05/2013 15:04:03 (00:30:14 ago)
    Last time status changed: 03/05/2013 15:34:08 (00:00:09 ago)
    MAC withdraw message: send 0 receive 0
    Statistics:
      packets: received 25632, sent 51735
      bytes: received 1743230, sent 3752406
RP/0/RSP0/CPU0:router2#
```

When the AC on router3 comes back up, router2 reactivates the primary PW to router3, and the PW to router4 goes back to a standby state.

The backup PW is also activated when router3 goes down, and router2 loses the route to its loopback.

The next logical step is to introduce two-way PW redundancy with two PEs at each site:



However, this full mesh of PWs encounters a problem when two PWs are active at the same time a loop is introduced into the network. The loop needs to be broken, generally by use of the Spanning Tree Protocol (STP). However, you do not want spanning tree instability at one site to propagate to the other site. Thus, it is better not to run spanning tree on these PWs and not to merge the spanning tree between the two sites . It is simpler if there is just one logical link between the two sites so that no spanning tree is required.

One solution is to use an MC-LAG bundle between the two PEs at one site and their local CE. Only one of the two PEs has its bundle members active so that its PW to the remote site is active. The other PE has its bundle members in standby state and has its PW to the remote site down. With only one PW active between the two sites, no loop is introduced. The PE with the active PW also has a standby PW to the second PE at the remote site.

Under stable conditions, the active bundle members are on router2 and router3, and the active PW is between them. This is the configuration on router3:

```
RP/0/RSP1/CPU0:router3#sh run redundancy
redundancy
 iccp
  group 2
   mlacp node 1
   mlacp system mac 0200.0000.0002
   mlacp system priority 1
   mlacp connect timeout 0
   member
    neighbor 10.0.0.14
   !
   backbone
    interface TenGigE0/0/0/0
    interface TenGigE0/0/0/1
   !
   isolation recovery-delay 300
  !
 !
!
```

```
RP/0/RSP1/CPU0:router3#sh run int bundle-ether 222
interface Bundle-Ether222
 lacp switchover suppress-flaps 100
 mlacp iccp-group 2
 mlacp switchover type revertive
 mlacp switchover recovery-delay 40
 mlacp port-priority 1
 mac-address 0.0.2
 bundle wait-while 0
 bundle maximum-active links 1
 load-interval 30
!

RP/0/RSP1/CPU0:router3#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p7
   interface Bundle-Ether222.2
   neighbor 10.0.0.11 pw-id 222
    backup neighbor 10.0.0.12 pw-id 222
    !
   !
  !
 !
!

RP/0/RSP1/CPU0:router3#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                    Segment 1               Segment 2
Group      Name       ST   Description    ST       Description            ST
------------------------   --------------------   --------------------------
test       p2p7       UP   BE222.2        UP       10.0.0.11       222    UP
                                                   Backup
                                                   10.0.0.12       222    DN
------------------------------------------------------------------------------

RP/0/RSP1/CPU0:router3#sh bundle bundle-ether 222

Bundle-Ether222
  Status:                                  Up
  Local links :   1 / 0 / 1
  Local bandwidth :     1000000 (1000000) kbps
  MAC address (source):                    0000.0000.0002 (Configured)
  Inter-chassis link:                      No
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    1
  Wait while timer:                        Off
  Load balancing:                          Default
  LACP:                                    Operational
    Flap suppression timer:                100 ms
    Cisco extensions:                      Disabled
  mLACP:                                   Operational
    ICCP Group:                            2
    Role:                                  Active
    Foreign links :       0 / 1
    Switchover type:                       Revertive
    Recovery delay:                        40 s
    Maximize threshold:                    1 link
  IPv4 BFD:                                Not configured
```

```
  Port                  Device           State     Port ID         B/W, kbps
  --------------------  ---------------  --------  --------------  ----------
  Gi0/0/0/1             Local            Active    0x8001, 0x9001     1000000
      Link is Active
  Gi0/0/0/1             10.0.0.14        Standby   0x8002, 0xa002     1000000
      Link is marked as Standby by mLACP peer
```

On router5, the local bundle member and the primary PW to router2 are in standby state, and the backup PW to router4 is down:

```
RP/0/RSP1/CPU0:router5#sh run redundancy
redundancy
 iccp
  group 2
   mlacp node 2
   mlacp system mac 0200.0000.0002
   mlacp system priority 1
   mlacp connect timeout 0
   member
    neighbor 10.0.0.13
   !
   backbone
    interface TenGigE0/1/0/0
    interface TenGigE0/1/0/1
   !
   isolation recovery-delay 300
  !
 !
!

RP/0/RSP1/CPU0:router5#sh run int bundle-ether 222
interface Bundle-Ether222
 lacp switchover suppress-flaps 100
 mlacp iccp-group 2
 mlacp switchover type revertive
 mlacp switchover recovery-delay 40
 mac-address 0.0.2
 bundle wait-while 0
 bundle maximum-active links 1
 load-interval 30
!

RP/0/RSP1/CPU0:router5#sh run l2vpn xconnect group test
l2vpn
 xconnect group test
  p2p p2p7
   interface Bundle-Ether222.2
   neighbor 10.0.0.11 pw-id 222
    backup neighbor 10.0.0.12 pw-id 222
    !
   !
  !
 !
!

RP/0/RSP1/CPU0:router5#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
```

```
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                    Segment 1                Segment 2
Group      Name       ST   Description     ST       Description            ST
-----------------------   ----------------------   -------------------------
test       p2p7       DN   BE222.2         UP       10.0.0.11       222    SB
                                                    Backup
                                                    10.0.0.12       222    DN
------------------------------------------------------------------------------

RP/0/RSP1/CPU0:router5#sh bundle bundle-ether 222

Bundle-Ether222
  Status:                                  mLACP hot standby
  Local links :   0 / 1 / 1
  Local bandwidth :     0 (0) kbps
  MAC address (source):                    0000.0000.0002 (Configured)
  Inter-chassis link:                      No
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    1
  Wait while timer:                        Off
  Load balancing:                          Default
  LACP:                                    Operational
    Flap suppression timer:                100 ms
    Cisco extensions:                      Disabled
  mLACP:                                   Operational
    ICCP Group:                            2
    Role:                                  Standby
    Foreign links :       1 / 1
    Switchover type:                       Revertive
    Recovery delay:                        40 s
    Maximize threshold:                    1 link
  IPv4 BFD:                                Not configured

  Port                  Device        State        Port ID         B/W, kbps
  --------------------  ------------  -----------  --------------  ----------
  Gi0/0/0/1             Local         Standby      0x8002, 0xa002     1000000
      mLACP peer is active
  Gi0/0/0/1             10.0.0.13     Active       0x8001, 0x9001     1000000
      Link is Active
```

On router6, the bundle member to router3 is active, while the bundle member to router5 is in standby state:

```
router6#sh etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
```

```
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
2      Po2(SU)         LACP      Gi0/1(P)    Gi0/2(w)
```

When the bundle member on router3 goes down, router6 has its active member to router5:

```
router6#sh etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
2      Po2(SU)         LACP      Gi0/1(D)    Gi0/2(P)
```

Since the bundle-ether222 is down on router5, the coupled PW to router2 goes down at the same time:

```
RP/0/RSP1/CPU0:router3#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                    Segment 1              Segment 2
Group      Name       ST   Description   ST       Description            ST
------------------------   --------------------   ----------------------------
test       p2p7       DN   BE222.2       DN       10.0.0.11       222    DN
                                                   Backup
                                                   10.0.0.12       222    DN
------------------------------------------------------------------------------
```

Router2 detects that its PW to router3 is down and activates its backup PW to router5:

```
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed
```

```
XConnect                    Segment 1              Segment 2
Group      Name       ST   Description   ST       Description             ST
-----------------------   --------------------   --------------------------
test       p2p7       UP   BE222.2       UP       10.0.0.13       222     DN
                                                   Backup
                                                   10.0.0.14       222     UP
------------------------------------------------------------------------------
```

Router5 has its bundle member active as well as its primary PW to router2:

```
RP/0/RSP1/CPU0:router5#sh bundle bundle-ether 222

Bundle-Ether222
  Status:                                  Up
  Local links :   1 / 0 / 1
  Local bandwidth :     1000000 (1000000) kbps
  MAC address (source):                    0000.0000.0002 (Configured)
  Inter-chassis link:                      No
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    1
  Wait while timer:                        Off
  Load balancing:                          Default
  LACP:                                    Operational
    Flap suppression timer:                100 ms
    Cisco extensions:                      Disabled
  mLACP:                                   Operational
    ICCP Group:                            2
    Role:                                  Active
    Foreign links :       0 / 1
    Switchover type:                       Revertive
    Recovery delay:                        40 s
    Maximize threshold:                    1 link
  IPv4 BFD:                                Not configured

  Port                  Device       State        Port ID         B/W, kbps
  --------------------  -----------  -----------  --------------  ----------
  Gi0/0/0/1             Local        Active       0x8002, 0xa002     1000000
      Link is Active
  Gi0/0/0/1             10.0.0.13    Configured   0x8003, 0x9001     1000000
      Link is down

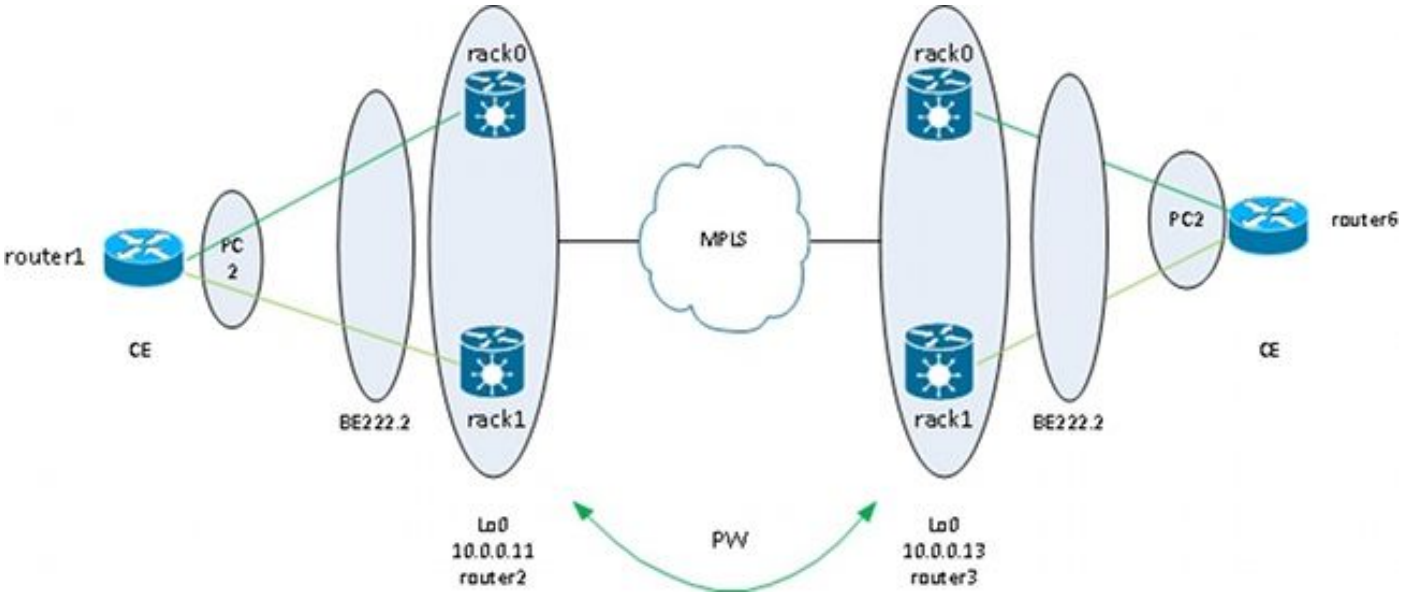
RP/0/RSP1/CPU0:router5#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                    Segment 1                Segment 2
Group      Name       ST   Description     ST       Description             ST
-----------------------   ----------------------   --------------------------
test       p2p7       UP   BE222.2         UP       10.0.0.11       222     UP
                                                   Backup
                                                   10.0.0.12       222     DN
------------------------------------------------------------------------------
```

3.2.5.4 ASR 9000 nV Edge Cluster

The previous design based on MC-LAG and PW redundancy works fine for redundancy but, because some bundle members are in standby state, they do not carry traffic under steady conditions.

If you want all bundle members active, even under stable conditions, you can use an ASR 9000 cluster with bundle members from the CE connected to each rack of the PE:



This design offers redundancy against a bundle member link failure between the CE and the PE, a rack failure, and a core link failure - as long as the cluster is dually attached to the MPLS core and there is redundancy in the core. The two racks do not have to be co-located and could be at different locations. Inter-rack links are not represented in this diagram.

If you want redundancy on the CE, you can use a multichassis solution for the CE:

- MC-LAG
- ASR 9000 nV clustering
- VSS
- vPC

The configuration on the ASR 9000 cluster is very basic:

```
interface TenGigE0/0/0/8
 bundle id 222 mode on
!
interface TenGigE1/0/0/8
 bundle id 222 mode on
!
interface Bundle-Ether222
!
interface Bundle-Ether222.2 l2transport
```

```
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
l2vpn
 xconnect group test
  p2p p2p8
   interface Bundle-Ether222.2
   neighbor 10.0.0.13 pw-id 8
   !
  !
 !
!
```

Cisco recommends you configure a static LACP system MAC address and a bundle MAC address in order to avoid a MAC address change caused by a designated shelf controller switchover. This example shows how to find the addresses:

```
RP/1/RSP0/CPU0:router2#sh int bundle-ether 222 | i address is
  Hardware is Aggregated Ethernet interface(s), address is 0024.f71e.d309
  Internet address is Unknown

RP/1/RSP0/CPU0:router2#
RP/1/RSP0/CPU0:router2#conf
RP/1/RSP0/CPU0:router2(config)#int bundle-ether 222
RP/1/RSP0/CPU0:router2(config-if)#mac-address 0024.f71e.d309
RP/1/RSP0/CPU0:router2(config-if)#commit
RP/1/RSP0/CPU0:router2(config-if)#end
RP/1/RSP0/CPU0:router2#
RP/1/RSP0/CPU0:router2#sh lacp system-id

Priority  MAC Address
--------  -----------------
  0x8000  00-24-f7-1e-d3-05
RP/1/RSP0/CPU0:router2#
RP/1/RSP0/CPU0:router2#conf
RP/1/RSP0/CPU0:router2(config)#lacp system mac 0024.f71e.d305
RP/1/RSP0/CPU0:router2(config)#commit
RP/1/RSP0/CPU0:router2(config)#end
```

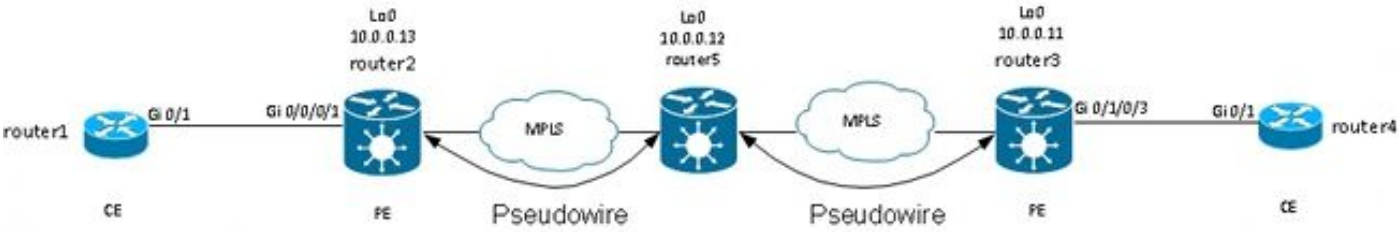
In summary, this is the bundle-ether 222 with a member on each rack (ten 0/0/0/8 on rack 0 and ten 1/0/0/8 on rack 1) and the bundle subinterface configured for a point-to-point cross connect:

```
RP/1/RSP0/CPU0:router2#sh l2vpn xconnect group test
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect                   Segment 1               Segment 2
Group      Name       ST   Description     ST      Description            ST
------------------------   ----------------------  ---------------------------
test       p2p8       UP   BE222.2         UP      10.0.0.13       8      UP
------------------------------------------------------------------------------
```

3.3 CDP

Cisco routers and switches usually send CDP packets without dot1q tags. There are multiple scenarios that determine what happens to these CDP packets when they are received by an IOS XR router configured for a cross connect:



In this topology, router1 can see its local PE router2 as a CDP neighbor or the remote CE router4, depending on the configuration.

3.3.1 CDP Not Enabled on Main Interface of L2VPN PE

The CDP packets from the L2VPN CE are transported over the cross connect. The two L2VPN CEs see each other (with use of the **show cdp neighbors** command) if the main interface is configured as l2transport or if there is a subinterface matching the untagged CDP frames.

This is an example of the main interface:

```
interface GigabitEthernet0/0/0/1
 l2transport
 !
!
l2vpn
 xconnect group test
  p2p p2p8
   interface GigabitEthernet0/0/0/1
   neighbor 10.0.0.11 pw-id 8
   !
  !
 !
!
```

This is an example of an untagged subinterface:

```
interface GigabitEthernet0/0/0/1.1 l2transport
 encapsulation untagged
```

```
!
l2vpn
 xconnect group test
  p2p p2p8
   interface GigabitEthernet0/0/0/1.1
   neighbor 10.0.0.11 pw-id 8
   !
  !
 !
!
```

In these two examples, the CDP packets are transported over the cross connect, and the CEs see each other as CDP neighbors. The CE does not see the PE as a CDP neighbor:

```
router1#sh cdp nei gigabitEthernet 0/1
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
router4          Gig 0/1           168              R S   ME-3400G- Gig 0/1
```

3.3.2 CDP Enabled on Main Interface of L2VPN PE

The PE processes the untagged CDP packets, and the PE and CE see each other as neighbors. However, the CE does not see the remote CE when CDP is enabled on the main interface of the L2VPN PE.

Note that:

- You cannot configure CDP on a main interface that is configured as l2transport.
- The PE intercepts the CDP packets when CDP is configured on the main non-l2transport interface. This occurs even if there is an l2transport subinterface configured to match the untagged CDP packets (with use of the **encapsulation untagged** or **encapsulation default** commands). CDP packets are not transported to the remote site in this case.

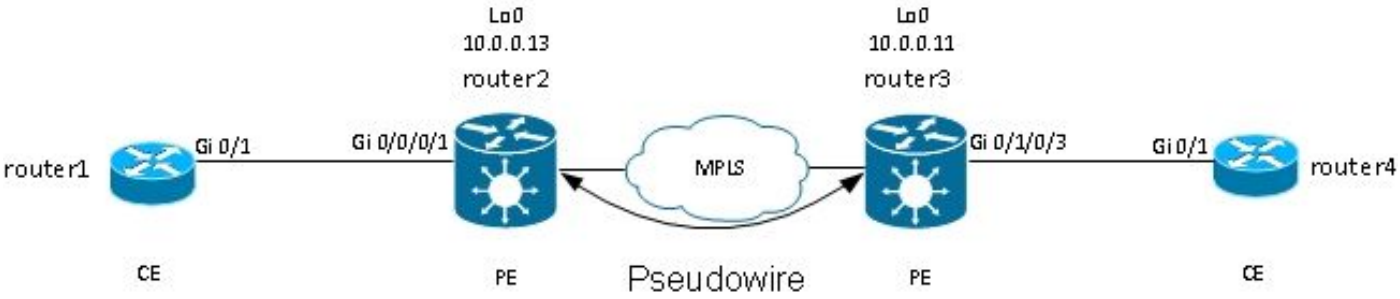
3.4 Spanning Tree

If the L2VPN CE is an Ethernet switch and is sending spanning tree BPDUs to the L2VPN PE, these BPDUs are handled as regular traffic and are transported according to the L2VPN configuration.

STP or MST BPDUs are sent untagged and are transported through the point-to-point cross connect if the main interface is configured as l2transport or if there is a l2transport subinterface configured with the **encapsulation untagged** or **encapsulation default** commands.

Per VLAN Spanning Tree Plus (PVST+) or Rapid PVST+ (PVRST+) send tagged BPDUs that are transported if there is a l2transport subinterface that matches the dot1q tag of the BPDUs.

This is an example topology:



Router2 and router3 are transporting untagged frames and frames with dot1q tag 2:

<#root>

```
interface GigabitEthernet0/0/0/1.1 l2transport
```

encapsulation untagged

```
!
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
l2vpn
 xconnect group test
  p2p p2p8
   interface GigabitEthernet0/0/0/1.2
   neighbor 10.0.0.11 pw-id 8
   !
  !
  p2p p2p9
   interface GigabitEthernet0/0/0/1.1
   neighbor 10.0.0.11 pw-id 9
   !
  !
 !
!
```

Switch1 receives the untagged BPDUs in VLAN 1 and the tagged BPDUs in VLAN2 from switch4; its root port is on Gi0/1 towards switch4:

```
switch1#sh spanning-tree vlan 1

VLAN0001
```

```
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
             Address     0024.985e.6a00
             Cost        8
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     001d.4603.1f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 4         128.1    P2p

switch1#sh spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32770
             Address     0019.552b.b580
             Cost        4
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     001d.4603.1f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 15

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 4         128.1    P2p
```

With this configuration, the spanning tree domain at site A is merged with the spanning tree domain at side B. A potential problem is that spanning tree instability at one site can propagate to the other site.

If you are confident that one site is connected only through one PW to another site and that there is no backdoor link that could introduce a physical loop, it is a good idea not to run spanning tree over the two sites. This keeps the two spanning tree domains isolated. To do this, configure a spanning tree bpdufilter on the CEs, or configure an ethernet-services access-list on the PEs to drop frames with the destination MAC address used by BPDUs. An ethernet-services access-list on the PEs can be used to drop frames with the BPDU destination MAC or other kinds of L2 protocols that you do not want to forward over the PW.

This is an access-list that you could use under each l2transport (sub)interface that is being transported between the two sites:

```
ethernet-services access-list block-invalid-frames
 10 deny any 0180.c200.0000 0000.0000.000f
 20 deny any host 0180.c200.0010
 30 deny any host 0100.0c00.0000
 40 deny any host 0100.0ccc.cccc
 50 deny any host 0100.0ccc.cccd
```

```
 60 deny any host 0100.0ccd.cdce
 70 permit any any
!

RP/0/RSP1/CPU0:router2#sh run int GigabitEthernet0/0/0/1.1
interface GigabitEthernet0/0/0/1.1 l2transport
 encapsulation untagged
 ethernet-services access-group block-invalid-frames ingress
 ethernet-services access-group block-invalid-frames egress
!

RP/0/RSP1/CPU0:router2#sh run int GigabitEthernet0/0/0/1.2
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
 ethernet-services access-group block-invalid-frames ingress
 ethernet-services access-group block-invalid-frames egress
!
```

The ethernet-services ACL starts to drop the BPDUs:

```
RP/0/RSP1/CPU0:router2#sh access-lists ethernet-services block-invalid-frames
   hardware ingress location 0/0/CPU0
ethernet-services access-list block-invalid-frames
 10 deny any 0180.c200.0000 0000.0000.000f (41 hw matches)
 20 deny any host 0180.c200.0010
 30 deny any host 0100.0c00.0000
 40 deny any host 0100.0ccc.cccc
 50 deny any host 0100.0ccc.cccd (63 hw matches)
 60 deny any host 0100.0ccd.cdce
 70 permit any any (8 hw matches)
```

Switch1 does not receive the BPDUs from switch4 anymore, so switch1 is now the root:

```
switch1#sh spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     001d.4603.1f00
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     001d.4603.1f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 15

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- ----------------------
Gi0/1               Desg FWD 4         128.1    P2p
```

```
switch1#sh spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32770
             Address     001d.4603.1f00
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     001d.4603.1f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 15

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- ----------------------
Gi0/1               Desg FWD 4         128.1    P2p
```

The risk of disabling spanning tree on a link is this: if a backdoor connection is created between the sites, it introduces a physical loop, and spanning tree cannot break the loop. So, when you disable spanning tree over a PW, ensure that there are no redundant links between sites and that the PW remains the only connection between the sites.

If there are multiple connections between sites, use a solution like VPLS along with an access gateway version of the spanning tree, such as MST Access Gateway (MSTAG) or PVST+ Access Gateway (PVSTAG). See the Multipoint Service section for details.

4. Multipoint Service

See Implementing Multipoint Layer 2 Services for a complete description of the multipoint L2 features.

With only two interfaces in a point-to-point cross connect, an L2VPN switch takes everything received on side and forwards it on the other side.

When there are more than two interfaces in a bridge-domain, an Ethernet switch has to make a switching decision in order to determine where to forward frames based on their destination MAC address. The switch does MAC learning based on the source MAC address of the frames that it receives and builds a mac-address-table.

The switch forwards frames in this method:

- Broadcast frames are flooded to all ports. Use storm control in order to limit the broadcast flooding rate.
- Multicast frames are flooded to all ports in the bridge-domain, except when Internet Group Management Protocol (IGMP) or Multicast Listener Discovery (MLD) snooping is configured. Use storm control in order to limit the multicast flooding rate.
- Unicast frames with a destination MAC address that is not part of the mac-address-table of the bridge-domain (unknown unicast) are flooded on all ports in the bridge-domain. Use storm control in order to limit the unknown-unicast flooding rate.
- Unicast frames with a destination MAC address that is part of the mac-address-table of the bridge-

domain are forwarded to the port where the destination MAC address has been learned.

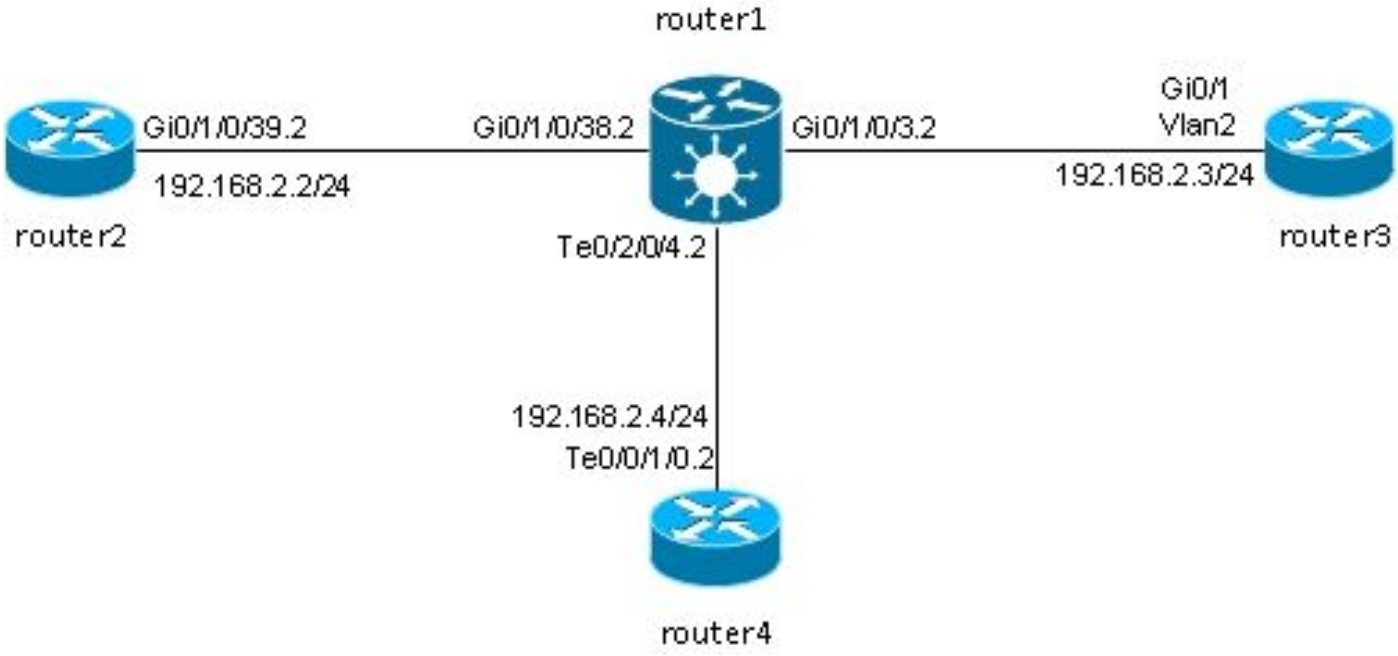
In Cisco IOS XR software, a broadcast domain or an emulated LAN is called a bridge-domain. This is similar to a VLAN in Cisco IOS software terminology, except that a VLAN in IOS is linked to a VLAN number that is used as the dot1q tag on the trunks. A bridge-domain in Cisco IOS XR software is not linked to a dot1q VLAN tag number. You can use the EVC model in order to manipulate the dot1q tags and have dot1q subinterfaces with different dot1q VLAN numbers in the same bridge-domain or to have untagged interfaces.

A bridge-domain is basically one broadcast domain where broadcasts and multicast frames are flooded. One mac-address-table is associated with each bridge-domain (unless MAC learning is disabled manually by configuration, which is very rare). This usually corresponds to one IPv4 or IPv6 subnet where all hosts in the bridge-domain are directly connected.

Bridge-domains can be grouped within a bridge group. This is a convenient way to check the configuration. You can execute one show command for a bridge group instead of one show command for each bridge-domain. A bridge group does not have a mac-address-table or other associations; it is just used for configuration and show commands.

4.1 Local Switching

This is a very basic example:



Router2, router3, and router4 are connected through an ASR 9000, which simulates a LAN between those three routers.

These are the interface configurations on those three routers:

```
RP/0/RSP0/CPU0:router2#sh run int gig 0/1/0/39.2
interface GigabitEthernet0/1/0/39.2
 ipv4 address 192.168.2.2 255.255.255.0
 encapsulation dot1q 2
!

router3#sh run int gig 0/1
Building configuration...

Current configuration : 203 bytes
!
interface GigabitEthernet0/1
 port-type nni
 switchport access vlan 2
 switchport trunk allowed vlan 1,2
 switchport mode trunk
end

router3#sh run int vlan 2
Building configuration...

Current configuration : 61 bytes
!
interface Vlan2
 ip address 192.168.2.3 255.255.255.0
end

router3#

RP/0/RSP0/CPU0:router4#sh run int ten 0/0/1/0.2
interface TenGigE0/0/1/0.2
 ipv4 address 192.168.2.4 255.255.255.0
 encapsulation dot1q 2
!
```

Packets are received by router1 with the dot1q tag 2 and are forwarded to the other routers with the dot1q tag 2.

In this basic scenario, there are two options on the ACs:

1. Since all ACs are using the dot1q tag 2, you can keep it on the frame and forward the frame on the egress interface with the same dot1q tag as received on the ingress interface. The **rewrite ingress tag pop 1 symmetric** command is not required.

2. You can pop the incoming dot1q tag 2 in the ingress direction and symmetrically push the dot1q tag 2 in the egress direction. While this is not required in this basic scenario, it is a good idea to configure the bridge-domain in this fashion at the start because it provides more flexibility for the future. Here are two examples of changes that can occur after initial configuration:

    1. If a routed BVI interface is introduced later in the bridge-domain, packets must be processed on the BVI without tags. See the BVI section for details.
    2. A new AC, which uses a different dot1q tag, is added later. The dot1q tag 2 would be popped in the ingress direction, and the other dot1q tag would be pushed on the new interface in the egress direction and vice versa.

Pop the dot1q tags on each AC on router1:

```
RP/0/RSP0/CPU0:router1#sh run int GigabitEthernet0/1/0/3.2
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!

RP/0/RSP0/CPU0:router1#sh run int GigabitEthernet0/1/0/38.2
interface GigabitEthernet0/1/0/38.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!

RP/0/RSP0/CPU0:router1#sh run int TenGigE0/2/0/4.2
interface TenGigE0/2/0/4.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
```

View the configuration of the bridge-domain with these three ACs:

```
RP/0/RSP0/CPU0:router1#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain engineering
   interface TenGigE0/2/0/4.2
   !
   interface GigabitEthernet0/1/0/3.2
   !
   interface GigabitEthernet0/1/0/38.2
   !
  !
 !
!
```

The bridge-domain must be configured under a bridge group. If other bridge-domains from this customer are needed, they can be configured under the same bridge group, customer1. If new bridge-domains belong to a different customer, you can create a new bridge group. These examples use the customer in order to group bridge-domains, but bridge-domains can be grouped by any criteria.

Use the **show run l2vpn bridge group customer1 bridge-domain engineering** command in order to display the configuration of the bridge-domain.

Use the **show run l2vpn bridge group customer1** command in order to view the configuration of all bridge-domains.

Use the **show l2vpn bridge-domain bd-name engineering** command or the **show l2vpn bridge-domain group customer1** command in order to display information about the bridge-domain.

```
RP/0/RSP0/CPU0:router1#show l2vpn bridge-domain group customer1 bd-name
   engineering
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 3 (3 up), VFIs: 0, PWs: 0 (0 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.2, state: up, Static MAC addresses: 0
    Gi0/1/0/38.2, state: up, Static MAC addresses: 0
    Te0/2/0/4.2, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:

RP/0/RSP0/CPU0:router1#show l2vpn bridge-domain group customer1 bd-name
   engineering det
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
  Split Horizon Group: none
  Dynamic ARP Inspection: disabled, Logging: disabled
  IP Source Guard: disabled, Logging: disabled
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Bridge MTU: 1500
  MIB cvplsConfigIndex: 6
  Filter MAC addresses:
  Create time: 28/05/2013 17:17:03 (00:18:06 ago)
  No status change since creation
  ACs: 3 (3 up), VFIs: 0, PWs: 0 (0 up), PBBs: 0 (0 up)
  List of ACs:
    AC: GigabitEthernet0/1/0/3.2, state is up
      Type VLAN; Num Ranges: 1
      VLAN ranges: [2, 2]
      MTU 1500; XC ID 0xc40003; interworking none
      MAC learning: enabled
      Flooding:
        Broadcast & Multicast: enabled
        Unknown unicast: enabled
      MAC aging time: 300 s, Type: inactivity
      MAC limit: 4000, Action: none, Notification: syslog
      MAC limit reached: no
```

```
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
  Split Horizon Group: none
  Dynamic ARP Inspection: disabled, Logging: disabled
  IP Source Guard: disabled, Logging: disabled
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Storm Control: disabled
  Static MAC addresses:
  Statistics:
    packets: received 185066, sent 465
    bytes: received 13422918, sent 34974
  Storm control drop counters:
    packets: broadcast 0, multicast 0, unknown unicast 0
    bytes: broadcast 0, multicast 0, unknown unicast 0
  Dynamic ARP inspection drop counters:
    packets: 0, bytes: 0
  IP source guard drop counters:
    packets: 0, bytes: 0
AC: GigabitEthernet0/1/0/38.2, state is up
  Type VLAN; Num Ranges: 1
  VLAN ranges: [2, 2]
  MTU 1500; XC ID 0xc40005; interworking none
  MAC learning: enabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
  Split Horizon Group: none
  Dynamic ARP Inspection: disabled, Logging: disabled
  IP Source Guard: disabled, Logging: disabled
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Storm Control: disabled
  Static MAC addresses:
  Statistics:
    packets: received 8, sent 12287
    bytes: received 770, sent 892418
  Storm control drop counters:
    packets: broadcast 0, multicast 0, unknown unicast 0
    bytes: broadcast 0, multicast 0, unknown unicast 0
  Dynamic ARP inspection drop counters:
    packets: 0, bytes: 0
  IP source guard drop counters:
    packets: 0, bytes: 0
AC: TenGigE0/2/0/4.2, state is up
  Type VLAN; Num Ranges: 1
  VLAN ranges: [2, 2]
  MTU 1500; XC ID 0x1040001; interworking none
  MAC learning: enabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
```

```
      Split Horizon Group: none
      Dynamic ARP Inspection: disabled, Logging: disabled
      IP Source Guard: disabled, Logging: disabled
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      Storm Control: disabled
      Static MAC addresses:
      Statistics:
        packets: received 463, sent 11839
        bytes: received 35110, sent 859028
      Storm control drop counters:
        packets: broadcast 0, multicast 0, unknown unicast 0
        bytes: broadcast 0, multicast 0, unknown unicast 0
      Dynamic ARP inspection drop counters:
        packets: 0, bytes: 0
      IP source guard drop counters:
        packets: 0, bytes: 0
  List of Access PWs:
  List of VFIs:
```

Use the **show l2vpn bridge-domain group customer1 bd-name engineering det** command if you want to check that packets are received and sent on each AC.

Add the mac-address keyword to the **show l2vpn forwarding bridge-domain** command if you want to check the mac-address-table:

<#root>

```
RP/0/RSP0/CPU0:router1#show l2vpn forwarding bridge-domain customer1:
   engineering
```

mac-address

```
 location 0/1/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on  LC learned Resync Age   Mapped to
--------------------------------------------------------------------------------
0019.552b.b581 dynamic Gi0/1/0/3.2               0/1/CPU0   0d 0h 0m 0s   N/A
0019.552b.b5c3 dynamic Gi0/1/0/3.2               0/1/CPU0   0d 0h 0m 0s   N/A
0024.986c.6417 dynamic Gi0/1/0/38.2              0/1/CPU0   0d 0h 0m 0s   N/A
6c9c.ed3e.e484 dynamic Te0/2/0/4.2               0/2/CPU0   0d 0h 0m 0s   N/A
```

The MAC learning is executed in hardware by the linecards each time a frame is received in the bridge-domain. There is also a software cache of the mac-address-table, but this software table cannot be updated continuously in order to match the hardware entries. When the **show** command is entered in recent code, it tries to resynchronize the software table with the hardware table. After a maximum of 15 seconds, it prints the current state of the software mac-address-table, even if the resynchronization is not complete (for example, if the table is large). Use the **l2vpn resynchronize forwarding mac-address-table** command in order to resynchronize the software and hardware tables manually.

```
RP/0/RSP0/CPU0:router1#term mon
RP/0/RSP0/CPU0:router1#l2vpn resynchronize forwarding mac-address-table
   location 0/1/CPU0
RP/0/RSP0/CPU0:router1#LC/0/1/CPU0:May 28 18:25:35.734 : vkg_l2fib_mac_cache[357]
   %PLATFORM-
PLAT_L2FIB_MAC_CACHE-6-RESYNC_COMPLETE : The resynchronization of the MAC
   address table is complete
0/1/CPU0

RP/0/RSP0/CPU0:router1#show l2vpn forwarding bridge-domain customer1:engineering
   mac-address location 0/1/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age   Mapped to
------------------------------------------------------------------------------
0019.552b.b581 dynamic Gi0/1/0/3.2              0/1/CPU0   0d 0h 0m 0s  N/A
0019.552b.b5c3 dynamic Gi0/1/0/3.2              0/1/CPU0   0d 0h 0m 0s  N/A
6c9c.ed3e.e484 dynamic Te0/2/0/4.2              0/2/CPU0   0d 0h 0m 0s  N/A
```

A syslog message indicates when the resynchronization process is complete, so it is useful to have **terminal monitor** enabled in order to see the message.

The Resync Age column displays the last time that the MAC address was resynchronized from the hardware table.

The location keyword is the location of an incoming or an outgoing linecard. The MAC addresses are exchanged between linecards in hardware, so MAC addresses can be known on each linecard where there is an AC or a PW. The detail keyword can provide a more up-to-date version of the software table:

<#root>

```
RP/0/RSP0/CPU0:router1#show l2vpn forwarding bridge-domain customer1:
   engineering mac-address
```

detail

location

```
 0/1/CPU0

Bridge-domain name: customer1:engineering, id: 5, state: up
 MAC learning: enabled
 MAC port down flush: enabled
 Flooding:
   Broadcast & Multicast: enabled
   Unknown unicast: enabled
 MAC aging time: 300 s, Type: inactivity
 MAC limit: 4000, Action: none, Notification: syslog
 MAC limit reached: no
 MAC Secure: disabled, Logging: disabled
 DHCPv4 snooping: profile not known on this node
 Dynamic ARP Inspection: disabled, Logging: disabled
 IP Source Guard: disabled, Logging: disabled
```

```
 IGMP snooping: disabled, flooding: enabled
 Bridge MTU: 1500 bytes
 Number of bridge ports: 3
 Number of MAC addresses: 4
 Multi-spanning tree instance: 0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location


  GigabitEthernet0/1/0/3.2, state: oper up
    Number of MAC: 2
    Statistics:
      packets: received 187106, sent 757
      bytes: received 13571342, sent 57446
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic arp inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0

 Mac Address: 0019.552b.b581, LC learned: 0/1/CPU0
```

Resync Age

```
: 0d 0h 0m 0s, Flag: local


 Mac Address: 0019.552b.b5c3, LC learned: 0/1/CPU0
```

Resync Age

```
: 0d 0h 0m 0s, Flag: local


  GigabitEthernet0/1/0/38.2, state: oper up
    Number of MAC: 1
    Statistics:
      packets: received 18, sent 14607
      bytes: received 1950, sent 1061882
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic arp inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0

 Mac Address: 0024.986c.6417, LC learned: 0/1/CPU0
```

Resync Age

```
: 0d 0h 0m 0s, Flag: local


  TenGigE0/2/0/4.2, state: oper up
    Number of MAC: 1
    Statistics:
      packets: received 0, sent 0
      bytes: received 0, sent 0
```

```
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic arp inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0

 Mac Address: 6c9c.ed3e.e484, LC learned: 0/2/CPU0

Resync Age

: 0d 0h 0m 0s, Flag: remote
```

The detailed version of the command provides the total number of MAC addresses learned in the bridge-domain, as well as the number of MAC addresses learned under each AC.

The hardware keyword polls the hardware mac-address-table directly from the ingress or egress forwarding engines:

<#root>

```
RP/0/RSP0/CPU0:router1#show l2vpn forwarding bridge-domain customer1:
   engineering mac-address

hardware

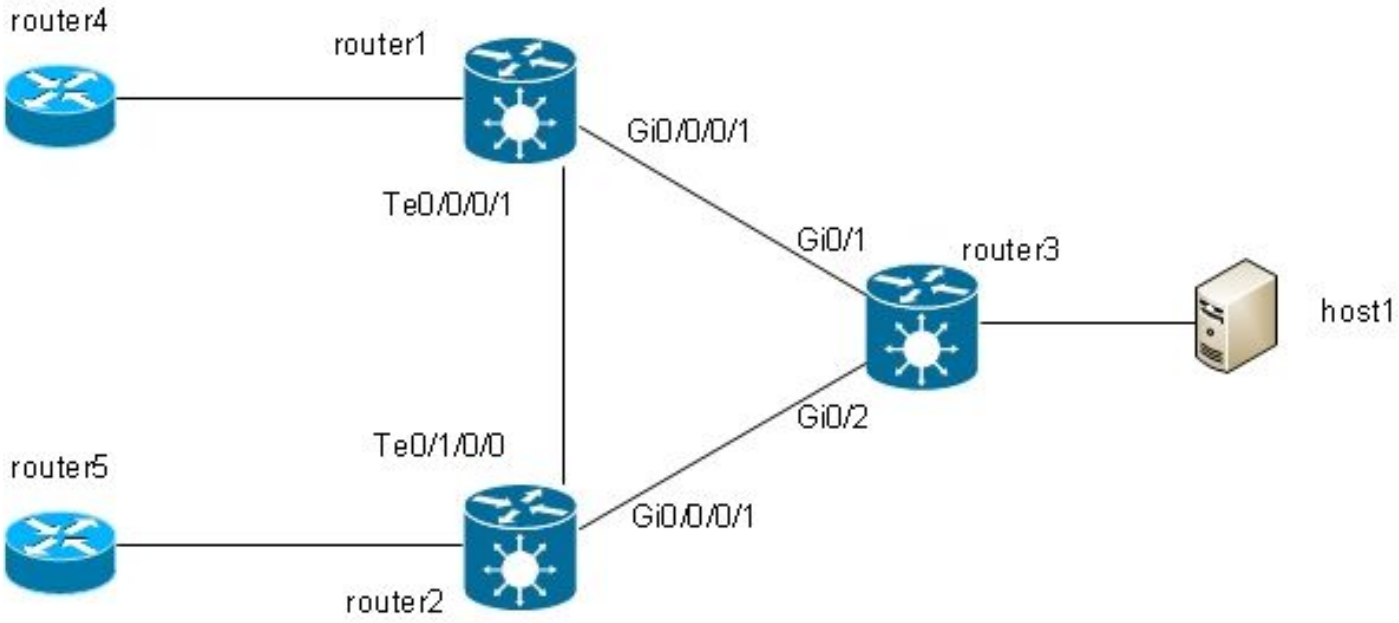
 ingress location 0/1/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age  Mapped to
------------------------------------------------------------------------------
0019.552b.b581 dynamic Gi0/1/0/3.2             0/1/CPU0   0d 0h 0m 0s N/A
0019.552b.b5c3 dynamic Gi0/1/0/3.2             0/1/CPU0   0d 0h 0m 0s N/A
0024.986c.6417 dynamic Gi0/1/0/38.2            0/1/CPU0   0d 0h 0m 0s N/A
6c9c.ed3e.e484 dynamic Te0/2/0/4.2             0/2/CPU0   0d 0h 0m 0s N/A
RP/0/RSP0/CPU0:router1#show l2vpn forwarding bridge-domain customer1:
   engineering mac-address hardware egress location 0/2/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age   Mapped to
-------------------------------------------------------------------------------
0019.552b.b581 dynamic Gi0/1/0/3.2             0/1/CPU0   0d 0h 0m 14s N/A
0019.552b.b5c3 dynamic Gi0/1/0/3.2             0/1/CPU0   0d 0h 0m 1s  N/A
0024.986c.6417 dynamic Gi0/1/0/38.2            0/1/CPU0   0d 0h 0m 10s N/A
6c9c.ed3e.e484 dynamic Te0/2/0/4.2             0/2/CPU0   0d 0h 0m 13s N/A
RP/0/RSP0/CPU0:router1#
```

4.2 Full MST

The previous examples of local switching were basic because only routers were connected to the bridge-domain. Once you start to connect L2 switches, however, you can introduce a loop and need the STP in order to break the loop:



In this topology, router1, router2, and router3 are each configured with a bridge-domain with all their interfaces in the diagram. If router4 sends a broadcast, such as an ARP request, to router1, router1 floods it to router2 and router3, router2 floods it to router3, and router3 floods it to router2. This results in a loop and a broadcast storm.

To break the loop, use an STP. There are multiple types of STPs, but Cisco IOS XR software offers only one full implementation, the MST.

There are also access gateway versions of the protocols supported in Cisco IOS XR software, such as PVSTAG and MSTAG. These are static, limited versions of the protocol to use in specific topologies, typically with VPLS, and are described in the MSTAG and PVSTAG sections. In Cisco IOS XR software, MST is the only option if there is a topology with multiple switches and if a full spanning tree implementation is required.

Two subinterfaces are configured on each router and added to a bridge-domain. For router1, the configuration is:

```
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface GigabitEthernet0/0/0/1.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!
interface TenGigE0/0/0/1.2 l2transport
```

```
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface TenGigE0/0/0/1.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!
l2vpn
 bridge group customer1
  bridge-domain finance
   interface TenGigE0/0/0/1.3
   !
   interface GigabitEthernet0/0/0/1.3
   !
  !
  bridge-domain engineering
   interface TenGigE0/0/0/1.2
   !
   interface GigabitEthernet0/0/0/1.2
   !
  !
 !
!
```

MST is configured on the main interface. In this example, VLAN 2 is assigned to instance 1, and all other VLANs remain the default instance 0. (A more realistic configuration would split VLANs evenly between instances.)

The selection of the root bridge within an STP network is determined by the configured priority and the embedded bridge ID of each device. The device with the lowest priority, or with equal lowest priority but the lowest bridge ID, is selected as the root bridge. In this example, router3 is configured with a lower priority then router1 for instance 0, so router3 is the root for instance 0. Router1 has a lower priority then router3 for instance 1, so router1 is the root for instance 1.

This is the configuration for router1:

```
spanning-tree mst customer1
 name customer1
 revision 1
 instance 0
  priority 28672
 !
 instance 1
  vlan-ids 2
  priority 24576
 !
 interface TenGigE0/0/0/1
 !
 interface GigabitEthernet0/0/0/1
 !
!
```

This is the configuration on router3:

```
spanning-tree mode mst
spanning-tree extend system-id
!
spanning-tree mst configuration
 name customer1
 revision 1
 instance 1 vlan 2
!
spanning-tree mst 0 priority 24576
spanning-tree mst 1 priority 28672
```

The name, revision, and VLAN-to-instance mapping must be the same on all switches.

Now, check the spanning tree status on router1:

```
RP/0/RSP1/CPU0:router1#sh spanning-tree mst customer1
Role:  ROOT=Root, DSGN=Designated, ALT=Alternate, BKP=Backup, MSTR=Master
State: FWD=Forwarding, LRN=Learning, BLK=Blocked, DLY=Bringup Delayed

Operating in dot1q mode


MSTI 0 (CIST):

  VLANS Mapped: 1,3-4094

  CIST Root  Priority    24576
             Address     001d.4603.1f00
             Ext Cost    0

  Root ID    Priority    24576
             Address     001d.4603.1f00
             Int Cost    20000
             Max Age 20 sec, Forward Delay 15 sec


  Bridge ID  Priority    28672 (priority 28672 sys-id-ext 0)
             Address     4055.3912.f1e6
             Max Age 20 sec, Forward Delay 15 sec
             Max Hops 20, Transmit Hold count  6


Interface    Port ID           Role State Designated           Port ID
             Pri.Nbr Cost                 Bridge ID            Pri.Nbr
------------ ------- --------- ---- ----- -------------------- -------
Gi0/0/0/1    128.2   20000     ROOT FWD   24576 001d.4603.1f00 128.1
Te0/0/0/1    128.1   2000      DSGN FWD   28672 4055.3912.f1e6 128.1


MSTI 1:
```

```
  VLANS Mapped: 2

  Root ID    Priority    24576
             Address     4055.3912.f1e6
             This bridge is the root
             Int Cost    0
             Max Age 20 sec, Forward Delay 15 sec


  Bridge ID  Priority    24576 (priority 24576 sys-id-ext 0)
             Address     4055.3912.f1e6
             Max Age 20 sec, Forward Delay 15 sec
             Max Hops 20, Transmit Hold count  6


Interface    Port ID           Role State Designated           Port ID
             Pri.Nbr Cost                 Bridge ID            Pri.Nbr
------------ ------- --------- ---- ----- -------------------- -------
Gi0/0/0/1    128.2   20000     DSGN FWD   24576 4055.3912.f1e6 128.2
Te0/0/0/1    128.1   2000      DSGN FWD   24576 4055.3912.f1e6 128.1
```

Router3 is the root for instance 0, so router1 has its root port on Gi0/0/0/1 towards router3. Router1 is the root for instance 1, so router1 is the designated bridge on all interfaces for that instance.

Router2 is blocked for instance 0 on Te0/1/0/0:

<#root>

```
RP/0/RSP1/CPU0:router2#sh spanning-tree mst customer1
Role:  ROOT=Root, DSGN=Designated, ALT=Alternate, BKP=Backup, MSTR=Master
State: FWD=Forwarding, LRN=Learning, BLK=Blocked, DLY=Bringup Delayed

Operating in dot1q mode


MSTI 0 (CIST):

  VLANS Mapped: 1,3-4094

  CIST Root  Priority    24576
             Address     001d.4603.1f00
             Ext Cost    0

  Root ID    Priority    24576
             Address     001d.4603.1f00
             Int Cost    20000
             Max Age 20 sec, Forward Delay 15 sec


  Bridge ID  Priority    32768 (priority 32768 sys-id-ext 0)
             Address     f025.72a7.b13e
             Max Age 20 sec, Forward Delay 15 sec
             Max Hops 20, Transmit Hold count  6


Interface    Port ID           Role State Designated           Port ID
```

```
             Pri.Nbr Cost                 Bridge ID            Pri.Nbr
------------ ------- --------- ---- ----- -------------------- -------
Gi0/0/0/1    128.2   20000     ROOT FWD   24576 001d.4603.1f00 128.2

Te0/1/0/0

    128.1   2000      ALT

BLK

   28672 4055.3912.f1e6 128.1


MSTI 1:

  VLANS Mapped: 2

  Root ID    Priority    24576
             Address     4055.3912.f1e6
             Int Cost    2000
             Max Age 20 sec, Forward Delay 15 sec


  Bridge ID  Priority    32768 (priority 32768 sys-id-ext 0)
             Address     f025.72a7.b13e
             Max Age 20 sec, Forward Delay 15 sec
             Max Hops 20, Transmit Hold count  6


Interface    Port ID           Role State Designated           Port ID
             Pri.Nbr Cost                 Bridge ID            Pri.Nbr
------------ ------- --------- ---- ----- -------------------- -------
Gi0/0/0/1    128.2   20000     DSGN FWD   32768 f025.72a7.b13e 128.2
Te0/1/0/0    128.1   2000      ROOT FWD   24576 4055.3912.f1e6 128.1
RP/0/RSP1/CPU0:router2#
```

Te0/1/0/0.2 is forwarding while Te0/1/0/0.3 is blocked. When the STP Blocked value is 0x0, the condition is false, so the interface is forwarding; when the STP Blocked value is 0x1, the condition is true, so the interface is blocked.

Use the **show uidb data** command in order to confirm this and display the interface data that is present in the network processor:

<#root>

```
RP/0/RSP1/CPU0:router2#sh uidb data location 0/1/CPU0 TenGigE0/1/0/0.2
   ingress | i Blocked

STP Blocked

0x0

RP/0/RSP1/CPU0:router2#sh uidb data location 0/1/CPU0 TenGigE0/1/0/0.3
   ingress | i Blocked
```

STP Blocked

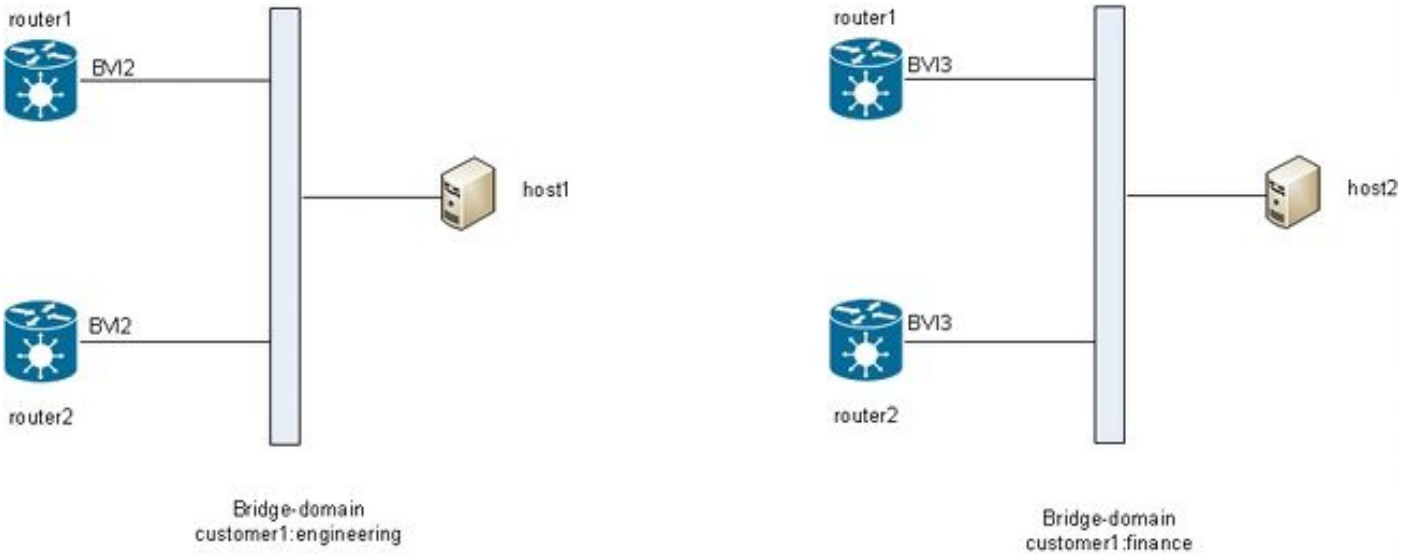
0x1

4.3 BVI

Configuration of a bridge-domain creates an L2 domain. In order to exit that L2 domain, connect L3 routers that route between hosts inside the bridge-domain and the outside world. In the previous diagram, host1 could use router4 or router5 in order to exit the local subnet and reach the internet.

Router1 and router2 where the bridge-domains are configured are ASR 9000 routers, which can route IPv4 and IPv6 traffic. So these two routers could take the IP traffic out of the bridge-domain and route it to the internet themselves, instead of relying on L3 routers. To do this, you need to configure a BVI, which is an L3 interface that plugs into a bridge-domain in order to route packets in and out of the bridge-domain.

This is how it looks like logically:



This is the configuration:

```
RP/0/RSP1/CPU0:router1#sh run int bvi 2
interface BVI2
 ipv4 address 192.168.2.1 255.255.255.0
!

RP/0/RSP1/CPU0:router1#sh run int bvi 3
interface BVI3
 ipv4 address 192.168.3.1 255.255.255.0
!
```

```
RP/0/RSP1/CPU0:router1#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface TenGigE0/0/0/1.3
   !
   interface GigabitEthernet0/0/0/1.3
   !
   routed interface BVI3
  !
  bridge-domain engineering
   interface TenGigE0/0/0/1.2
   !
   interface GigabitEthernet0/0/0/1.2
   !
   routed interface BVI2
  !
 !
!

RP/0/RSP1/CPU0:router1#sh run int gig 0/0/0/1.2
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
```

A BVI is an untagged L3 interface, so if you want to have the BVI process the packets received on the ACs of the bridge-domain, the ACs must be configured to pop all incoming tags. Otherwise, the BVI cannot understand the tag and drops the packets. There is no way to configure a dot1q subinterface on a BVI, so the tags must be popped ingress on the ACs as was done on Gi0/0/0/1.2 in the previous example.

Since a BVI interface is a virtual interface, there are some restrictions on the features that can be enabled. These restrictions are documented in Configuring Integrated Routing and Bridging on the Cisco ASR 9000 Series Router: Restrictions for Configuring IRB. These features are not supported on the BVI interfaces on the ASR 9000:

- Access Control Lists (ACLs). However, L2 ACLs can be configured on each L2 port of the bridge-domain.
- IP Fast Reroute (FRR)
- NetFlow
- MoFRR (Multicast only Fast Re-Route)
- MPLS label switching
- mVPNv4
- Quality of Service (QoS)
- Traffic mirroring
- Unnumbered interface for BVI
- Video monitoring (Vidmon)

The BVI can be in a Virtual Routing and Forwarding (VRF) configuration, so that traffic received on the BVI is forwarded over MPLS, but per-vrf label-allocation-mode must be used.

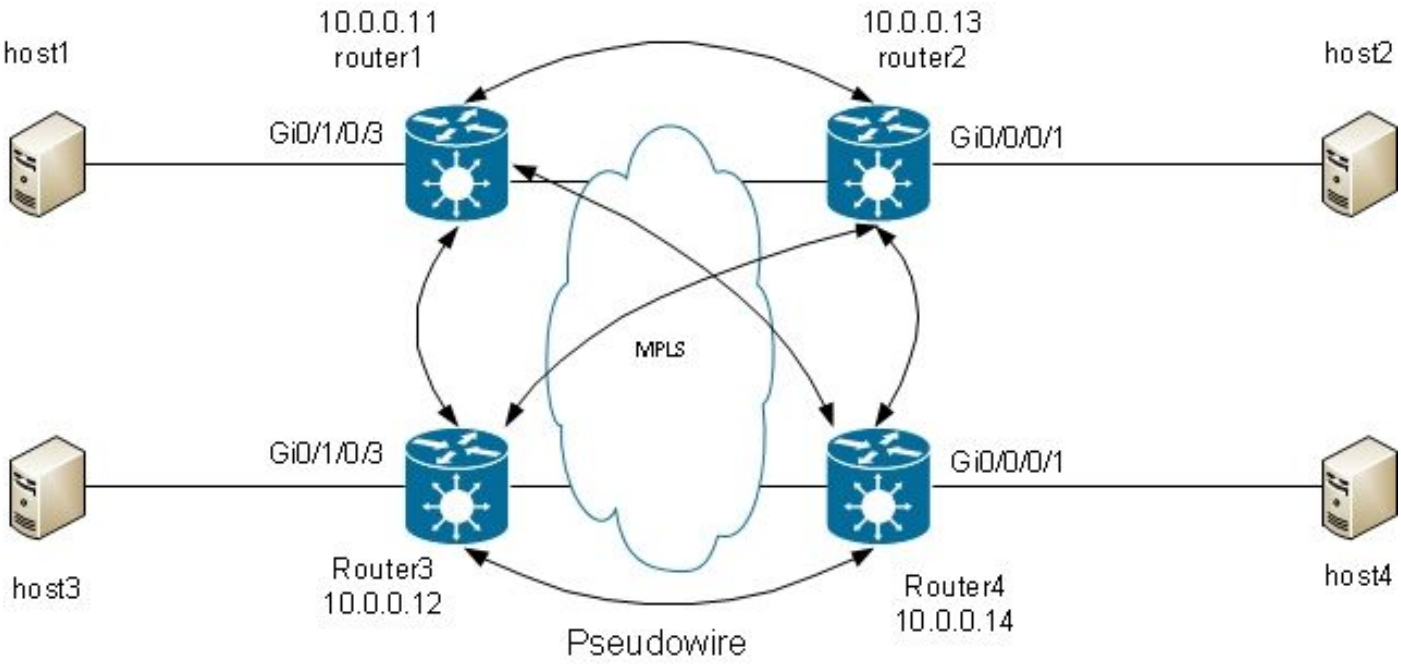
If one of these restricted features is required, you cannot use a BVI. Another solution is to use an external

loopback cable between two ports on the router, where one port is in the bridge-domain and one port is configured as a normal routed interface where all features can be configured.

4.4 VPLS

4.4.1 Overview

VPLS provides the ability to combine bridge-domains at multiple sites into one large bridge-domain through MPLS PWs. Hosts at the different sites appear to be directly connected to the same L2 segment because their traffic is transparently encapsulated over the full mesh of MPLS PWs between L2VPN PEs:



A full mesh of PWs is required in order to ensure that each host can receive traffic from all other hosts. The consequence is that a L2VPN PE does not forward a frame received on a VPLS PW over its other VPLS PWs. There can be a full mesh of PWs, so each PE receives the traffic directly and does not need to forward traffic between PWs since forwarding would cause a loop. This is called the split horizon rule.

The router is running MAC learning. Once a MAC address is present in the mac-address-table, you forward only the frame for that destination MAC address over the PW to the L2VPN PE where this MAC address has been learned from. This avoids unnecessary duplication of traffic in the core. Broadcasts and multicasts are flooded over all PWs in order to ensure that all hosts can receive them. A feature such as IGMP snooping is useful because it allows multicast frames to be sent to PEs only where there are receivers or multicast routers. This reduces the amount of traffic in the core, although there are still multiple copies of the same packets that must be sent to each PE when there is interest for that group.

The full mesh of PWs must be configured under a Virtual Forwarding Instance (VFI):

```
RP/0/RSP0/CPU0:router1#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/1/0/3.3
   !
   vfi customer1-finance
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.13 pw-id 3
    !
    neighbor 10.0.0.14 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!
```

The PWs configured under the VFI are those that are fully meshed in the core. They are part of the same split horizon group (SHG) in order to make sure that frames received on one PW are not forwarded to another PW.

It is possible to configure access PWs, which are considered a type of AC and are not configured under the VFI. See the H-VPLS section for details.

The configuration on router2, router3, and router4 is very similar, and all have the other three routers as neighbors under the VFI.

```
RP/0/RSP0/CPU0:router1#sh l2vpn bridge-domain bd-name engineering detail
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
```

```
MAC limit reached: no
MAC port down flush: enabled
MAC Secure: disabled, Logging: disabled
Split Horizon Group: none
Dynamic ARP Inspection: disabled, Logging: disabled
IP Source Guard: disabled, Logging: disabled
DHCPv4 snooping: disabled
IGMP Snooping profile: none
Bridge MTU: 1500
MIB cvplsConfigIndex: 6
Filter MAC addresses:
Create time: 28/05/2013 17:17:03 (23:06:02 ago)
No status change since creation
ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
List of ACs:
  AC: GigabitEthernet0/1/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1500; XC ID 0xc40003; interworking none
    MAC learning: enabled
    Flooding:
      Broadcast & Multicast: enabled
      Unknown unicast: enabled
    MAC aging time: 300 s, Type: inactivity
    MAC limit: 4000, Action: none, Notification: syslog
    MAC limit reached: no
    MAC port down flush: enabled
    MAC Secure: disabled, Logging: disabled
    Split Horizon Group: none
    Dynamic ARP Inspection: disabled, Logging: disabled
    IP Source Guard: disabled, Logging: disabled
    DHCPv4 snooping: disabled
    IGMP Snooping profile: none
    Storm Control: disabled
    Static MAC addresses:
    Statistics:
      packets: received 234039, sent 7824
      bytes: received 16979396, sent 584608
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic ARP inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0
List of Access PWs:
List of VFIs:
  VFI customer1-engineering (up)
    PW: neighbor 10.0.0.12, PW ID 2, state is up ( established )
      PW class not set, XC ID 0xc0000009
      Encapsulation MPLS, protocol LDP
      Source address 10.0.0.11
      PW type Ethernet, control word disabled, interworking none
      PW backup disable delay 0 sec
      Sequencing not set

      PW Status TLV in use
        MPLS         Local                          Remote
        ------------ ------------------------------ ----------------------
        Label        16049                          16042
        Group ID     0x5                            0x1
        Interface    customer1-engineering          customer1-engineering
```

```
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ----------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 3221225481
  Create time: 29/05/2013 15:36:17 (00:46:49 ago)
  Last time status changed: 29/05/2013 15:57:36 (00:25:29 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 555, sent 285
    bytes: received 36308, sent 23064
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.13, PW ID 2, state is up ( established )
  PW class not set, XC ID 0xc000000a
  Encapsulation MPLS, protocol LDP
  Source address 10.0.0.11
  PW type Ethernet, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

  PW Status TLV in use
    MPLS         Local                          Remote
    ------------ ------------------------------ ----------------------
    Label        16050                          16040
    Group ID     0x5                            0x3
    Interface    customer1-engineering          customer1-engineering
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ----------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 3221225482
  Create time: 29/05/2013 15:36:17 (00:46:49 ago)
  Last time status changed: 29/05/2013 16:00:56 (00:22:09 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 184, sent 158
    bytes: received 12198, sent 14144
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.14, PW ID 2, state is up ( established )
  PW class not set, XC ID 0xc000000b
  Encapsulation MPLS, protocol LDP
  Source address 10.0.0.11
  PW type Ethernet, control word disabled, interworking none
  PW backup disable delay 0 sec
```

```
    Sequencing not set

    PW Status TLV in use
      MPLS         Local                          Remote
      ------------ ------------------------------ ----------------------
      Label        16051                          289974
      Group ID     0x5                            0x6
      Interface    customer1-engineering          customer1-engineering
      MTU          1500                           1500
      Control word disabled                       disabled
      PW type      Ethernet                       Ethernet
      VCCV CV type 0x2                            0x2
                   (LSP ping verification)        (LSP ping verification)
      VCCV CC type 0x6                            0x6
                   (router alert label)           (router alert label)
                   (TTL expiry)                   (TTL expiry)
      ------------ ------------------------------ ----------------------
    Incoming Status (PW Status TLV):
      Status code: 0x0 (Up) in Notification message
    MIB cpwVcIndex: 3221225483
    Create time: 29/05/2013 15:36:17 (00:46:49 ago)
    Last time status changed: 29/05/2013 16:02:38 (00:20:27 ago)
    MAC withdraw message: send 0 receive 0
    Static MAC addresses:
    Statistics:
      packets: received 0, sent 137
      bytes: received 0, sent 12064
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  VFI Statistics:
    drops: illegal VLAN 0, illegal length 0
```

The local label for the PW to 10.0.0.12 is 16049, which means that Ethernet frames are received with the label 16049. The switching decision is based on this MPLS label because the penultimate MPLS hop can have popped the IGP label. There can still be an explicit null label, but the switching decision is based on the PW label:

<#root>

```
RP/0/RSP0/CPU0:router1#sh mpls forwarding labels
```

16049

```
Local  Outgoing    Prefix             Outgoing     Next Hop        Bytes
Label  Label       or ID              Interface                    Switched
------ ----------- ------------------ ------------ --------------- ----------
```

16049

```
  Pop         PW(
```

10.0.0.12

```
:2)    BD=5         point2point     58226
```

The **show mpls forwarding labels** command for the label gives the bridge-domain number, which you can

use in order to find the destination mac-address and the PW (neighbor and pw-id) where the packet was received. You can then create entries in the mac-address-table that point at that neighbor:

```
RP/0/RSP0/CPU0:router1#sh l2vpn forwarding bridge-domain customer1:
   engineering mac-address location 0/1/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age  Mapped to
--------------------------------------------------------------------------------
0019.552b.b5c3 dynamic Gi0/1/0/3.2              0/1/CPU0   0d 0h 0m 0s N/A
0024.985e.6a01 dynamic (10.0.0.12, 2)           0/1/CPU0   0d 0h 0m 0s N/A
0024.985e.6a42 dynamic (10.0.0.12, 2)           0/1/CPU0   0d 0h 0m 0s N/A
001d.4603.1f42 dynamic (10.0.0.13, 2)           0/1/CPU0   0d 0h 0m 0s N/A
```

4.4.2 PW Types and Transported Tags

VPLS PWs are negotiated as type 5 (Ethernet) PWs by default. Whatever comes into the AC after any VLAN tag manipulation (when the **rewrite** command is configured) is sent over the PW.

Cisco IOS XR Software Release 4.1.0 for LDP signaling and Release 4.3.1 with BGP let you configure a pw-class under a neighbor and configure **transport mode vlan passthrough** under the pw-class. This negotiates a virtual connection (VC)-type 4 (Ethernet VLAN) PW, which transports whatever comes out of the AC after the VLAN tag manipulation when the **rewrite** command is configured.

The VLAN tag manipulation on the EFP ensures that there is at least one VLAN tag left on the frame because you need a dot1q tag on the frame if there are VC-type 4 PWs. No dummy tag 0 is added to the frame when you use the transport mode vlan passthrough mode.

A mix of type 4 and type 5 PWs under the same VFI is not supported. All PWs must be of the same type.

```
RP/0/RSP0/CPU0:router1#sh run l2vpn bridge group customer1 bridge-domain
   engineering
l2vpn
 bridge group customer1
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.12 pw-id 2
     pw-class VC4-PT
    !
    neighbor 10.0.0.13 pw-id 2
     pw-class VC4-PT
    !
    neighbor 10.0.0.14 pw-id 2
     pw-class VC4-PT
```

```
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:router1#sh l2vpn bridge-domain bd-name engineering detail |
   i "PW:|PW type"
    MAC withdraw for Access PW: enabled
      PW: neighbor 10.0.0.12, PW ID 2, state is up ( established )
        PW type Ethernet VLAN, control word disabled, interworking none
          PW type      Ethernet VLAN                  Ethernet VLAN
      PW: neighbor 10.0.0.13, PW ID 2, state is up ( established )
        PW type Ethernet VLAN, control word disabled, interworking none
          PW type      Ethernet VLAN                  Ethernet VLAN
      PW: neighbor 10.0.0.14, PW ID 2, state is up ( established )
        PW type Ethernet VLAN, control word disabled, interworking none
          PW type      Ethernet VLAN                  Ethernet VLAN
```

4.4.3 Autodiscovery and Signaling

The previous examples were based on manual configuration of all the neighbors under the VFI. MPLS LDP was used for the signaling of the PW with the neighbor.

When you add a new VPLS PE to the network, configure the PE in order to have a PW to all existing PEs in each of its local bridge-domains. All existing PEs must then be reconfigured in order to have a PW to the new PE because all PEs must be fully meshed. This can become an operational challenge as the number of PEs and bridge-domains increase.

One solution is to have PEs discover other PEs automatically through BGP. While there is also a full-mesh requirement for IBGP, it can be lifted by the use of route-reflectors. So, a new PE is typically configured in order to peer with a small number of route-reflectors, all other PEs receive its updates, and the new PE receives the updates from the other PEs.

In order to discover other PEs through BGP, each PE is configured for the vpls-vpws address-family and advertises in BGP the bridge-domains in which they want to participate. Once the other PEs that are part of the same bridge-domain are discovered, a PW is established to each of them. BGP is the protocol used for this autodiscovery.

There are two options for the signaling of the PW to the autodiscovered PEs: BGP and LDP. In these examples, you convert the previous topology to BGP autodiscovery with BGP signaling and LDP signaling.

4.4.3.1 BGP Autodiscovery and BGP Signaling

Configure the **address-family l2vpn vpls-vpws** under router bgp and the neighbors, which are other PEs or the route-reflectors:

```
router bgp 65000
 address-family l2vpn vpls-vpws
 !
 neighbor-group IOX-LAB-RR
  address-family l2vpn vpls-vpws
  !
 neighbor 10.0.0.3
  use neighbor-group IOX-LAB-RR
 !
 neighbor 10.0.0.10
  use neighbor-group IOX-LAB-RR
 !
```

The new address-family becomes active with the neighbors, but no PE has yet advertised its participation in a bridge-domain:

```
RP/0/RSP0/CPU0:router1#sh bgp neighbor 10.0.0.3 | i Address family L2VPN
    Address family L2VPN VPLS: advertised and received

P/0/RSP0/CPU0:router1#sh bgp l2vpn vpls summary
BGP router identifier 10.0.0.11, local AS number 65000
BGP generic scan interval 60 secs
BGP table state: Active
Table ID: 0x0   RD version: 3890838096
BGP main routing table version 77
BGP scan interval 60 secs

BGP is operating in STANDALONE mode.


Process      RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker             77         77         77         77          77          77

Neighbor       Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.0.3         0 65000  252950   53252       77    0    0     1w0d          0
10.0.0.10        0 65000  941101   47439       77    0    0 00:10:18          0
```

Configure **autodiscovery bgp** and **signaling-protocol bgp** under the L2VPN bridge-domain configuration mode. The configuration on router1 is:

<#root>

```
RP/0/RSP0/CPU0:router1#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/1/0/3.3
   !
   vfi customer1-finance
    vpn-id 3
```

autodiscovery bgp

```
     rd auto
     route-target 0.0.0.1:3

signaling-protocol bgp


      ve-id 11
     !
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   vfi customer1-engineering
    vpn-id 2

autodiscovery bgp


     rd auto
     route-target 0.0.0.1:2

signaling-protocol bgp


      ve-id 11
     !
    !
   !
  !
 !
!
```

The configuration on router2 is:

<#root>

```
RP/0/RSP1/CPU0:router2#sh run l2vpn bridge group customer1
Thu May 30 15:25:55.638 CEST
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/0/0/1.3
   !
   vfi customer1-finance
    vpn-id 3

autodiscovery bgp


     rd auto
     route-target 0.0.0.1:3
```

```
signaling-protocol bgp


      ve-id 13
     !
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/0/0/1.2
   !
   vfi customer1-engineering
    vpn-id 2


autodiscovery bgp


     rd auto
     route-target 0.0.0.1:2


signaling-protocol bgp


      ve-id 13
     !
    !
   !
  !
 !
!
```

The vpn-id and the route-target are the same on the different PEs for each bridge-domain, but each PE has a unique Virtual Edge Identifier (VE-ID). Each PE discovers the other PEs in the VPN through BGP and uses BGP in order to signal the PWs. The result is a full mesh of PWs:

```
RP/0/RSP0/CPU0:router1#sh bgp l2vpn vpls summary
BGP router identifier 10.0.0.11, local AS number 65000
BGP generic scan interval 60 secs
BGP table state: Active
Table ID: 0x0   RD version: 3890838096
BGP main routing table version 103
BGP scan interval 60 secs

BGP is operating in STANDALONE mode.


Process   RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker         103        103        103        103         103         103

Neighbor    Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.0.3      0 65000  254944   53346      103    0    0     1w0d          6
10.0.0.10     0 65000  944859   47532      103    0    0 01:40:22          6

RP/0/RSP0/CPU0:router1#sh bgp l2vpn vpls
BGP router identifier 10.0.0.11, local AS number 65000
BGP generic scan interval 60 secs
```

```
BGP table state: Active
Table ID: 0x0   RD version: 3890838096
BGP main routing table version 103
BGP scan interval 60 secs

Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop        Rcvd Label      Local Label
Route Distinguisher: 10.0.0.11:32769 (default for vrf customer1:finance)
*> 11:10/32           0.0.0.0         nolabel         16060
*>i12:10/32           10.0.0.12       16060           nolabel
*>i13:10/32           10.0.0.13       16060           nolabel
*>i14:10/32           10.0.0.14       289959          nolabel
Route Distinguisher: 10.0.0.11:32770 (default for vrf customer1:engineering)
*> 11:10/32           0.0.0.0         nolabel         16075
*>i12:10/32           10.0.0.12       16075           nolabel
*>i13:10/32           10.0.0.13       16075           nolabel
*>i14:10/32           10.0.0.14       289944          nolabel
Route Distinguisher: 10.0.0.12:32768
*>i12:10/32           10.0.0.12       16060           nolabel
* i                    10.0.0.12       16060           nolabel
Route Distinguisher: 10.0.0.12:32769
*>i12:10/32           10.0.0.12       16075           nolabel
* i                    10.0.0.12       16075           nolabel
Route Distinguisher: 10.0.0.13:32769
*>i13:10/32           10.0.0.13       16060           nolabel
* i                    10.0.0.13       16060           nolabel
Route Distinguisher: 10.0.0.13:32770
*>i13:10/32           10.0.0.13       16075           nolabel
* i                    10.0.0.13       16075           nolabel
Route Distinguisher: 10.0.0.14:32768
*>i14:10/32           10.0.0.14       289959          nolabel
* i                    10.0.0.14       289959          nolabel
Route Distinguisher: 10.0.0.14:32769
*>i14:10/32           10.0.0.14       289944          nolabel
* i                    10.0.0.14       289944          nolabel

Processed 14 prefixes, 20 paths
```

These are the prefixes advertised by router3 (10.0.0.13) as seen on router1; the prefixes are received through the two route-reflectors, 10.0.0.3 and 10.0.0.10:

```
RP/0/RSP0/CPU0:router1#sh bgp l2vpn vpls rd 10.0.0.13:32770 13:10/32
BGP routing table entry for 13:10/32, Route Distinguisher: 10.0.0.13:32770
Versions:
  Process           bRIB/RIB  SendTblVer
  Speaker                92          92
Last Modified: May 30 15:10:44.100 for 01:23:38
Paths: (2 available, best #1)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Not advertised to any peer
  Local
    10.0.0.13 (metric 5) from 10.0.0.3 (10.0.0.13)
      Received Label 16075
      Origin IGP, localpref 100, valid, internal, best, group-best,
```

```
         import-candidate, not-in-vrf, import suspect
       Received Path ID 0, Local Path ID 1, version 92
       Extended community: RT:0.0.0.1:2 L2VPN:19:0:1500
       Originator: 10.0.0.13, Cluster list: 10.0.0.3
       Block Size:10
  Path #2: Received by speaker 0
  Not advertised to any peer
  Local
    10.0.0.13 (metric 5) from 10.0.0.10 (10.0.0.13)
      Received Label 16075
      Origin IGP, localpref 100, valid, internal, not-in-vrf, import suspect
      Received Path ID 0, Local Path ID 0, version 0
      Extended community: RT:0.0.0.1:2 L2VPN:19:0:1500
      Originator: 10.0.0.13, Cluster list: 10.0.0.10
      Block Size:10

RP/0/RSP0/CPU0:router1#sh bgp l2vpn vpls rd 10.0.0.13:32769 13:10/32
BGP routing table entry for 13:10/32, Route Distinguisher: 10.0.0.13:32769
Versions:
  Process           bRIB/RIB  SendTblVer
  Speaker                 93          93
Last Modified: May 30 15:10:44.100 for 01:25:02
Paths: (2 available, best #1)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Not advertised to any peer
  Local
    10.0.0.13 (metric 5) from 10.0.0.3 (10.0.0.13)
      Received Label 16060
      Origin IGP, localpref 100, valid, internal, best, group-best,
         import-candidate, not-in-vrf, import suspect
      Received Path ID 0, Local Path ID 1, version 93
      Extended community: RT:0.0.0.1:3 L2VPN:19:0:1500
      Originator: 10.0.0.13, Cluster list: 10.0.0.3
      Block Size:10
  Path #2: Received by speaker 0
  Not advertised to any peer
  Local
    10.0.0.13 (metric 5) from 10.0.0.10 (10.0.0.13)
      Received Label 16060
      Origin IGP, localpref 100, valid, internal, not-in-vrf, import suspect
      Received Path ID 0, Local Path ID 0, version 0
      Extended community: RT:0.0.0.1:3 L2VPN:19:0:1500
      Originator: 10.0.0.13, Cluster list: 10.0.0.10
      Block Size:10
```

Router1 has established some PWs:

```
RP/0/RSP0/CPU0:router1#sh l2vpn discovery bridge-domain

Service Type: VPLS,  Connected
  List of VPNs (2 VPNs):
  Bridge group: customer1, bridge-domain: finance, id: 3, signaling
     protocol: BGP
    List of Local Edges (1 Edges):
      Local Edge ID: 11, Label Blocks (1 Blocks)
        Label base        Offset    Size   Time Created
```

```
        ----------        ------    ----   -------------------
        16060             10        10     05/30/2013 15:07:39
    List of Remote Edges (3 Edges):
      Remote Edge ID: 12, NLRIs (1 NLRIs)
        Label base        Offset    Size   Peer ID      Time Created
        ----------        ------    ----   ------------ -------------------
        16060             10        10     10.0.0.12    05/30/2013 15:09:53
      Remote Edge ID: 13, NLRIs (1 NLRIs)
        Label base        Offset    Size   Peer ID      Time Created
        ----------        ------    ----   ------------ -------------------
        16060             10        10     10.0.0.13    05/30/2013 15:10:43
      Remote Edge ID: 14, NLRIs (1 NLRIs)
        Label base        Offset    Size   Peer ID      Time Created
        ----------        ------    ----   ------------ -------------------
        289959            10        10     10.0.0.14    05/30/2013 15:11:22

  Bridge group: customer1, bridge-domain: engineering, id: 5, signaling
     protocol: BGP
    List of Local Edges (1 Edges):
      Local Edge ID: 11, Label Blocks (1 Blocks)
        Label base        Offset    Size   Time Created
        ----------        ------    ----   -------------------
        16075             10        10     05/30/2013 15:08:54
    List of Remote Edges (3 Edges):
      Remote Edge ID: 12, NLRIs (1 NLRIs)
        Label base        Offset    Size   Peer ID      Time Created
        ----------        ------    ----   ------------ -------------------
        16075             10        10     10.0.0.12    05/30/2013 15:09:53
      Remote Edge ID: 13, NLRIs (1 NLRIs)
        Label base        Offset    Size   Peer ID      Time Created
        ----------        ------    ----   ------------ -------------------
        16075             10        10     10.0.0.13    05/30/2013 15:10:43
      Remote Edge ID: 14, NLRIs (1 NLRIs)
        Label base        Offset    Size   Peer ID      Time Created
        ----------        ------    ----   ------------ -------------------
        289944            10        10     10.0.0.14    05/30/2013 15:11:22

RP/0/RSP0/CPU0:router1#sh l2vpn bridge-domain autodiscovery bgp
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of VFIs:
    VFI customer1-finance (up)
      Neighbor 10.0.0.12 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 3, state: up, Static MAC addresses: 0
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.12 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 2, state: up, Static MAC addresses: 0

RP/0/RSP0/CPU0:router1#sh l2vpn bridge-domain group customer1
Legend: pp = Partially Programmed.
```

```
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.3, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-finance (up)
      Neighbor 10.0.0.12 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 3, state: up, Static MAC addresses: 0
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.2, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.12 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 2, state: up, Static MAC addresses: 0

RP/0/RSP0/CPU0:router1#sh l2vpn bridge-domain group customer1 detail
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
  Split Horizon Group: none
  Dynamic ARP Inspection: disabled, Logging: disabled
  IP Source Guard: disabled, Logging: disabled
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
  Bridge MTU: 1500
  MIB cvplsConfigIndex: 4
  Filter MAC addresses:
  Create time: 29/05/2013 15:36:17 (1d01h ago)
  No status change since creation
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    AC: GigabitEthernet0/1/0/3.3, state is up
      Type VLAN; Num Ranges: 1
      VLAN ranges: [3, 3]
      MTU 1500; XC ID 0xc40006; interworking none
      MAC learning: enabled
      Flooding:
```

```
      Broadcast & Multicast: enabled
      Unknown unicast: enabled
    MAC aging time: 300 s, Type: inactivity
    MAC limit: 4000, Action: none, Notification: syslog
    MAC limit reached: no
    MAC port down flush: enabled
    MAC Secure: disabled, Logging: disabled
    Split Horizon Group: none
    Dynamic ARP Inspection: disabled, Logging: disabled
    IP Source Guard: disabled, Logging: disabled
    DHCPv4 snooping: disabled
    IGMP Snooping profile: none
    Storm Control: disabled
    Static MAC addresses:
    Statistics:
      packets: received 10120, sent 43948
      bytes: received 933682, sent 2989896
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic ARP inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0
List of Access PWs:
List of VFIs:
  VFI customer1-finance (up)
    VPN-ID: 3, Auto Discovery: BGP, state is Provisioned
       (Service Connected)
    Route Distinguisher:  (auto) 10.0.0.11:32769
    Import Route Targets:
      0.0.0.1:3
    Export Route Targets:
      0.0.0.1:3
    Signaling protocol: BGP
    Local VE-ID: 11 ,  Advertised Local VE-ID : 11
    VE-Range: 10
    PW: neighbor 10.0.0.12, PW ID 3, state is up ( established )
      PW class not set, XC ID 0xc000000c
      Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
      Source address 10.0.0.11
      PW type VPLS, control word disabled, interworking none
      PW backup disable delay 0 sec
      Sequencing not set

        MPLS         Local                          Remote
        ------------ ------------------------------ ------------------
        Label        16062                          16061
        MTU          1500                           1500
        Control word disabled                       disabled
        PW type      VPLS                           VPLS
        VE-ID        11                             12
        ------------ ------------------------------ ------------------
      MIB cpwVcIndex: 3221225484
      Create time: 30/05/2013 15:09:52 (01:29:44 ago)
      Last time status changed: 30/05/2013 15:09:52 (01:29:44 ago)
      MAC withdraw message: send 0 receive 0
      Static MAC addresses:
      Statistics:
        packets: received 2679, sent 575
        bytes: received 171698, sent 51784
    DHCPv4 snooping: disabled
```

```
      IGMP Snooping profile: none
      PW: neighbor 10.0.0.13, PW ID 3, state is up ( established )
        PW class not set, XC ID 0xc000000e
        Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
        Source address 10.0.0.11
        PW type VPLS, control word disabled, interworking none
        PW backup disable delay 0 sec
        Sequencing not set

          MPLS         Local                          Remote
          ------------ ------------------------------ ------------------
          Label        16063                          16061
          MTU          1500                           1500
          Control word disabled                       disabled
          PW type      VPLS                           VPLS
          VE-ID        11                             13
          ------------ ------------------------------ ------------------
        MIB cpwVcIndex: 3221225486
        Create time: 30/05/2013 15:10:43 (01:28:54 ago)
        Last time status changed: 30/05/2013 15:10:43 (01:28:54 ago)
        MAC withdraw message: send 0 receive 0
        Static MAC addresses:
        Statistics:
          packets: received 11, sent 574
          bytes: received 1200, sent 51840
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      PW: neighbor 10.0.0.14, PW ID 3, state is up ( established )
        PW class not set, XC ID 0xc0000010
        Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
        Source address 10.0.0.11
        PW type VPLS, control word disabled, interworking none
        PW backup disable delay 0 sec
        Sequencing not set

          MPLS         Local                          Remote
          ------------ ------------------------------ ------------------
          Label        16064                          289960
          MTU          1500                           1500
          Control word disabled                       disabled
          PW type      VPLS                           VPLS
          VE-ID        11                             14
          ------------ ------------------------------ ------------------
        MIB cpwVcIndex: 3221225488
        Create time: 30/05/2013 15:11:22 (01:28:15 ago)
        Last time status changed: 30/05/2013 15:11:22 (01:28:15 ago)
        MAC withdraw message: send 0 receive 0
        Static MAC addresses:
        Statistics:
          packets: received 0, sent 561
          bytes: received 0, sent 50454
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      VFI Statistics:
        drops: illegal VLAN 0, illegal length 0
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
```

```
Flooding:
  Broadcast & Multicast: enabled
  Unknown unicast: enabled
MAC aging time: 300 s, Type: inactivity
MAC limit: 4000, Action: none, Notification: syslog
MAC limit reached: no
MAC port down flush: enabled
MAC Secure: disabled, Logging: disabled
Split Horizon Group: none
Dynamic ARP Inspection: disabled, Logging: disabled
IP Source Guard: disabled, Logging: disabled
DHCPv4 snooping: disabled
IGMP Snooping profile: none
Bridge MTU: 1500
MIB cvplsConfigIndex: 6
Filter MAC addresses:
Create time: 28/05/2013 17:17:03 (1d23h ago)
No status change since creation
ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
List of ACs:
  AC: GigabitEthernet0/1/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1500; XC ID 0xc40007; interworking none
    MAC learning: enabled
    Flooding:
      Broadcast & Multicast: enabled
      Unknown unicast: enabled
    MAC aging time: 300 s, Type: inactivity
    MAC limit: 4000, Action: none, Notification: syslog
    MAC limit reached: no
    MAC port down flush: enabled
    MAC Secure: disabled, Logging: disabled
    Split Horizon Group: none
    Dynamic ARP Inspection: disabled, Logging: disabled
    IP Source Guard: disabled, Logging: disabled
    DHCPv4 snooping: disabled
    IGMP Snooping profile: none
    Storm Control: disabled
    Static MAC addresses:
    Statistics:
      packets: received 243532, sent 51089
      bytes: received 17865888, sent 3528732
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic ARP inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0
List of Access PWs:
List of VFIs:
  VFI customer1-engineering (up)
    VPN-ID: 2, Auto Discovery: BGP, state is Provisioned
       (Service Connected)
    Route Distinguisher:  (auto) 10.0.0.11:32770
    Import Route Targets:
      0.0.0.1:2
    Export Route Targets:
      0.0.0.1:2
    Signaling protocol: BGP
    Local VE-ID: 11 ,  Advertised Local VE-ID : 11
```

```
VE-Range: 10
PW: neighbor 10.0.0.12, PW ID 2, state is up ( established )
  PW class not set, XC ID 0xc000000d
  Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
  Source address 10.0.0.11
  PW type VPLS, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------
    Label        16077                          16076
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      VPLS                           VPLS
    VE-ID        11                             12
    ------------ ------------------------------ ------------------
  MIB cpwVcIndex: 3221225485
  Create time: 30/05/2013 15:09:52 (01:29:45 ago)
  Last time status changed: 30/05/2013 15:09:52 (01:29:45 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 2677, sent 574
    bytes: received 171524, sent 51670
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.13, PW ID 2, state is up ( established )
  PW class not set, XC ID 0xc000000f
  Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
  Source address 10.0.0.11
  PW type VPLS, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------
    Label        16078                          16076
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      VPLS                           VPLS
    VE-ID        11                             13
    ------------ ------------------------------ ------------------
  MIB cpwVcIndex: 3221225487
  Create time: 30/05/2013 15:10:43 (01:28:54 ago)
  Last time status changed: 30/05/2013 15:10:43 (01:28:54 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 17, sent 572
    bytes: received 1560, sent 51636
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.14, PW ID 2, state is up ( established )
  PW class not set, XC ID 0xc0000011
  Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
  Source address 10.0.0.11
  PW type VPLS, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

    MPLS         Local                          Remote
```

```
          ------------ ------------------------------ ------------------
          Label        16079                          289945
          MTU          1500                           1500
          Control word disabled                       disabled
          PW type      VPLS                           VPLS
          VE-ID        11                             14
          ------------ ------------------------------ ------------------
        MIB cpwVcIndex: 3221225489
        Create time: 30/05/2013 15:11:22 (01:28:16 ago)
        Last time status changed: 30/05/2013 15:11:22 (01:28:16 ago)
        MAC withdraw message: send 0 receive 0
        Static MAC addresses:
        Statistics:
          packets: received 0, sent 559
          bytes: received 0, sent 50250
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      VFI Statistics:
        drops: illegal VLAN 0, illegal length 0
```

4.4.3.2 BGP Autodiscovery and LDP Signaling

The BGP configuration with the **address-family l2vpn vpls-vpws** command is exactly the same as with BGP signaling. The L2VPN configuration is modified in order to use LDP signaling with the **signaling-protocol ldp** command.

The same configuration is used on all four PEs:

<#root>

```
router bgp 65000
 address-family l2vpn vpls-vpws
 !
 neighbor-group IOX-LAB-RR
  address-family l2vpn vpls-vpws
  !
 neighbor 10.0.0.3
  use neighbor-group IOX-LAB-RR
 !
 neighbor 10.0.0.10
  use neighbor-group IOX-LAB-RR
 !
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/1/0/3.3
   !
   vfi customer1-finance
    vpn-id 3
    autodiscovery bgp
     rd auto
     route-target 0.0.0.1:3
     signaling-protocol ldp
      vpls-id 65000:3
     !
```

```
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   vfi customer1-engineering
    vpn-id 2
    autodiscovery bgp
     rd auto
     route-target 0.0.0.1:2
```

```
signaling-protocol ldp
```

```
      vpls-id 65000:2
```

```
     !
    !
   !
  !
 !
!
```

The vpls-id is made of the BGP Autonomous System (AS) number and the vpn-id.

Three show commands from router1 illustrate that the PWs have been established with the discovered PEs:

<#root>

```
RP/0/RSP0/CPU0:router1#
```

```
sh l2vpn discovery
```

```
Service Type: VPLS,  Connected
  List of VPNs (2 VPNs):
  Bridge group: customer1, bridge-domain: finance, id: 3,
     signaling protocol: LDP
    VPLS-ID: 65000:3
    Local L2 router id: 10.0.0.11
    List of Remote NLRI (3 NLRIs):
    Local Addr      Remote Addr     Remote L2 RID   Time Created
    --------------- --------------- --------------- -------------------
    10.0.0.11       10.0.0.12       10.0.0.12       05/30/2013 17:10:18
    10.0.0.11       10.0.0.13       10.0.0.13       05/30/2013 17:10:18
    10.0.0.11       10.0.0.14       10.0.0.14       05/30/2013 17:11:46

  Bridge group: customer1, bridge-domain: engineering, id: 5,
     signaling protocol: LDP
    VPLS-ID: 65000:2
    Local L2 router id: 10.0.0.11
    List of Remote NLRI (3 NLRIs):
    Local Addr      Remote Addr     Remote L2 RID   Time Created
```

```
    -------------- -------------- -------------- -------------------
    10.0.0.11      10.0.0.12      10.0.0.12      05/30/2013 17:10:18
    10.0.0.11      10.0.0.13      10.0.0.13      05/30/2013 17:10:18
    10.0.0.11      10.0.0.14      10.0.0.14      05/30/2013 17:11:46

RP/0/RSP0/CPU0:router1#
```

sh l2vpn bridge-domain group customer1

```
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.3, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-finance (up)
      Neighbor 10.0.0.12 pw-id 65000:3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 65000:3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 65000:3, state: up, Static MAC addresses: 0
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.2, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.12 pw-id 65000:2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 65000:2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 65000:2, state: up, Static MAC addresses: 0

RP/0/RSP0/CPU0:router1#
```

sh l2vpn bridge-domain group customer1 det

```
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
  Split Horizon Group: none
  Dynamic ARP Inspection: disabled, Logging: disabled
  IP Source Guard: disabled, Logging: disabled
  DHCPv4 snooping: disabled
```

```
IGMP Snooping profile: none
Bridge MTU: 1500
MIB cvplsConfigIndex: 4
Filter MAC addresses:
Create time: 29/05/2013 15:36:17 (1d01h ago)
No status change since creation
ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
List of ACs:
  AC: GigabitEthernet0/1/0/3.3, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [3, 3]
    MTU 1500; XC ID 0xc40006; interworking none
    MAC learning: enabled
    Flooding:
      Broadcast & Multicast: enabled
      Unknown unicast: enabled
    MAC aging time: 300 s, Type: inactivity
    MAC limit: 4000, Action: none, Notification: syslog
    MAC limit reached: no
    MAC port down flush: enabled
    MAC Secure: disabled, Logging: disabled
    Split Horizon Group: none
    Dynamic ARP Inspection: disabled, Logging: disabled
    IP Source Guard: disabled, Logging: disabled
    DHCPv4 snooping: disabled
    IGMP Snooping profile: none
    Storm Control: disabled
    Static MAC addresses:
    Statistics:
      packets: received 10362, sent 45038
      bytes: received 956240, sent 3064016
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic ARP inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0
List of Access PWs:
List of VFIs:
  VFI customer1-finance (up)
    VPN-ID: 3, Auto Discovery: BGP, state is Provisioned
       (Service Connected)
    Route Distinguisher:  (auto) 10.0.0.11:32769
    Import Route Targets:
      0.0.0.1:3
    Export Route Targets:
      0.0.0.1:3
    Signaling protocol: LDP
    AS Number: 65000
    VPLS-ID: 65000:3
    L2VPN Router ID: 10.0.0.11
    PW: neighbor 10.0.0.12, PW ID 65000:3, state is up ( established )
      PW class not set, XC ID 0xc0000003
      Encapsulation MPLS, Auto-discovered (BGP), protocol LDP
      Source address 10.0.0.11
      PW type Ethernet, control word disabled, interworking none
      PW backup disable delay 0 sec
      Sequencing not set

      PW Status TLV in use
        MPLS         Local                          Remote
```

```
    ------------ ------------------------------ ------------------
    Label        16006                          16033
    BGP Peer ID  10.0.0.11                      10.0.0.12
    LDP ID       10.0.0.11                      10.0.0.12
    AII          10.0.0.11                      10.0.0.12
    AGI          65000:3                        65000:3
    Group ID     0x3                            0x0
    Interface    customer1-finance              customer1-finance
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 3221225475
  Create time: 30/05/2013 17:10:18 (00:06:32 ago)
  Last time status changed: 30/05/2013 17:10:24 (00:06:25 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 190, sent 40
    bytes: received 12160, sent 3600
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.13, PW ID 65000:3, state is up ( established )
  PW class not set, XC ID 0xc0000004
  Encapsulation MPLS, Auto-discovered (BGP), protocol LDP
  Source address 10.0.0.11
  PW type Ethernet, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

  PW Status TLV in use
    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------
    Label        16016                          16020
    BGP Peer ID  10.0.0.11                      10.0.0.13
    LDP ID       10.0.0.11                      10.0.0.13
    AII          10.0.0.11                      10.0.0.13
    AGI          65000:3                        65000:3
    Group ID     0x3                            0x4
    Interface    customer1-finance              customer1-finance
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 3221225476
  Create time: 30/05/2013 17:10:18 (00:06:32 ago)
  Last time status changed: 30/05/2013 17:10:27 (00:06:22 ago)
  MAC withdraw message: send 0 receive 0
```

```
        Static MAC addresses:
        Statistics:
          packets: received 0, sent 40
          bytes: received 0, sent 3600
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      PW: neighbor 10.0.0.14, PW ID 65000:3, state is up ( established )
        PW class not set, XC ID 0xc0000009
        Encapsulation MPLS, Auto-discovered (BGP), protocol LDP
        Source address 10.0.0.11
        PW type Ethernet, control word disabled, interworking none
        PW backup disable delay 0 sec
        Sequencing not set

        PW Status TLV in use
          MPLS         Local                          Remote
          ------------ ------------------------------ ------------------
          Label        16049                          289970
          BGP Peer ID  10.0.0.11                      10.0.0.14
          LDP ID       10.0.0.11                      10.0.0.14
          AII          10.0.0.11                      10.0.0.14
          AGI          65000:3                        65000:3
          Group ID     0x3                            0x4
          Interface    customer1-finance              customer1-finance
          MTU          1500                           1500
          Control word disabled                       disabled
          PW type      Ethernet                       Ethernet
          VCCV CV type 0x2                            0x2
                       (LSP ping verification)        (LSP ping verification)
          VCCV CC type 0x6                            0x6
                       (router alert label)           (router alert label)
                       (TTL expiry)                   (TTL expiry)
          ------------ ------------------------------ ------------------
        Incoming Status (PW Status TLV):
          Status code: 0x0 (Up) in Notification message
        MIB cpwVcIndex: 3221225481
        Create time: 30/05/2013 17:11:46 (00:05:04 ago)
        Last time status changed: 30/05/2013 17:11:51 (00:04:59 ago)
        MAC withdraw message: send 0 receive 0
        Static MAC addresses:
        Statistics:
          packets: received 0, sent 31
          bytes: received 0, sent 2790
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none
      VFI Statistics:
        drops: illegal VLAN 0, illegal length 0
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
```

```
Split Horizon Group: none
Dynamic ARP Inspection: disabled, Logging: disabled
IP Source Guard: disabled, Logging: disabled
DHCPv4 snooping: disabled
IGMP Snooping profile: none
Bridge MTU: 1500
MIB cvplsConfigIndex: 6
Filter MAC addresses:
Create time: 28/05/2013 17:17:03 (1d23h ago)
No status change since creation
ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
List of ACs:
  AC: GigabitEthernet0/1/0/3.2, state is up
    Type VLAN; Num Ranges: 1
    VLAN ranges: [2, 2]
    MTU 1500; XC ID 0xc40007; interworking none
    MAC learning: enabled
    Flooding:
      Broadcast & Multicast: enabled
      Unknown unicast: enabled
    MAC aging time: 300 s, Type: inactivity
    MAC limit: 4000, Action: none, Notification: syslog
    MAC limit reached: no
    MAC port down flush: enabled
    MAC Secure: disabled, Logging: disabled
    Split Horizon Group: none
    Dynamic ARP Inspection: disabled, Logging: disabled
    IP Source Guard: disabled, Logging: disabled
    DHCPv4 snooping: disabled
    IGMP Snooping profile: none
    Storm Control: disabled
    Static MAC addresses:
    Statistics:
      packets: received 243774, sent 52179
      bytes: received 17888446, sent 3602852
    Storm control drop counters:
      packets: broadcast 0, multicast 0, unknown unicast 0
      bytes: broadcast 0, multicast 0, unknown unicast 0
    Dynamic ARP inspection drop counters:
      packets: 0, bytes: 0
    IP source guard drop counters:
      packets: 0, bytes: 0
List of Access PWs:
List of VFIs:
  VFI customer1-engineering (up)
    VPN-ID: 2, Auto Discovery: BGP, state is Provisioned (Service Connected)
    Route Distinguisher:  (auto) 10.0.0.11:32770
    Import Route Targets:
      0.0.0.1:2
    Export Route Targets:
      0.0.0.1:2
    Signaling protocol: LDP
    AS Number: 65000
    VPLS-ID: 65000:2
    L2VPN Router ID: 10.0.0.11
    PW: neighbor 10.0.0.12, PW ID 65000:2, state is up ( established )
      PW class not set, XC ID 0xc0000005
      Encapsulation MPLS, Auto-discovered (BGP), protocol LDP
      Source address 10.0.0.11
      PW type Ethernet, control word disabled, interworking none
      PW backup disable delay 0 sec
      Sequencing not set
```

```
  PW Status TLV in use
    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------
    Label        16027                          16042
    BGP Peer ID  10.0.0.11                      10.0.0.12
    LDP ID       10.0.0.11                      10.0.0.12
    AII          10.0.0.11                      10.0.0.12
    AGI          65000:2                        65000:2
    Group ID     0x5                            0x1
    Interface    customer1-engineering          customer1-engineering
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 0
  Create time: 30/05/2013 17:10:18 (00:06:33 ago)
  Last time status changed: 30/05/2013 17:10:24 (00:06:26 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 190, sent 41
    bytes: received 12160, sent 3690
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.13, PW ID 65000:2, state is up ( established )
  PW class not set, XC ID 0xc0000006
  Encapsulation MPLS, Auto-discovered (BGP), protocol LDP
  Source address 10.0.0.11
  PW type Ethernet, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

  PW Status TLV in use
    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------
    Label        16043                          16021
    BGP Peer ID  10.0.0.11                      10.0.0.13
    LDP ID       10.0.0.11                      10.0.0.13
    AII          10.0.0.11                      10.0.0.13
    AGI          65000:2                        65000:2
    Group ID     0x5                            0x3
    Interface    customer1-engineering          customer1-engineering
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 0
```

```
  Create time: 30/05/2013 17:10:18 (00:06:33 ago)
  Last time status changed: 30/05/2013 17:10:27 (00:06:23 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 0, sent 40
    bytes: received 0, sent 3600
DHCPv4 snooping: disabled
IGMP Snooping profile: none
PW: neighbor 10.0.0.14, PW ID 65000:2, state is up ( established )
  PW class not set, XC ID 0xc000000a
  Encapsulation MPLS, Auto-discovered (BGP), protocol LDP
  Source address 10.0.0.11
  PW type Ethernet, control word disabled, interworking none
  PW backup disable delay 0 sec
  Sequencing not set

  PW Status TLV in use
    MPLS         Local                          Remote
    ------------ ------------------------------ ------------------
    Label        16050                          289974
    BGP Peer ID  10.0.0.11                      10.0.0.14
    LDP ID       10.0.0.11                      10.0.0.14
    AII          10.0.0.11                      10.0.0.14
    AGI          65000:2                        65000:2
    Group ID     0x5                            0x6
    Interface    customer1-engineering          customer1-engineering
    MTU          1500                           1500
    Control word disabled                       disabled
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x2                            0x2
                 (LSP ping verification)        (LSP ping verification)
    VCCV CC type 0x6                            0x6
                 (router alert label)           (router alert label)
                 (TTL expiry)                   (TTL expiry)
    ------------ ------------------------------ ------------------
  Incoming Status (PW Status TLV):
    Status code: 0x0 (Up) in Notification message
  MIB cpwVcIndex: 3221225482
  Create time: 30/05/2013 17:11:46 (00:05:05 ago)
  Last time status changed: 30/05/2013 17:11:51 (00:05:00 ago)
  MAC withdraw message: send 0 receive 0
  Static MAC addresses:
  Statistics:
    packets: received 0, sent 31
    bytes: received 0, sent 2790
DHCPv4 snooping: disabled
IGMP Snooping profile: none
VFI Statistics:
  drops: illegal VLAN 0, illegal length 0
```

4.4.4 MAC Flushes and Withdrawals

Forwarding in VPLS is based on the mac-address-table, which is dynamically built by learning the source MAC addresses of the frames being received. If there is a topology change in a bridge-domain, a host can become reachable through a different AC or VPLS neighbor. Traffic for that host cannot reach its destination if frames continue to be forwarded according to the existing mac-address-table.

For a L2VPN PE, there are multiple ways to detect a topology change:

- A port in the bridge-domain goes up or down.
- A spanning tree Topology Change Notification (TCN) BPDU is processed when the L2VPN PE runs the full MST implementation or a spanning tree access gateway protocol. The failing link cannot be local on the PE but can be farther away in the topology. The PE intercepts the TCN.

When a L2VPN PE detects a topology change, it takes two actions:

1. The PE flushes the mac-address-table of the bridge-domains impacted by the topology change. When the PE is configured for PVSTAG or Per-VLAN Rapid Spanning Tree Access Gateway (PVRSTAG), a TCN BPDU detected in one VLAN subinterface affects all VLANs and bridge-domains on that physical interface.
2. The PE signals to the VPLS neighbors through an MPLS LDP MAC withdrawal message that they can flush their mac-address-table. All remote L2VPN PEs receiving the MAC withdrawal LDP message flush their mac-address-tables, and traffic is flooded again. The mac-address-tables are rebuilt based on the new topology.

The default behavior of the MAC withdrawal message in case of port flap has changed over time:

- Traditionally in Cisco IOS XR software, a L2VPN PE sent MAC withdrawal messages when an AC was going down. The intent was to have remote PEs flush their MAC address tables for the impacted bridge-domain so that the MAC addresses pointing behind the downed port would be learned from another port.
- However, this created an interoperability problem with some remote PEs that use RFC 4762 and purge MAC addresses that point at all PEs except the one which is sending the MAC withdrawal message. RFC 4762 assumes that a PE would send a MAC withdrawal message when an AC comes up but not when an AC goes down. After Cisco IOS XR Software Release 4.2.1, the default behavior is to send LDP MAC withdrawal messages only when a bridge-domain port comes up in order to better comply with the RFC. A configuration command was added in order to revert to the old behavior.

This is a show command with the default behavior after Cisco IOS XR Software Release 4.2.1:

<#root>

```
RP/0/RSP1/CPU0:router3#sh l2vpn bridge-domain bd-name engineering det |
   i "PW:|VFI|neighbor|MAC w"
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
```

MAC withdraw sent on bridge port down: disabled

```
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of VFIs:
    VFI customer1-engineering (up)
      PW: neighbor 10.0.0.11, PW ID 2, state is up ( established )
        MAC withdraw message: send 0 receive 0
      PW: neighbor 10.0.0.12, PW ID 2, state is up ( established )
```

```
    MAC withdraw message: send 0 receive 4
  PW: neighbor 10.0.0.14, PW ID 2, state is up ( established )
    MAC withdraw message: send 0 receive 2
  VFI Statistics:
```

The important line is the MAC withdraw sent on bridge port down, which is now disabled by default after Cisco IOS XR Software Release 4.2.1. The command also gives the number of MAC withdrawal messages sent and received in the bridge-domain. A high number of withdrawal messages indicates instability in the bridge-domain.

This is the configuration that reverts to the old behavior:

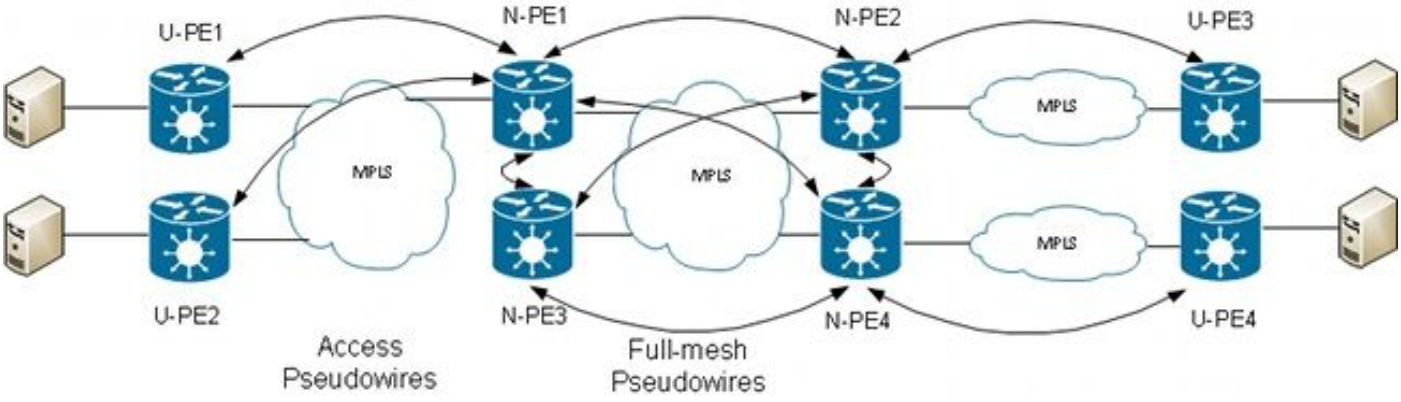
```
l2vpn
 bridge group customer1
  bridge-domain finance
   mac
    withdraw state-down
   !
  !
 !
!
```

4.4.5 H-VPLS

VPLS requires a full mesh of PWs between L2VPN PEs in order to ensure that any PE can reach, in one hop, a host behind any other PE without the need for one PE to reflect frames from one PW to another PW. This is the basis for the split horizon rule, which prevents a PE from forwarding frames from one PW to another PW. Even in special cases, where the destination MAC address in the mac-address-table points at another PW, the frame is dropped.

A full mesh of PWs means that the number of PWs can become very high as the number of PEs grows, so this can introduce scalability issues.

You can decrease the number of PWs in this topology with a hierarchy of PEs:



In this topology:

- A user Provider Edge (U-PE) device has ACs to the CEs.
- The U-PE device transports the CE traffic over an MPLS point-to-point PW to a network Provider Edge (N-PE) device.
- The N-PE is a core VPLS PE that is fully meshed with other N-PEs.
- On the N-PE, the PW coming from the U-PE is considered an access PW much like an AC. The U-PE is not part of the mesh with the other N-PEs, so the N-PE can consider the access PW as an AC and forward traffic from that access PW to the core PWs that are part of the VPLS full mesh.
- The core PWs between N-PEs are configured under a VFI in order to ensure that the split horizon rule is applied to all the core PWs configured under the VFI.
- Access PWs from U-PEs are not configured under a VFI, so they do not belong to the same SHG as the VFI PWs. Traffic can be forwarded from an access PW to a VFI PW and vice versa.
- U-PEs can use the PW redundancy feature in order to have a primary PW to a primary N-PE and have a standby PW to a standby N-PE. The standby takes over when the primary PW goes down.

This is an example where U-PE1 (10.0.0.15) is configured with PW redundancy to N-PE1 (10.0.0.11) and N-PE2 (10.0.0.12):

```
RP/0/RP0/CPU0:U-PE1#sh run int ten 0/1/0/5.2
interface TenGigE0/1/0/5.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!

RP/0/RP0/CPU0:U-PE1#sh run l2vpn xconnect group customer1
l2vpn
 xconnect group customer1
  p2p engineering-0-1-0-5
   interface TenGigE0/1/0/5.2
   neighbor 10.0.0.11 pw-id 15
    backup neighbor 10.0.0.12 pw-id 15
    !
   !
  !
 !
!

RP/0/RP0/CPU0:U-PE1#sh l2vpn xconnect group customer1
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed
```

```
XConnect                    Segment 1                Segment 2
Group      Name       ST   Description     ST       Description             ST
-----------------------   ----------------------   -----------------------------
customer1  engineering-0-1-0-5
                      UP   Te0/1/0/5.2     UP       10.0.0.11       15      UP
                                                    Backup
                                                    10.0.0.12       15      SB
------------------------------------------------------------------------------
```

The PW to 10.0.0.12 is in standby state. On N-PE1, there are an access PW to 10.0.0.15 and an AC that are not under the VFI.

N-PE1 is learning some MAC addresses over the access PW and the VFI PWs:

```
RP/0/RSP0/CPU0:N-PE1#sh run l2vpn bridge group customer1 bridge-domain
   engineering
l2vpn
 bridge group customer1
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   neighbor 10.0.0.15 pw-id 15
   !
   vfi customer1-engineering
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:N-PE1#sh l2vpn bridge-domain bd-name engineering
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 4 (4 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.2, state: up, Static MAC addresses: 0
  List of Access PWs:
      Neighbor 10.0.0.15 pw-id 15, state: up, Static MAC addresses: 0
  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.12 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 2, state: up, Static MAC addresses: 0

RP/0/RSP0/CPU0:N-PE1#sh l2vpn forwarding bridge-domain customer1:engineering
   mac-address location 0/0/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location
```

```
Mac Address    Type    Learned from/Filtered on LC learned Resync Age  Mapped to
--------------------------------------------------------------------------------
6c9c.ed3e.e46d dynamic (10.0.0.15, 15)          0/0/CPU0   0d 0h 0m 0s N/A
0019.552b.b5c3 dynamic (10.0.0.12, 2)           0/0/CPU0   0d 0h 0m 0s N/A
0024.985e.6a42 dynamic (10.0.0.12, 2)           0/0/CPU0   0d 0h 0m 0s N/A
001d.4603.1f42 dynamic (10.0.0.13, 2)           0/0/CPU0   0d 0h 0m 0s N/A
```

On N-PE2 (10.0.0.12), the access PW is in standby state:

<#root>

```
RP/0/RSP0/CPU0:N-PE2#sh run l2vpn bridge group customer1 bridge-domain
   engineering
l2vpn
 bridge group customer1
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   neighbor 10.0.0.15 pw-id 15
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:N-PE2#sh l2vpn bridge-domain bd-name engineering
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 1, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 4 (3 up), PBBs: 0 (0 up)
  List of ACs:
    Gi0/1/0/3.2, state: up, Static MAC addresses: 0
  List of Access PWs:


Neighbor 10.0.0.15 pw-id 15, state: standby, Static MAC addresses: 0


  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.11 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 2, state: up, Static MAC addresses: 0
```

4.4.6 Split Horizon Groups (SHGs)

The split horizon rule dictates that a frame received on one VFI PW cannot be forwarded over another VFI PW. VFI N-PEs can be fully meshed.

This split horizon is enforced through a SHG:

- Members from one SHG cannot forward frames to each other, but can forward frames to members of other SHGs.
- All VFI PWs are assigned to SHG 1 by default. This ensures that there is no forwarding between VFI PWs so that the split horizon rule is enforced. Packets received on a VFI PW can be forwarded to ACs and access PWs because they are not part of the same SHG.
- All ACs and access PWs are not part of a SHG group by default, which means that packets received on an AC or access PW can be forwarded to another AC or access PW in the same bridge-domain.
- ACs and access PWs can be assigned to the SHG 2 with the **split-horizon group** command if the goal is to prevent forwarding between them.

```
RP/0/RSP0/CPU0:N-PE1#sh run l2vpn bridge group customer1 bridge-domain
   engineering
l2vpn
 bridge group customer1
  bridge-domain engineering
   interface GigabitEthernet0/0/0/1.2
    split-horizon group
   !
   interface GigabitEthernet0/1/0/3.2
    split-horizon group
   !
   neighbor 10.0.0.15 pw-id 15
    split-horizon group
   !
   vfi customer1-engineering
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!
```

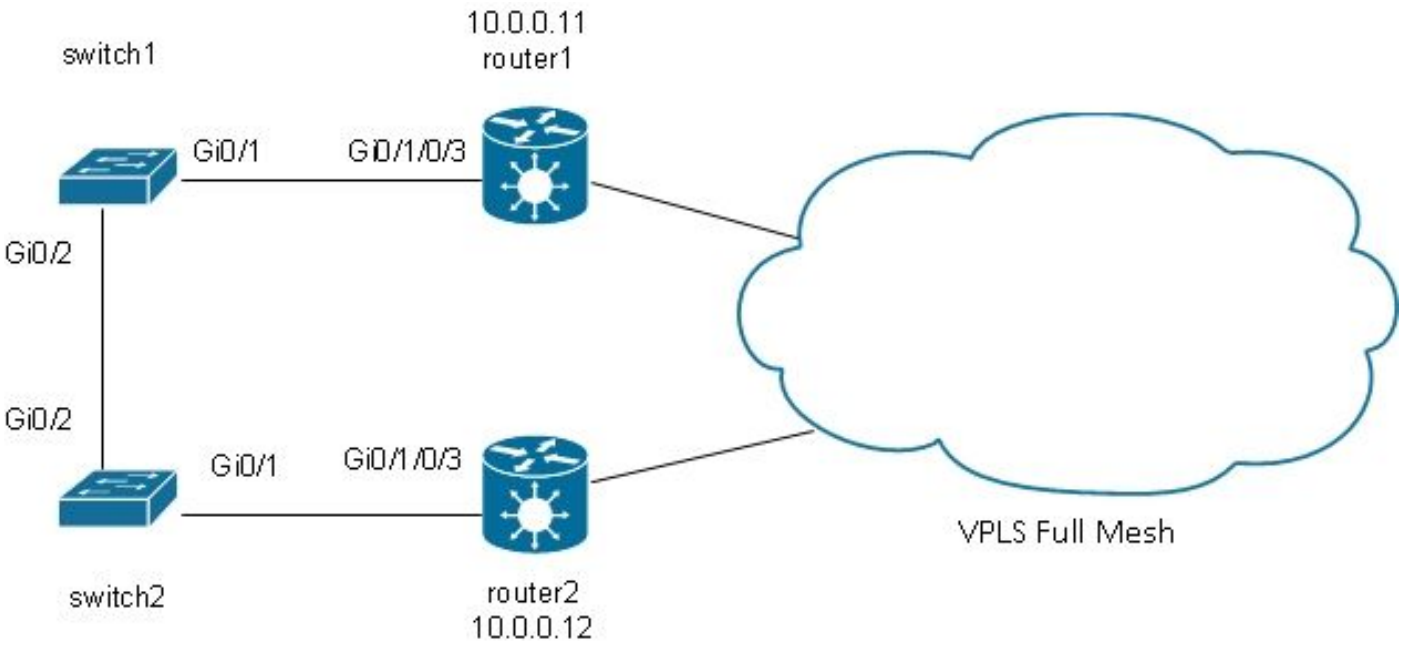
In this configuration, there is no forwarding between Gi 0/0/0/1.2 and Gi 0/1/0/3.2, Gi 0/0/0/1.2 and 10.0.0.15, or Gi 0/1/0/3.2 and 10.0.0.15. But there can still be traffic forwarding between the ACs and the VFI PWs because they are part of different SHGs (1 and 2).

```
RP/0/RSP0/CPU0:N-PE1#sh l2vpn bridge-domain bd-name engineering detail  |
   i "state is|List of|VFI|Split"
  Split Horizon Group: none
```

```
ACs: 2 (1 up), VFIs: 1, PWs: 4 (4 up), PBBs: 0 (0 up)
List of ACs:
  AC: GigabitEthernet0/0/0/1.2, state is unresolved
    Split Horizon Group: enabled
  AC: GigabitEthernet0/1/0/3.2, state is up
    Split Horizon Group: enabled
List of Access PWs:
  PW: neighbor 10.0.0.15, PW ID 15, state is up ( established )
    Split Horizon Group: enabled
List of VFIs:
  VFI customer1-engineering (up)
    PW: neighbor 10.0.0.12, PW ID 2, state is up ( established )
    PW: neighbor 10.0.0.13, PW ID 2, state is up ( established )
    PW: neighbor 10.0.0.14, PW ID 2, state is up ( established )
    VFI Statistics:
```

4.4.7 Redundancy

In an attempt to introduce redundancy, you can have a site that is dual attached to the VPLS domain:



If a host connected to switch1 sends a broadcast, switch1 forwards it to router1 and to switch2. Router1 has a full mesh of PWs, so there is a PW to router2, and router1 forwards the broadcast over that PW. Router2 forwards the broadcast to switch2, which forwards it to switch1. This results in a physical loop.

4.4.7.1 Spanning Tree

The full MST implementation does not work with VPLS because that implementation sends MST BPDUs on a main interface in order to control the forwarding state of all VLANs on that interface. With VPLS, there are VFIs for each bridge-domain, so you cannot send BPDUs on a main interface for all of those VFIs.

Spanning tree BPDUs are transported over VPLS and point-to-point PWs by default.

If switch1 and switch2 are sending per-VLAN BPDUs or untagged MST BPDUs and if the BPDUs match l2transport subinterfaces on router1 and router2, the BPDUs are transported through VPLS. The switches see each other's BPDUs on the Gi 0/1 interfaces, and spanning tree breaks the loop and blocks one port.

Switch2 is the root for VLAN 2:

```
switch2#sh spanning-tree vlan 2

MST0
  Spanning tree enabled protocol mstp
  Root ID    Priority    32768
             Address     0024.985e.6a00
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768  (priority 32768 sys-id-ext 0)
             Address     0024.985e.6a00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Desg FWD 20000     128.1    P2p Bound(PVST)
Gi0/2               Desg FWD 20000     128.2    P2p Bound(PVST)
```

Switch1 has its root port on Gi 0/1 and is blocking Gi 0/2:

```
switch1#sh spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
             Address     0024.985e.6a00
             Cost        4
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     0019.552b.b580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 4         128.1    P2p
Gi0/2               Altn BLK 4         128.2    P2p
```

The problem is that the BPDUs are also transported to remote sites, and spanning tree instability in one site

propagates to all sites connected to the VPLS domain. It is safer to isolate each site and not transport BPDUs over VPLS.

One solution is use of an access gateway version of the STP. This is a limited implementation of the protocol, where the L2VPN PEs are configured to send some static BPDUs in order to appear connected to the spanning tree root. The L2VPN PE does not transport the BPDUs received from the CEs to the remote sites, so each site has its own spanning tree domain.

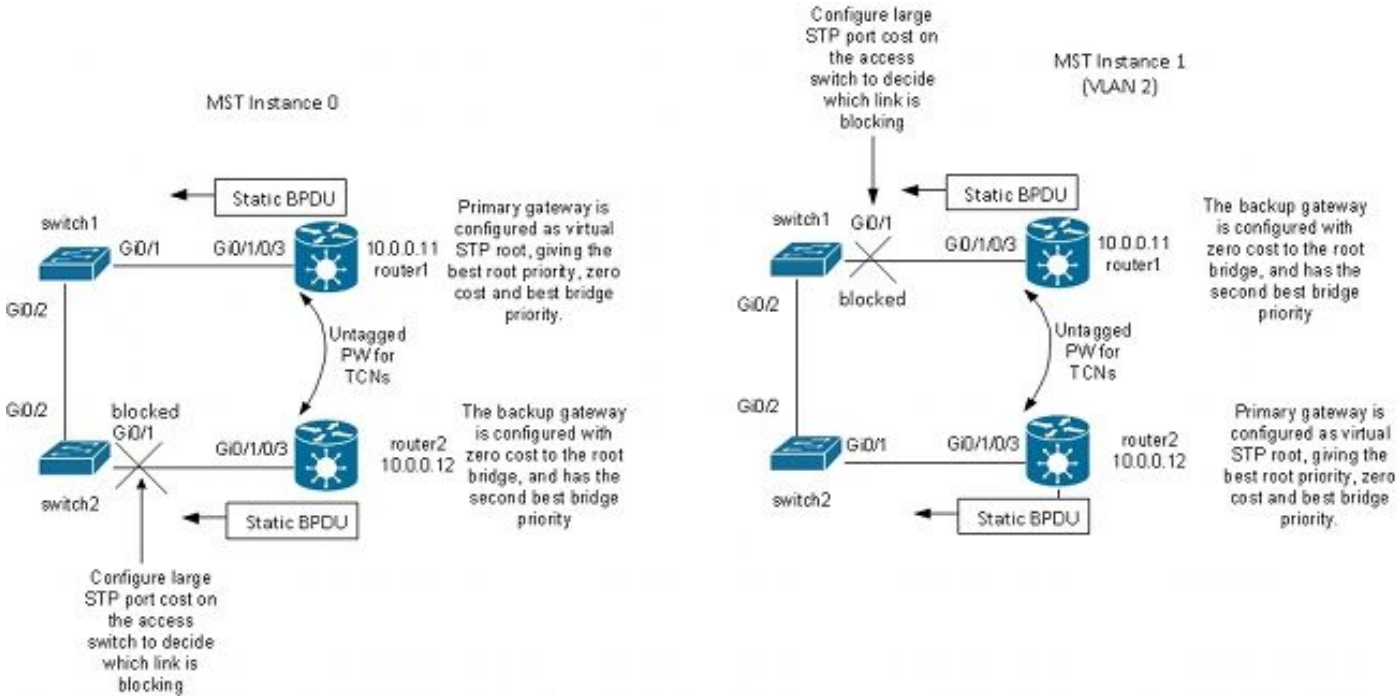
4.4.7.2 MSTAG

As explained in the Spanning Tree section, MST sends untagged BPDUs, but these BPDUs control the forwarding state of all VLANs on the interface.

VLANs can be grouped into multiple instances, and each instance has its own forwarding state.

VLANs are usually grouped so that traffic can be spread evenly between multiple paths. When there are two paths, half of the traffic belongs to an instance that is forwarding on the first path and blocking on the second path. The other half of the traffic belongs to an instance that is blocking on the first path and forwarding on the second path. This allows for loadbalancing between the two paths under stable conditions. Otherwise, you have one path that is ordinarily completely blocked and becomes active only when the primary path is down.

Here is a typical MSTAG topology:



In this lab example, instance 1 has VLAN 2, and instance 0 has the other VLANs. (In a more realistic scenario, VLANs are spread between multiple instance in order to achieve good traffic loadbalancing between the instances.) Because some VLANs have much more traffic than others, there are not always the same number of VLANs in each instance.

This is the configuration for MST instance 0:

- Router1 and router2 are sending some static BPDUs based on the MSTAG configuration. They are not processing the incoming BPDUs from the network or trying to run a full implementation. With MSTAG, the two L2VPN PEs just send static BPDUs based on their MSTAG configuration.
- Router1 is configured in order to attract traffic of instance 0 by appearing to be the root for that instance.
- Router2 is configured with the second-best root priority for instance 0, so that it becomes the new root in case of router1 failure or AC failure between switch1 and router1.
- Switch2 is configured with a high spanning tree cost on the port Gi 0/1 to router2 in order to ensure that its primary path to the root is on Gig 0/2 through switch1 and router1.
- Switch2 selects Gi 0/2 as root port for instance0 and selects Gi 0/1 as an alternate port in case the root is lost.
- Thus, traffic from that site in the VLANs belonging to instance 0 reaches other sites over VPLS through router1.

For MST instance 1 (VLAN 2), the configuration is reversed:

- Router2 is configured in order to attract traffic of instance 1 by appearing to be the root for that instance.
- Router1 is configured with the second-best root priority for instance 1, so that it becomes the new root in case of router2 failure or AC failure between switch2 and router2.
- Switch1 is configured with an high spanning tree cost on the port Gi 0/1 to router1 in order to ensure that its primary path to the root is on Gig 0/2 through switch2 and router2.
- Switch1 selects Gi 0/2 as root port for instance 1 and selects Gi 0/1 as an alternate port in case the root is lost.
- Thus, traffic from that site in the VLANs belonging to instance 1 (VLAN 2 in this example) reaches other sites over VPLS through router2.
- There must be a subinterface on router1 and router2 in order to catch the untagged TCNs and forward them through a point-to-point PW to the other router. Because switch1 and switch2 could lose their direct links and become isolated from each other, router1 and router2 must forward the TCNs between them through that point-to-point PW.
- The PEs also intercept the TCNs, flush their mac-address-tables, and send LDP MAC withdrawal to remote PEs.

This is the configuration on router1:

```
RP/0/RSP0/CPU0:router1#sh run int gigabitEthernet 0/1/0/3.*
interface GigabitEthernet0/1/0/3.1 l2transport
 encapsulation untagged
!
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
 ethernet-services access-group filter-stp egress
!
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
 ethernet-services access-group filter-stp egress
```

```
!

RP/0/RSP0/CPU0:router1#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/1/0/3.3
   !
   vfi customer1-finance
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.13 pw-id 3
    !
    neighbor 10.0.0.14 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:router1#sh run l2vpn xconnect group customer1
l2vpn
 xconnect group customer1
  p2p mstag-gi-0-1-0-3
   interface GigabitEthernet0/1/0/3.1
   neighbor 10.0.0.13 pw-id 103
   !
  !
 !
!

RP/0/RSP0/CPU0:router1#sh run spanning-tree mstag customer1-0-1-0-3
spanning-tree mstag customer1-0-1-0-3
 interface GigabitEthernet0/1/0/3.1
  name customer1
  revision 1
  bridge-id 0000.0000.0001
  instance 0
   root-id 0000.0000.0001
   priority 4096
   root-priority 4096
  !
  instance 1
   vlan-ids 2
   root-id 0000.0000.0002
   priority 8192
   root-priority 4096
  !
 !
!
```

```
RP/0/RSP0/CPU0:router1#sh spanning-tree mstag customer1-0-1-0-3
GigabitEthernet0/1/0/3.1
  Pre-empt delay is disabled
  Name:            customer1
  Revision:        1
  Max Age:         20
  Provider Bridge: no
  Bridge ID:       0000.0000.0001
  Port ID:         1
  External Cost:   0
  Hello Time:      2
  Active:          yes
  BPDUs sent:      3048
  MSTI 0 (CIST):
    VLAN IDs:         1,3-4094
    Role:             Designated
    Bridge Priority:  4096
    Port Priority:    128
    Cost:             0
    Root Bridge:      0000.0000.0001
    Root Priority:    4096
    Topology Changes: 369
  MSTI 1
    VLAN IDs:         2
    Role:             Designated
    Bridge Priority:  8192
    Port Priority:    128
    Cost:             0
    Root Bridge:      0000.0000.0002
    Root Priority:    4096
    Topology Changes: 322
```

In this configuration, note that:

- In MST instance 0, the root bridge is 0000.0000.0001, which is the bridge ID of router1.
- In MST instance 1, the root bridge is 0000.0000.0002, which is the bridge ID of router2.
- The bridge priority of router1 is 4096 in instance 0 (to become the root) and 8192 in instance 1 (to become the second-best root).
- The bridge priority of router1 is 8192 in instance 0 (to become the second-best root) and 4096 in instance 1 (to become the root).
- The point-to-point cross connect on GigabitEthernet0/1/0/3.1 carries the untagged MST TCNs to the other router.

An egress ACL has been configured on the dot1q subinterfaces in order to drop per-VLAN BPDUs that can be sent by another site that has not been migrated to MST yet. This configuration prevents the CE switch from declaring that the interface as inconsistent when it receives a per-VLAN BPDU on an interface configured for MST.

The configuration on router2 is very similar:

```
RP/0/RSP0/CPU0:router2#sh run int gig 0/1/0/3.*
interface GigabitEthernet0/1/0/3.1 l2transport
```

```
 encapsulation untagged
!
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
 ethernet-services access-group filter-stp egress
!
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
 ethernet-services access-group filter-stp egress
!

RP/0/RSP0/CPU0:router2#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/1/0/3.3
   !
   vfi customer1-finance
    neighbor 10.0.0.11 pw-id 3
    !
    neighbor 10.0.0.13 pw-id 3
    !
    neighbor 10.0.0.14 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:router2#sh run l2vpn xconnect group customer1
l2vpn
 xconnect group customer1
  p2p mstag-gi-0-1-0-3
   interface GigabitEthernet0/1/0/3.1
   neighbor 10.0.0.13 pw-id 103
   !
  !
 !
!

RP/0/RSP0/CPU0:router2#sh run spanning-tree mstag customer1-0-1-0-3
spanning-tree mstag customer1-0-1-0-3
 interface GigabitEthernet0/1/0/3.1
  name customer1
  revision 1
  bridge-id 0000.0000.0002
  instance 0
   root-id 0000.0000.0001
```

```
   priority 8192
   root-priority 4096
  !
  instance 1
   vlan-ids 2
   root-id 0000.0000.0002
   priority 4096
   root-priority 4096
  !
 !
!

RP/0/RSP0/CPU0:router2#sh spanning-tree mstag customer1-0-1-0-3
GigabitEthernet0/1/0/3.1
  Pre-empt delay is disabled
  Name:            customer1
  Revision:        1
  Max Age:         20
  Provider Bridge: no
  Bridge ID:       0000.0000.0002
  Port ID:         1
  External Cost:   0
  Hello Time:      2
  Active:          yes
  BPDUs sent:      3186
  MSTI 0 (CIST):
    VLAN IDs:         1,3-4094
    Role:             Designated
    Bridge Priority:  8192
    Port Priority:    128
    Cost:             0
    Root Bridge:      0000.0000.0001
    Root Priority:    4096
    Topology Changes: 365
  MSTI 1
    VLAN IDs:         2
    Role:             Designated
    Bridge Priority:  4096
    Port Priority:    128
    Cost:             0
    Root Bridge:      0000.0000.0002
    Root Priority:    4096
    Topology Changes: 177
```

This is the basic configuration on switch 1:

```
switch1#sh run | b spanning-tree
spanning-tree mode mst
spanning-tree extend system-id
!
spanning-tree mst configuration
 name customer1
 revision 1
 instance 1 vlan 2
!

switch1#sh run int gig 0/1 | i spanning
```

```
 spanning-tree mst 1 cost 100000

switch1#sh spanning-tree

MST0
  Spanning tree enabled protocol mstp
  Root ID    Priority    4096
             Address     0000.0000.0001
             Cost        0
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768  (priority 32768 sys-id-ext 0)
             Address     0019.552b.b580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 20000     128.1    P2p
Gi0/2               Desg FWD 20000     128.2    P2p



MST1
  Spanning tree enabled protocol mstp
  Root ID    Priority    4097
             Address     0000.0000.0002
             Cost        40000
             Port        2 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0019.552b.b580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Altn BLK 100000    128.1    P2p
Gi0/2               Root FWD 20000     128.2    P2p
```

Thus, traffic in instance 0 is forwarded through router1 and the traffic in instance 1 is forwarded through switch2 and router2.

The configuration on switch2 uses the same commands as switch1:

```
switch2#sh run | b spanning
spanning-tree mode mst
spanning-tree extend system-id
!
spanning-tree mst configuration
 name customer1
 revision 1
 instance 1 vlan 2
!

switch2#sh run int gig 0/1 | i spanning
```

```
 spanning-tree mst 0 cost 100000

switch2#sh spanning-tree

MST0
  Spanning tree enabled protocol mstp
  Root ID    Priority    4096
             Address     0000.0000.0001
             Cost        0
             Port        2 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768  (priority 32768 sys-id-ext 0)
             Address     0024.985e.6a00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Altn BLK 100000    128.1    P2p
Gi0/2               Root FWD 20000     128.2    P2p



MST1
  Spanning tree enabled protocol mstp
  Root ID    Priority    4097
             Address     0000.0000.0002
             Cost        20000
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0024.985e.6a00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 20000     128.1    P2p
Gi0/2               Desg FWD 20000     128.2    P2p
```

Switch2 goes through switch1 and router1 for instance0 and through router2 for instance1.

The traffic is loadbalanced because one instance exits the site through router1 and the other instance exits the site through router2.

If the link between router1 and switch1 is down, both instances go through router2.

```
switch1#sh spanning-tree

MST0
  Spanning tree enabled protocol mstp
  Root ID    Priority    4096
             Address     0000.0000.0001
             Cost        0
```

```
             Port        2 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768  (priority 32768 sys-id-ext 0)
             Address     0019.552b.b580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/2               Root FWD 20000     128.2    P2p



MST1
  Spanning tree enabled protocol mstp
  Root ID    Priority    4097
             Address     0000.0000.0002
             Cost        40000
             Port        2 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0019.552b.b580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/2               Root FWD 20000     128.2    P2p


switch2#sh spanning-tree

MST0
  Spanning tree enabled protocol mstp
  Root ID    Priority    4096
             Address     0000.0000.0001
             Cost        0
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768  (priority 32768 sys-id-ext 0)
             Address     0024.985e.6a00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 100000    128.1    P2p
Gi0/2               Desg FWD 20000     128.2    P2p



MST1
  Spanning tree enabled protocol mstp
  Root ID    Priority    4097
             Address     0000.0000.0002
             Cost        20000
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0024.985e.6a00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

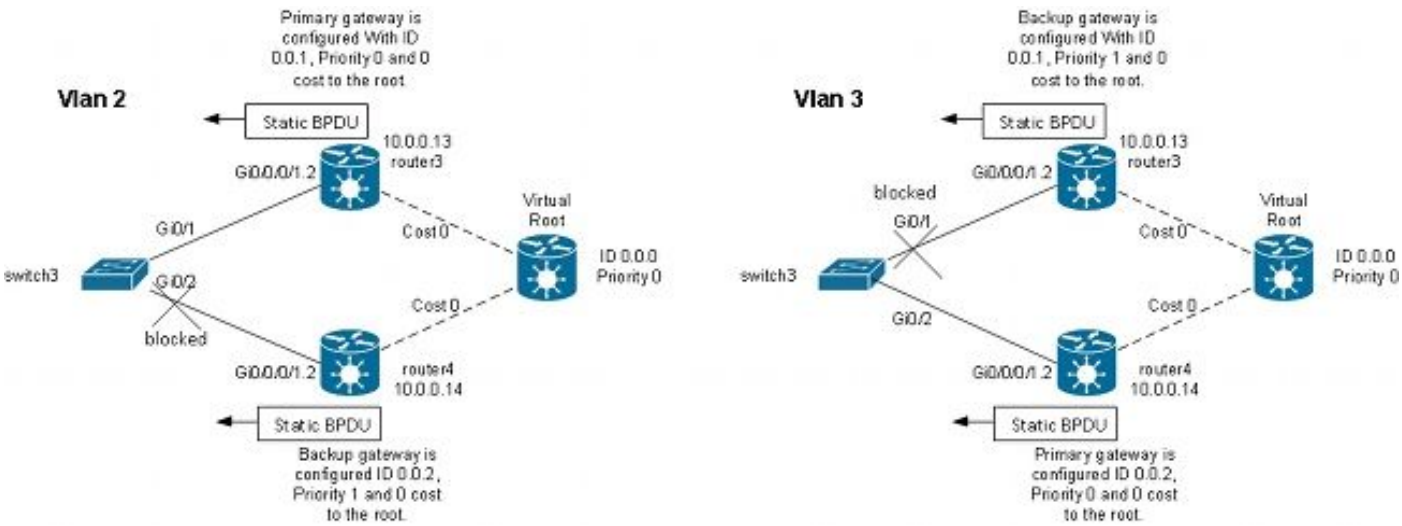
```
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 20000     128.1    P2p
Gi0/2               Desg FWD 20000     128.2    P2p
```

Rapid convergence can be achieved in this type of failure because the path through the second-best root was already selected as the alternate path. With MSTAG, MST BPDUs are not transported over VPLS so sites are isolated from the instability in other sites.

4.4.7.3 PVSTAG or PVRSTAG

MSTAG is the preferred access gateway protocol for VPLS because it uses the rapid spanning tree and because it is scalable with its use of instances rather than BPDUs on each VLAN.

If a site cannot be migrated to MST and the only solution is to keep running PVST+ or PVRST, you can use PVSTAG or PVRSTAG, but the implementation is limited to one specific topology:



In this topology, the most important restriction is that there can be only one CE switch. You cannot have two switches as in the MSTAG topology. In MSTAG, you can configure a point-to-point PW in order to transport the untagged traffic (including the BPDU TCNs) from one PE to the other when the site is split into two parts. With PVST and PVRST, the TCNs are sent tagged so they match the same subinterface as the data traffic to be transported over VPLS. The router would have to identify the BPDUs based on the MAC address and protocol type in order to forward the TCNs to the other side. Because this is not currently supported, there is a requirement to have only one CE device.

Another requirement in releases earlier than Cisco IOS XR Software Release 4.3.0 is that bundle interfaces cannot be used as ACs. This restriction has been lifted in Cisco IOS XR Software Release 4.3.0.

The principle is very much the same as with MSTAG. The PVSTAG router sends static BPDUs so that the CE appears to be connected to switches that are directly connected to the (virtual) root with a cost 0. In order to loadbalance the traffic, some VLANs can be configured with the root on router3 and others with the root

on router4.

This is a configuration example on router3:

```
RP/0/RSP1/CPU0:router3#sh run int gigabitEthernet 0/0/0/1.*
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface GigabitEthernet0/0/0/1.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!

RP/0/RSP1/CPU0:router3#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/0/0/1.3
   !
   vfi customer1-finance
    neighbor 10.0.0.11 pw-id 3
    !
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.14 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/0/0/1.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP1/CPU0:router3#sh run spanning-tree pvstag customer1-0-0-0-1
spanning-tree pvstag customer1-0-0-0-1
 interface GigabitEthernet0/0/0/1
  vlan 2
   root-priority 0
   root-id 0000.0000.0000
   root-cost 0
   priority 0
   bridge-id 0000.0000.0001
  !
  vlan 3
   root-priority 0
   root-id 0000.0000.0000
   root-cost 0
```

```
   priority 1
   bridge-id 0000.0000.0001
  !
 !
!

RP/0/RSP1/CPU0:router3#sh spanning-tree pvstag customer1-0-0-0-1
GigabitEthernet0/0/0/1
  VLAN 2
    Pre-empt delay is disabled
    Sub-interface:    GigabitEthernet0/0/0/1.2 (Up)
    Max Age:          20
    Root Priority:    0
    Root Bridge:      0000.0000.0000
    Cost:             0
    Bridge Priority:  0
    Bridge ID:        0000.0000.0001
    Port Priority:    128
    Port ID           1
    Hello Time:       2
    Active:           Yes
    BPDUs sent:       202821
    Topology Changes: 0
  VLAN 3
    Pre-empt delay is disabled
    Sub-interface:    GigabitEthernet0/0/0/1.3 (Up)
    Max Age:          20
    Root Priority:    0
    Root Bridge:      0000.0000.0000
    Cost:             0
    Bridge Priority:  1
    Bridge ID:        0000.0000.0001
    Port Priority:    128
    Port ID           1
    Hello Time:       2
    Active:           Yes
    BPDUs sent:       202821
    Topology Changes: 0
```

This is a configuration example on router4:

```
RP/0/RSP1/CPU0:router4#sh run int gig 0/0/0/1.*
interface GigabitEthernet0/0/0/1.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface GigabitEthernet0/0/0/1.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!

RP/0/RSP1/CPU0:router4#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface GigabitEthernet0/0/0/1.3
   !
```

```
   vfi customer1-finance
    neighbor 10.0.0.11 pw-id 3
    !
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.13 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface GigabitEthernet0/0/0/1.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP1/CPU0:router4#sh run spanning-tree pvstag customer1-0-0-0-1
spanning-tree pvstag customer1-0-0-0-1
 interface GigabitEthernet0/0/0/1
  vlan 2
   root-priority 0
   root-id 0000.0000.0000
   root-cost 0
   priority 1
   bridge-id 0000.0000.0002
  !
  vlan 3
   root-priority 0
   root-id 0000.0000.0000
   root-cost 0
   priority 0
   bridge-id 0000.0000.0002
  !
 !
!

RP/0/RSP1/CPU0:router4#sh spanning-tree pvstag customer1-0-0-0-1
GigabitEthernet0/0/0/1
  VLAN 2
    Pre-empt delay is disabled
    Sub-interface:    GigabitEthernet0/0/0/1.2 (Up)
    Max Age:          20
    Root Priority:    0
    Root Bridge:      0000.0000.0000
    Cost:             0
    Bridge Priority:  1
    Bridge ID:        0000.0000.0002
    Port Priority:    128
    Port ID           1
    Hello Time:       2
    Active:           Yes
    BPDUs sent:       202799
    Topology Changes: 0
  VLAN 3
```

```
    Pre-empt delay is disabled
    Sub-interface:    GigabitEthernet0/0/0/1.3 (Up)
    Max Age:          20
    Root Priority:    0
    Root Bridge:      0000.0000.0000
    Cost:             0
    Bridge Priority:  0
    Bridge ID:        0000.0000.0002
    Port Priority:    128
    Port ID           1
    Hello Time:       2
    Active:           Yes
    BPDUs sent:       202799
    Topology Changes: 0
```

This is a configuration example on the CE switch3:

```
switch3#sh spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    0
             Address     0000.0000.0000
             Cost        4
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     001d.4603.1f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Root FWD 4         128.1    P2p
Gi0/2               Altn BLK 4         128.2    P2p

switch3#sh spanning-tree vlan 3

VLAN0003
  Spanning tree enabled protocol ieee
  Root ID    Priority    0
             Address     0000.0000.0000
             Cost        4
             Port        2 (GigabitEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32771  (priority 32768 sys-id-ext 3)
             Address     001d.4603.1f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/1               Altn BLK 4         128.1    P2p
Gi0/2               Root FWD 4         128.2    P2p
```

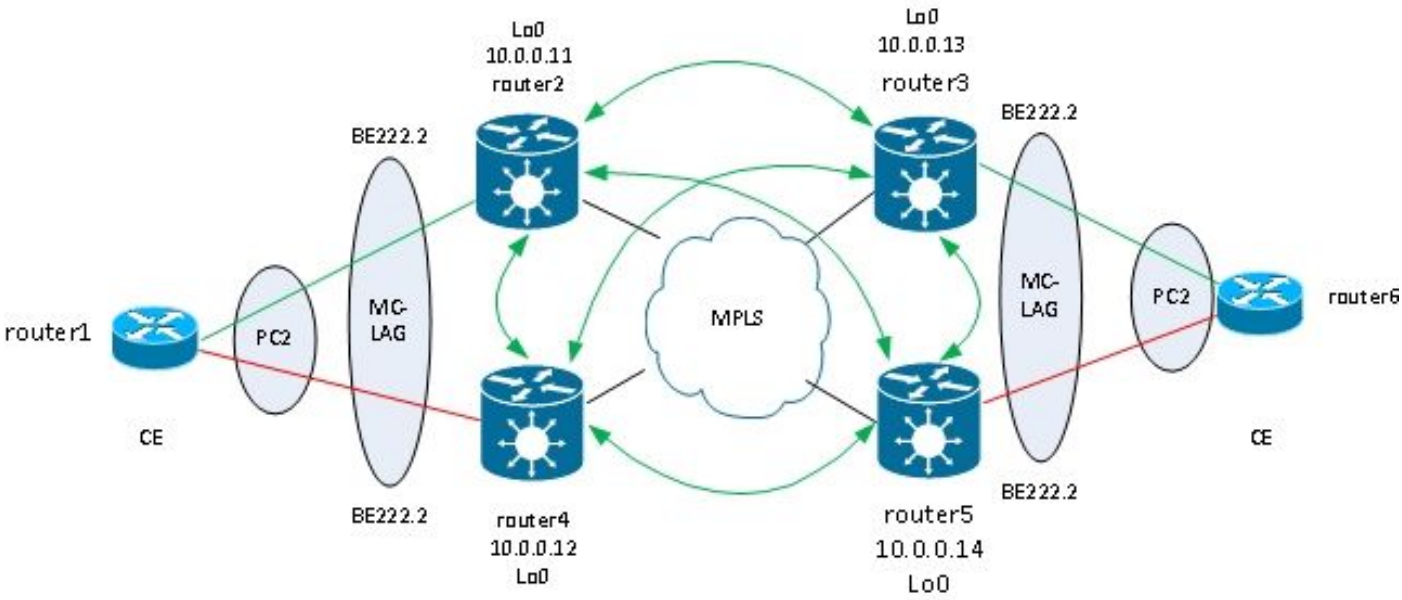
The configuration for PVSTAG is very similar to MSTAG except that the root-priority and the priority of the primary gateway are configured as 4096 and the priority of the backup gateway is configured as 8192 in the MSTAG example.

All other switches in the domains can have priorities higher than the ones configured in PVSTAG or PVRSTAG.

You can tune the interface cost on the CE switches in order to influence which port becomes the root port and which port is blocked.

4.4.7.4 MC-LAG

The MC-LAG configuration with VPLS is simpler than point-to-point PWs with two-way PW redundancy. Instead of one primary PW and three standby PWs, the PEs only need a full mesh of VPLS PWs, which is standard with VPLS:



In this topology, note that:

- MC-LAG runs between the two VPLS PEs on the left: router2 and router4.
- Under normal conditions, the bundle members are active between router1 and router2 and in standby state between router1 and router4.
- Router2 has the bundle subinterfaces configured under VPLS bridge-domains, so router2 forwards the traffic to remote VPLS PEs. There are two sites illustrated in the topology diagram but there could be many more.
- The remote PEs learn the MAC addresses from router1 and devices behind through router2, so the PEs forward traffic for these destination MAC addresses through router2.
- When the link between router1 and router2 goes down or when router2 goes down, the bundle member between router1 and router4 goes active.
- Like router 2, router4 has its bundle subinterfaces configured under VPLS bridge-domains.
- When the bundle subinterfaces come up on router4, router4 sends LDP MAC withdrawal messages to the remote VPLS PEs in order to let them know that there is a topology change.

This is the configuration on router3:

```
RP/0/RSP1/CPU0:router3#sh run redundancy
redundancy
 iccp
  group 2
   mlacp node 1
   mlacp system mac 0200.0000.0002
   mlacp system priority 1
   mlacp connect timeout 0
   member
    neighbor 10.0.0.14
   !
   backbone
    interface TenGigE0/0/0/0
    interface TenGigE0/0/0/1
   !
   isolation recovery-delay 300
  !
 !
!

RP/0/RSP1/CPU0:router3#sh run int bundle-ether 222
interface Bundle-Ether222
 lacp switchover suppress-flaps 100
 mlacp iccp-group 2
 mlacp switchover type revertive
 mlacp switchover recovery-delay 40
 mlacp port-priority 1
 mac-address 0.0.2
 bundle wait-while 0
 bundle maximum-active links 1
 load-interval 30
!

RP/0/RSP1/CPU0:router3#sh run int bundle-ether 222.*
interface Bundle-Ether222.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!
interface Bundle-Ether222.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!

RP/0/RSP1/CPU0:router3#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface Bundle-Ether222.3
   !
   vfi customer1-finance
    neighbor 10.0.0.11 pw-id 3
    !
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.14 pw-id 3
    !
   !
```

```
  !
  bridge-domain engineering
   interface Bundle-Ether222.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!
```

Once the MC-LAG bundle is configured, add it under the VPLS configuration like any other AC.

This is the corresponding configuration on router5:

```
RP/0/RSP1/CPU0:router5#sh run redundancy
redundancy
 iccp
  group 2
   mlacp node 2
   mlacp system mac 0200.0000.0002
   mlacp system priority 1
   mlacp connect timeout 0
   member
    neighbor 10.0.0.13
   !
   backbone
    interface TenGigE0/1/0/0
    interface TenGigE0/1/0/1
   !
   isolation recovery-delay 300
  !
 !
!

RP/0/RSP1/CPU0:router5#sh run int bundle-ether 222
interface Bundle-Ether222
 lacp switchover suppress-flaps 100
 mlacp iccp-group 2
 mlacp switchover type revertive
 mlacp switchover recovery-delay 40
 mac-address 0.0.2
 bundle wait-while 0
 bundle maximum-active links 1
 load-interval 30
!

RP/0/RSP1/CPU0:router5#sh run int bundle-ether 222.*
interface Bundle-Ether222.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
```

```
!
interface Bundle-Ether222.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!

RP/0/RSP1/CPU0:router5#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface Bundle-Ether222.3
   !
   vfi customer1-finance
    neighbor 10.0.0.11 pw-id 3
    !
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.13 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface Bundle-Ether222.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
   !
  !
 !
!
```

Under normal circumstances, the bundle member between router3 and router6 is active, and the member between router5 and router6 is in standby state:

```
RP/0/RSP1/CPU0:router3#sh bundle bundle-ether 222

Bundle-Ether222
  Status:                                  Up
  Local links :   1 / 0 / 1
  Local bandwidth :     1000000 (1000000) kbps
  MAC address (source):                    0000.0000.0002 (Configured)
  Inter-chassis link:                      No
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    1
  Wait while timer:                        Off
  Load balancing:                          Default
  LACP:                                    Operational
    Flap suppression timer:                100 ms
    Cisco extensions:                      Disabled
  mLACP:                                   Operational
    ICCP Group:                            2
    Role:                                  Active
```

```
    Foreign links :       0 / 1
    Switchover type:                        Revertive
    Recovery delay:                         40 s
    Maximize threshold:                     1 link
  IPv4 BFD:                                 Not configured

  Port                  Device        State        Port ID         B/W, kbps
  --------------------  ------------  -----------  --------------  ----------
  Gi0/0/0/1             Local         Active       0x0001, 0x9001     1000000
      Link is Active
  Gi0/0/0/1             10.0.0.14     Standby      0x8000, 0xa002     1000000
      Link is marked as Standby by mLACP peer
RP/0/RSP1/CPU0:router3#

router6#sh etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
2      Po2(SU)         LACP      Gi0/1(P)    Gi0/2(w)

router6#
```

Traffic from the CE is received on router3 and forwarded to remote PEs:

<#root>

```
RP/0/RSP1/CPU0:router3#sh l2vpn bridge-domain group customer1
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: finance, id: 4, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    BE222.3, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-finance (up)
      Neighbor 10.0.0.11 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.12 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 3, state: up, Static MAC addresses: 0
Bridge group: customer1, bridge-domain: engineering, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
```

```
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 3 (3 up), PBBs: 0 (0 up)
  List of ACs:
    BE222.2, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.11 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.12 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 2, state: up, Static MAC addresses: 0

RP/0/RSP1/CPU0:
```

```
router3#sh l2vpn forwarding bridge-domain customer1:
   engineering mac location 0/0/CPU0


 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age  Mapped to
--------------------------------------------------------------------------------
001d.4603.1f01 dynamic BE222.2                  0/0/CPU0   0d 0h 0m 0s N/A
001d.4603.1f42 dynamic BE222.2                  0/0/CPU0   0d 0h 0m 0s N/A
6c9c.ed3e.e46d dynamic (10.0.0.11, 2)           0/0/CPU0   0d 0h 0m 0s N/A
0019.552b.b5c3 dynamic (10.0.0.12, 2)           0/0/CPU0   0d 0h 0m 0s N/A
```

The last command illustrates that router3 is learning some MAC addresses on its bundle and the active members are on router3. On router5, there is no MAC address learned over the bundle as the local member is in standby state:

```
RP/0/RSP1/CPU0:router5#sh l2vpn forwarding bridge-domain customer1:engineering
   mac location 0/0/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age  Mapped to
--------------------------------------------------------------------------------
6c9c.ed3e.e46d dynamic (10.0.0.11, 2)           0/0/CPU0   0d 0h 0m 0s N/A
0019.552b.b5c3 dynamic (10.0.0.12, 2)           0/0/CPU0   0d 0h 0m 0s N/A
001d.4603.1f01 dynamic (10.0.0.13, 2)           0/0/CPU0   0d 0h 0m 0s N/A
```

When the bundle member between router3 and router6 goes down, the bundle member becomes active on router5. The MC-LAG VPLS PEs send an LDP MAC withdrawal message so that remote PEs purge their mac-address-tables and learn the MAC address through the new active MC-LAG PE router5.

Router2 receives a MAC withdrawal messages from router3 and router5 when the active MC-LAG bundle member moves from router3 to router5:

```
RP/0/RSP0/CPU0:router2#sh l2vpn bridge-domain group customer1 detail |
   i "state is|withd|bridge-domain"
```

```
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
    AC: GigabitEthernet0/1/0/3.3, state is up
      PW: neighbor 10.0.0.12, PW ID 3, state is up ( established )
        MAC withdraw message: send 0 receive 0
      PW: neighbor 10.0.0.13, PW ID 3, state is up ( established )
        MAC withdraw message: send 0 receive 1
      PW: neighbor 10.0.0.14, PW ID 3, state is up ( established )
        MAC withdraw message: send 0 receive 1
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
    AC: GigabitEthernet0/0/0/1.2, state is unresolved
    AC: GigabitEthernet0/1/0/3.2, state is up
    PW: neighbor 10.0.0.15, PW ID 15, state is up ( established )
      MAC withdraw message: send 2 receive 0
      PW: neighbor 10.0.0.12, PW ID 2, state is up ( established )
        MAC withdraw message: send 0 receive 0
      PW: neighbor 10.0.0.13, PW ID 2, state is up ( established )
        MAC withdraw message: send 0 receive 1
      PW: neighbor 10.0.0.14, PW ID 2, state is up ( established )
        MAC withdraw message: send 0 receive 1
```

The MAC addresses on router2 move from router3 (10.0.0.13) to router5 (10.0.0.14):

```
RP/0/RSP0/CPU0:router2#sh l2vpn forwarding bridge-domain customer1:
   engineering mac-address location 0/0/CPU0
 To Resynchronize MAC table from the Network Processors, use the command...
    l2vpn resynchronize forwarding mac-address-table location

Mac Address    Type    Learned from/Filtered on LC learned Resync Age  Mapped to
--------------------------------------------------------------------------------
6c9c.ed3e.e46d dynamic (10.0.0.15, 15)          0/0/CPU0   0d 0h 0m 0s N/A
0019.552b.b5c3 dynamic (10.0.0.12, 2)           0/0/CPU0   0d 0h 0m 0s N/A
001d.4603.1f02 dynamic (10.0.0.14, 2)           0/0/CPU0   0d 0h 0m 0s N/A
001d.4603.1f42 dynamic (10.0.0.14, 2)           0/0/CPU0   0d 0h 0m 0s N/A
```

With MC-LAG, a site can use a single bundle to be attached to the other sites through VPLS. MC-LAG provides the link and PE redundancy, but logically it is still one bundle interface to reach other sites. Spanning tree is not required on that bundle, and a BPDU filter could be configured on the CE in order to ensure that BPDUs are not exchanged between sites over VPLS.

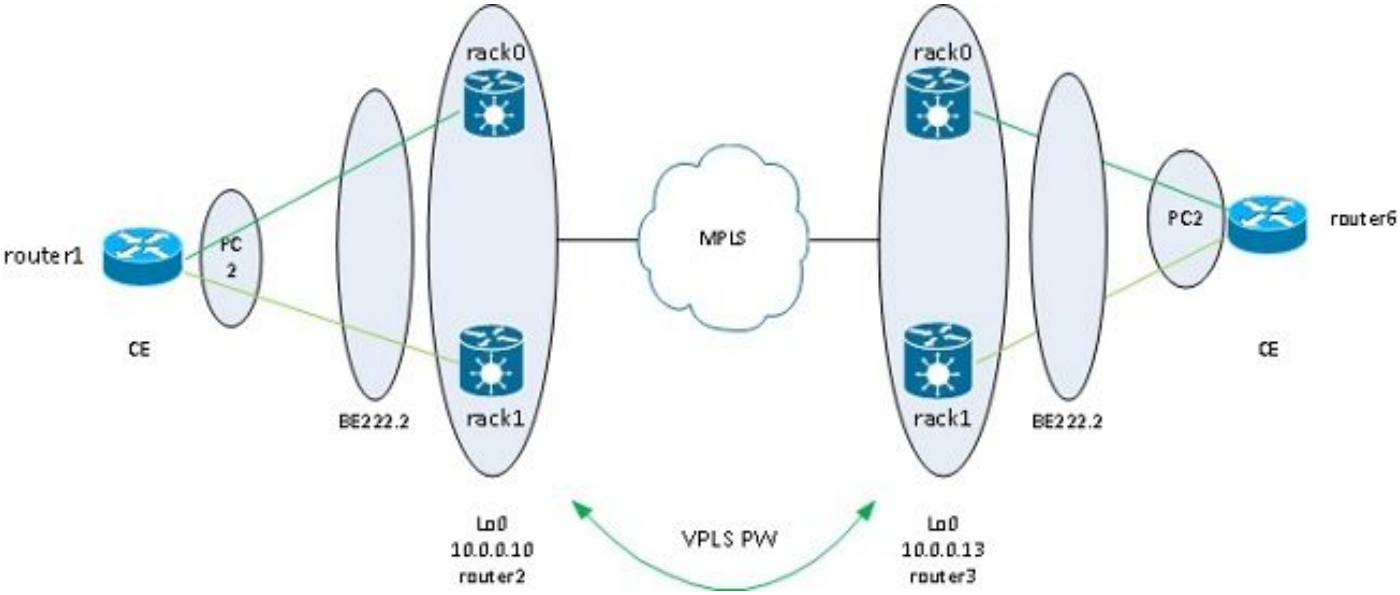
Another option is configuration of an ethernet-services access-list on the ACs on the bundle in order to drop the destination MAC addresses of the BPDUs so the BPDUs are not transported between sites. However, if a backdoor link is introduced between the sites, spanning tree cannot break the loop because it is not running on the MC-LAG bundle. So, evaluate carefully whether to disable spanning tree on the MC-LAG bundle. If the topology between sites is carefully maintained, it is nice to have redundancy through MC-LAG without

the need for spanning tree.

4.4.7.5 ASR 9000 nV Edge Cluster

The MC-LAG solution provided redundancy without the need to use spanning tree. One drawback is that the bundle members to one MC-LAG PE are in standby state, so it is an active-standby solution that does not maximize the link usage.

Another design option is use of an ASR 9000 nV Edge cluster so that CEs can have bundle members to each cluster rack that are all active at the same time:



Another benefit of this solution is that the number of PWs is reduced because there is only one PW per cluster for each of the clusters at each site. When there are two PEs per site, each PE must have a PW to each of the two PEs at each site.

The simplicity of the configuration is another benefit. The configuration looks like a very basic VPLS configuration with a bridge-domain with bundle ACs and VFI PWs:

```
RP/1/RSP0/CPU0:router2#sh bundle bundle-ether 222

Bundle-Ether222
  Status:                                  Up
  Local links :   2 / 0 / 2
  Local bandwidth :     20000000 (20000000) kbps
  MAC address (source):                    0024.f71e.d309 (Configured)
  Inter-chassis link:                      No
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    64
  Wait while timer:                        2000 ms
  Load balancing:                          Default
  LACP:                                    Not operational
    Flap suppression timer:                Off
```

```
    Cisco extensions:                       Disabled
  mLACP:                                    Not configured
  IPv4 BFD:                                 Not configured

  Port                  Device         State        Port ID         B/W, kbps
  --------------------  -------------  -----------  --------------  ----------
  Te0/0/0/8             Local          Active       0x8000, 0x0005    10000000
      Link is Active
  Te1/0/0/8             Local          Active       0x8000, 0x0001    10000000
      Link is Active

RP/1/RSP0/CPU0:router2#sh run int bundle-ether 222.2
interface Bundle-Ether222.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
!

RP/1/RSP0/CPU0:router2#sh run int bundle-ether 222.3
interface Bundle-Ether222.3 l2transport
 encapsulation dot1q 3
 rewrite ingress tag pop 1 symmetric
!

RP/1/RSP0/CPU0:router2#sh run l2vpn bridge group customer1
l2vpn
 bridge group customer1
  bridge-domain finance
   interface Bundle-Ether222.3
   !
   vfi customer1-finance
    neighbor 10.0.0.11 pw-id 3
    !
    neighbor 10.0.0.12 pw-id 3
    !
    neighbor 10.0.0.13 pw-id 3
    !
    neighbor 10.0.0.14 pw-id 3
    !
   !
  !
  bridge-domain engineering
   interface Bundle-Ether222.2
   !
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
    !
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/1/RSP0/CPU0:router2#sh l2vpn bridge-domain group customer1
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: finance, id: 3, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
```

```
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 4 (4 up), PBBs: 0 (0 up)
  List of ACs:
    BE222.3, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-finance (up)
      Neighbor 10.0.0.11 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.12 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 3, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 3, state: up, Static MAC addresses: 0
Bridge group: customer1, bridge-domain: engineering, id: 4, state: up,
   ShgId: 0, MSTi: 0
  Aging: 300 s, MAC limit: 4000, Action: none, Notification: syslog
  Filter MAC addresses: 0
  ACs: 1 (1 up), VFIs: 1, PWs: 4 (4 up), PBBs: 0 (0 up)
  List of ACs:
    BE222.2, state: up, Static MAC addresses: 0
  List of Access PWs:
  List of VFIs:
    VFI customer1-engineering (up)
      Neighbor 10.0.0.11 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.12 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.13 pw-id 2, state: up, Static MAC addresses: 0
      Neighbor 10.0.0.14 pw-id 2, state: up, Static MAC addresses: 0
```

Redundancy is provided by the bundle AC dual homed to the two racks so that the bundle remains up in case of bundle member failure or rack failure.

When a site is attached to the VPLS domain only through a cluster, the topology is similar to MC-LAG with regards to spanning tree. So spanning tree is not required on that bundle, and a BPDU filter could be configured on the CE in order to ensure that BPDUs are not exchanged between sites over VPLS.

Another option is configuration of an ethernet-services access-list on the ACs on the bundle in order to drop the destination MAC addresses of the BPDUs so the BPDUs are not transported between sites. However, if a backdoor link is introduced between the sites, spanning tree cannot break the loop because it is not running on the CE-PE bundle. So, evaluate carefully whether to disable spanning tree on that CE-PE bundle. If the topology between sites is carefully maintained, it is nice to have redundancy through the cluster without the need for spanning tree.

4.4.7.6 ICCP-Based Service Multi-Homing (ICCP-SM) (PMCLAG (Pseudo MCLAG) and Active/Active)

There is a new feature introduced in Version 4.3.1 in order to overcome the limitation of MC-LAG, where some bundle links are unused as they remain in standby mode. In the new feature, called Pseudo MCLAG, all of the links from the DHD to the points of attachments (PoA) are in use, but the VLANs are split between the different bundles.

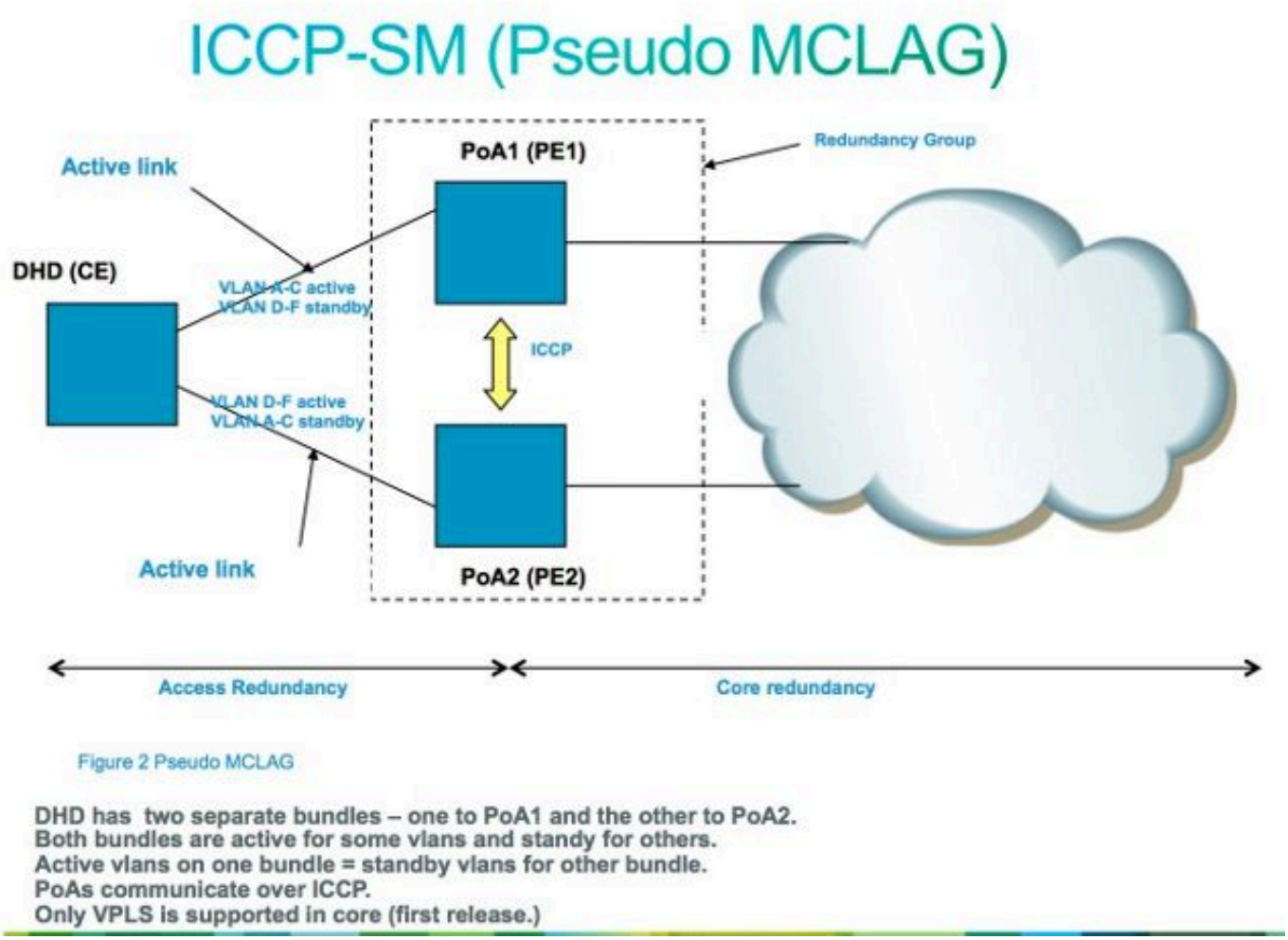


Figure 2 Pseudo MCLAG

4.5 Traffic Storm Control

In an L2 broadcast domain, there is the risk that a host can misbehave and send a high rate of broadcast or multicast frames that must be flooded everywhere in the bridge-domain. Another risk is creation of an L2 loop (that is not broken by spanning tree), which results in broadcasts and multicasts packets looping. A high rate of broadcasts and multicasts packets impacts the performance of the hosts in the broadcast domains.

Performance of switching devices in the network can also be affected by the replication of one input frame (broadcast, multicast or a unknown unicast frame) to multiple egress ports in the bridge-domain. Creation of multiple copies of the same packet can be resource-intensive, depending upon the place inside the device where the packet has to be replicated. For instance, replicating a broadcast to multiple different slots is not a problem because of the multicast replication capabilities of the fabric. The performance of a network processor can be impacted when it has to create multiple copies of the same packet to be sent on some ports that the network processor is handling.

In order to protect devices in case of a storm, the traffic storm control feature lets you configure a maximum rate of broadcasts, multicast and unknown unicasts to be accepted on a bridge-domain AC. See Implementing Traffic Storm Control under a VPLS Bridge for details.

Traffic storm control is not supported on a bundle AC interfaces or VFI PWs, but is supported on non-

bundle ACs and access PWs. The feature is disabled by default; unless you set up storm control, you accept any rate of broadcasts, multicast, and unknown unicasts.

Here is an example configuration:

<#root>

```
RP/0/RSP0/CPU0:router2#sh run l2vpn bridge group customer1 bridge-domain
   engineering
l2vpn
 bridge group customer1
  bridge-domain engineering
   interface GigabitEthernet0/1/0/3.2
    storm-control unknown-unicast pps 10000
    storm-control multicast pps 10000
    storm-control broadcast pps 1000
   !
   neighbor 10.0.0.15 pw-id 15
    storm-control unknown-unicast pps 10000
    storm-control multicast pps 10000
    storm-control broadcast pps 1000
   !
   vfi customer1-engineering
    neighbor 10.0.0.10 pw-id 2
    !
    neighbor 10.0.0.12 pw-id 2
    !
    neighbor 10.0.0.13 pw-id 2
    !
    neighbor 10.0.0.14 pw-id 2
    !
   !
  !
 !
!

RP/0/RSP0/CPU0:router2#sh l2vpn bridge-domain bd-name engineering det
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 5, state: up,
   ShgId: 0, MSTi: 0
  Coupled state: disabled
  MAC learning: enabled
  MAC withdraw: enabled
    MAC withdraw for Access PW: enabled
    MAC withdraw sent on bridge port down: disabled
  Flooding:
    Broadcast & Multicast: enabled
    Unknown unicast: enabled
  MAC aging time: 300 s, Type: inactivity
  MAC limit: 4000, Action: none, Notification: syslog
  MAC limit reached: no
  MAC port down flush: enabled
  MAC Secure: disabled, Logging: disabled
  Split Horizon Group: none
  Dynamic ARP Inspection: disabled, Logging: disabled
  IP Source Guard: disabled, Logging: disabled
  DHCPv4 snooping: disabled
  IGMP Snooping profile: none
```

```
  Bridge MTU: 1500
  MIB cvplsConfigIndex: 6
  Filter MAC addresses:
  Create time: 28/05/2013 17:17:03 (1w1d ago)
  No status change since creation
  ACs: 1 (1 up), VFIs: 1, PWs: 5 (5 up), PBBs: 0 (0 up)
  List of ACs:
    AC: GigabitEthernet0/1/0/3.2, state is up
      Type VLAN; Num Ranges: 1
      VLAN ranges: [2, 2]
      MTU 1500; XC ID 0xc40007; interworking none
      MAC learning: enabled
      Flooding:
        Broadcast & Multicast: enabled
        Unknown unicast: enabled
      MAC aging time: 300 s, Type: inactivity
      MAC limit: 4000, Action: none, Notification: syslog
      MAC limit reached: no
      MAC port down flush: enabled
      MAC Secure: disabled, Logging: disabled
      Split Horizon Group: none
      Dynamic ARP Inspection: disabled, Logging: disabled
      IP Source Guard: disabled, Logging: disabled
      DHCPv4 snooping: disabled
      IGMP Snooping profile: none

Storm Control:

       Broadcast: enabled(1000)

       Multicast: enabled(10000)

       Unknown unicast: enabled(10000)

      Static MAC addresses:
      Statistics:
        packets: received 251295, sent 3555258
        bytes: received 18590814, sent 317984884

Storm control drop counters:

        packets: broadcast 0, multicast 0, unknown unicast 0

        bytes: broadcast 0, multicast 0, unknown unicast 0

      Dynamic ARP inspection drop counters:
        packets: 0, bytes: 0
      IP source guard drop counters:
        packets: 0, bytes: 0
<snip>
```

The storm control drop counters are always present in the output of the **show l2vpn bridge-domain detail** command. Because the feature is disabled by default, the counters start to report drops only when the feature has been configured.

The configured rates can vary upon the traffic pattern from one network to another network. Before configuring a rate, Cisco recommends you understand the rate of broadcast, multicast or unknown unicast frames under normal circumstances. Then add a margin in the configured rate greater than the normal rate.

4.6 MAC Moves

In case of network instability like an interface flap, a MAC address can be learned from a new interface. This is normal network convergence, and the mac-address-table is updated dynamically.

However, constant MAC moves often indicate network instability, such as severe instability during an L2 loop. The MAC address security feature lets you report MAC moves and take corrective actions such as shutting down an offending port.

Even if a corrective action is not configured, you can configure the **logging** command so you are alerted of network instability through the MAC move messages:

```
l2vpn
 bridge group customer1
  bridge-domain engineering
   mac
    secure
     action none
     logging
    !
   !
```

In this example, the action is configured to none, so nothing is done when a MAC move is detected except that a syslog message is logged. This is an example message:

```
LC/0/0/CPU0:Dec 13 13:38:23.396 : l2fib[239]:
   %L2-L2FIB-5-SECURITY_MAC_SECURE_VIOLATION_AC : MAC secure in AC
   GigabitEthernet0_0_0_4.1310 detected violated packet - source MAC:
   0000.0000.0001, destination MAC: 0000.0001.0001; action: none
```

4.7 IGMP and MLD Snooping

By default, multicast frames are flooded to all ports in a bridge-domain. When you are using high rate streams like IP television (IPTV) services, there can be a significant amount of traffic forwarded on all ports

and replicated over multiple PWs. If all TV streams are forwarded over one interface, this can congest ports. The only option is configuration of a feature such as IGMP or MLD snooping, which intercepts multicast control packets in order to track the receivers and multicast routers and forward streams on the ports only when appropriate.

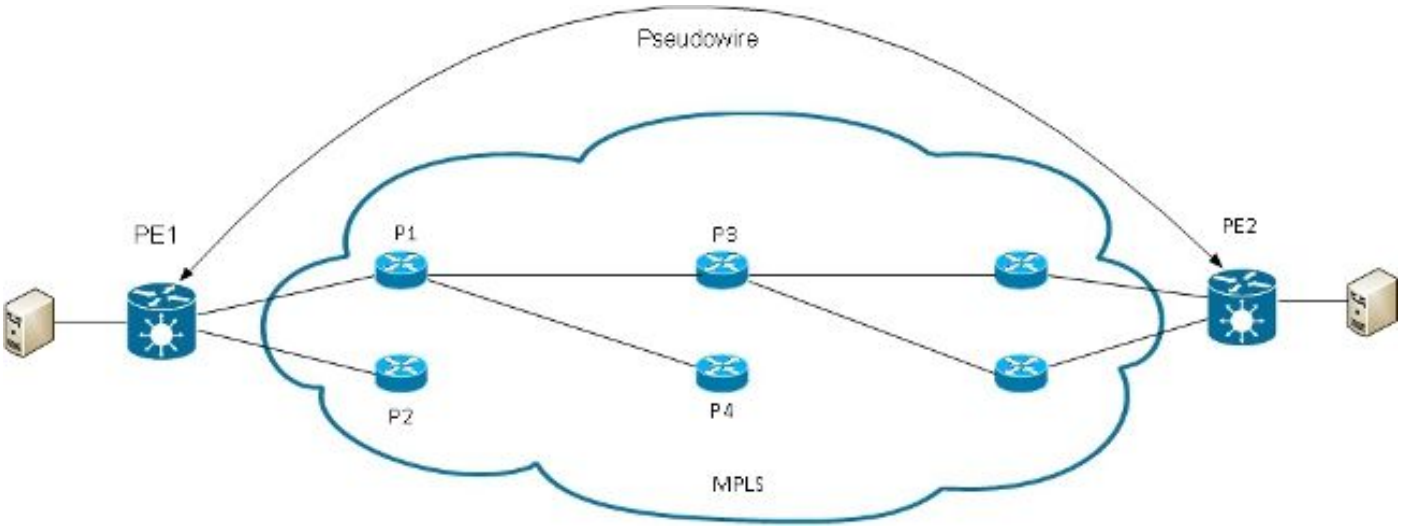
See the Multicast Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.9.x for further information on these features.

5. Additional L2VPN Topics

5.1 Loadbalancing

When an L2VPN PE needs to send a frame over an MPLS PW, the Ethernet frame is encapsulated into an MPLS frame with one or more MPLS labels; there is at least one PW label and perhaps an IGP label in order to reach the remote PE.

The MPLS frame is transported by the MPLS network to the remote L2VPN PE. There are typically multiple paths to reach the destination PE:



Note: Not all links are represented in this diagram.

PE1 can choose between P1 and P2 as the first MPLS P router towards PE2. If P1 is selected, PE1 then chooses between P3 and P4, and so on. The available paths are based on IGP topology and the MPLS TE tunnel path.

MPLS service providers prefer to have all links equally utilized rather than one congested link with other underutilized links. This goal is not always easy to achieve because some PWs carry much more traffic than others and because the path taken by a PW traffic depends upon the hashing algorithm used in the core. Multiple high bandwidth PWs can be hashed to the same links, which creates congestion.

A very important requirement is that all packets from one flow can follow the same path. Otherwise, this

leads to out-of-order frames, which can impact the quality or the performance of the applications.

Loadbalancing in an MPLS network on Cisco routers is typically based upon the data that follows the bottom MPLS label.

- If the data immediately after the bottom label starts with 0x4 or 0x6, an MPLS P router assumes that there is an IPv4 or IPv6 packet inside the MPLS packet and tries to loadbalance based on a hash of the source and destination IPv4 or IPv6 addresses extracted from the frame. In theory, this can not apply to an Ethernet frame that is encapsulated and transported over a PW because the destination MAC address follows the bottom label. But recently some MAC address ranges that start with 0x4 and 0x6 have been assigned. The MPLS P router can incorrectly consider that the Ethernet header is actually an IPv4 header and hash the frame based upon what it assumes is the IPv4 source and destination addresses. Ethernet frames from a PW can be hashed over different paths in the MPLS core, which leads to out-of-sequence frames in the PW and application quality issues. The solution is configuration of a control-word under a pw-class that can be attached to a point-to-point or VPLS PW. The control word is inserted immediately after the MPLS labels. The control word does not start with 0x4 or 0x6 so the problem is avoided.

<#root>

```
RP/1/RSP0/CPU0:router#sh run l2vpn bridge group customer1 bridge-domain
   engineering
l2vpn


pw-class control-word


  encapsulation mpls
   control-word
  !
 !
 bridge group customer1
  bridge-domain engineering
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
     pw-class control-word
    !
<snip>

RP/1/RSP0/CPU0:router#sh l2vpn bridge-domain bd-name engineering det
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 4, state: up,
   ShgId: 0, MSTi: 0
<snip>
  List of VFIs:
    VFI customer1-engineering (up)
      PW: neighbor 10.0.0.11, PW ID 2, state is up ( established )
        PW class control-word, XC ID 0xc000000a
        Encapsulation MPLS, protocol LDP
        Source address 10.0.0.10
        PW type Ethernet, control word enabled, interworking none
        Sequencing not set

        PW Status TLV in use
          MPLS         Local                          Remote
          ------------ ------------------------------ ------------------
```

```
Label        281708                         16043
Group ID     0x4                            0x5
Interface    customer1-engineering          customer1-engineering
MTU          1500                           1500
Control word enabled                        enabled
PW type      Ethernet                       Ethernet
VCCV CV type 0x2                            0x2
             (LSP ping verification)        (LSP ping verification)
VCCV CC type 0x7                            0x7
             (control word)                 (control word)
             (router alert label)           (router alert label)
             (TTL expiry)                   (TTL expiry)
------------ ------------------------------ ------------------
```

- If the data immediately after the bottom of the MPLS label stack does not start with 0x4 or 0x6, the P router loadbalances based upon the bottom label. All traffic from one PW follows the same path, so out-of-order packets do not occur, but this can lead to congestion on some links in case of high bandwidth PWs. With Cisco IOS XR Software Release 4.2.1, the ASR 9000 supports the Flow Aware Transport (FAT) PW feature. This feature runs on the L2VPN PEs, where it is negotiated between the two ends of a point-to-point or VPLS PW. The ingress L2VPN PE detects flows on the AC and the L2VPN configuration and inserts a new MPLS flow label under the PW MPLS label at the bottom of the MPLS label stack. The ingress PE detects flows based upon the source and destination MAC addresses (default) or the source and destination IPv4 addresses (configurable). Use of the MAC addresses is the default; use of IPv4 addresses is recommended, but must be configured manually.

  With the FAT PW feature, the ingress L2VPN PE inserts one bottom MPLS label per src-dst-mac or per src-dst-ip. The MPLS P routers (between the PEs) hash frames over the available paths, then reach the destination PE based on that FAT PW flow label at the bottom of the MPLS stack. This generally provides much better bandwidth utilization in the core unless a PW carries only a small number of src-dst-mac or src-dst-ip conversations. Cisco recommends that you use a control word so you can avoid having MAC addresses that start with 0x4 and 0x6 immediately after the flow label. This ensures the hash is correctly based upon the pseudo IP addresses and not based on the flow label.

  With this feature, the traffic from one PW is loadbalanced over multiple paths in the core when available. Application traffic does not suffer from out-of-order packets because all traffic from the same source (MAC or IP) to the same destination (MAC or IP) follows the same path.

This is an example configuration:

```
l2vpn
 pw-class fat-pw
  encapsulation mpls
   control-word
   load-balancing
    flow-label both
   !
  !
 !
 bridge group customer1
  bridge-domain engineering
   vfi customer1-engineering
    neighbor 10.0.0.11 pw-id 2
     pw-class fat-pw
```

```
RP/1/RSP0/CPU0:router#sh l2vpn bridge-domain bd-name engineering det
Legend: pp = Partially Programmed.
Bridge group: customer1, bridge-domain: engineering, id: 4, state: up,
   ShgId: 0, MSTi: 0
<snip>
  List of VFIs:
    VFI customer1-engineering (up)
      PW: neighbor 10.0.0.11, PW ID 2, state is up ( established )
        PW class fat-pw, XC ID 0xc000000a
        Encapsulation MPLS, protocol LDP
        Source address 10.0.0.10
        PW type Ethernet, control word enabled, interworking none
        Sequencing not set
        Load Balance Hashing: src-dst-ip
        Flow Label flags configured (Tx=1,Rx=1), negotiated (Tx=1,Rx=1)

        PW Status TLV in use
          MPLS         Local                          Remote
          ------------ ------------------------------ ------------------
          Label        281708                         16043
          Group ID     0x4                            0x5
          Interface    customer1-engineering          customer1-engineering
          MTU          1500                           1500
          Control word enabled                        enabled
          PW type      Ethernet                       Ethernet
          VCCV CV type 0x2                            0x2
                       (LSP ping verification)        (LSP ping verification)
          VCCV CC type 0x7                            0x7
                       (control word)                 (control word)
                       (router alert label)           (router alert label)
                       (TTL expiry)                   (TTL expiry)
          ------------ ------------------------------ ------------------
```

5.2 Logging

Different types of logging messages can be configured in the L2VPN configuration mode. Configure l2vpn logging in order to receive syslog alerts for L2VPN events, and configure logging pseudowire in order to determine when a PW status changes:

```
<#root>
l2vpn


logging


  bridge-domain


pseudowire


  nsr
 !
```

If many PWs are configured, messages can flood the log.

5.3 ethernet-services access-list

You can use an ethernet-services access-list in order to drop traffic from specific hosts or verify if a router is getting packets from a host on an l2transport interface:

<#root>

```
RP/0/RSP0/CPU0:router#sh run
```

ethernet-services access-list

```
 count-packets
ethernet-services access-list count-packets
 10 permit host 001d.4603.1f42 host 0019.552b.b5c3
 20 permit any any
!

RP/0/RSP0/CPU0:router#sh run int gig 0/1/0/3.2
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
 ethernet-services access-group count-packets egress
!

RP/0/RSP0/CPU0:router#sh access-lists ethernet-services count-packets
```

hardware

egress

```
 location 0/1/CPU0
ethernet-services access-list count-packets
 10 permit host 001d.4603.1f42 host 0019.552b.b5c3 (5 hw matches)
 20 permit any any (30 hw matches)
```

The hardware matches can be seen only with the hardware keyword. Use the ingress or egress keyword depending upon the direction of the access-group. The linecard location of the interface where the access-list is applied is also specified.

You can also apply an ipv4 access-list on a l2transport interface as a security or troubleshooting feature:

<#root>

```
RP/0/RSP0/CPU0:router#sh run
```

ipv4 access-list

```
 count-pings
ipv4 access-list count-pings
 10 permit icmp host 192.168.2.1 host 192.168.2.2
 20 permit ipv4 any any
!

RP/0/RSP0/CPU0:router#sh run int gig 0/1/0/3.2
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
 rewrite ingress tag pop 1 symmetric
 ipv4 access-group count-pings ingress
!

RP/0/RSP0/CPU0:router#sh access-lists ipv4 count-pings hardware ingress
   location 0/1/CPU0
ipv4 access-list count-pings
 10 permit icmp host 192.168.2.1 host 192.168.2.2 (5 hw matches)
 20 permit ipv4 any any (6 hw matches)
```

5.4 ethernet egress-filter

In the egress direction of an AC, suppose there is no **rewrite ingress tag pop <> symmetric** command that determines the egress VLAN tags. In that case, there is no check in order to ensure that the outgoing frame has the correct VLAN tags according to the **encapsulation** command.

This is an example configuration:

```
interface GigabitEthernet0/1/0/3.2 l2transport
 encapsulation dot1q 2
!
interface GigabitEthernet0/1/0/3.3 l2transport
 encapsulation dot1q 3
!
interface GigabitEthernet0/1/0/39.2 l2transport
 encapsulation dot1q 2
!
l2vpn
 bridge group customer2
  bridge-domain test
   interface GigabitEthernet0/1/0/3.2
   !
   interface GigabitEthernet0/1/0/3.3
   !
   interface GigabitEthernet0/1/0/39.2
   !
  !
 !
!
```

In this configuration:

- A broadcast received with a dot1q tag 2 on GigabitEthernet0/1/0/39.2 keeps its incoming tag because there is no **rewrite ingress** command.
- That broadcast is flooded out of GigabitEthernet0/1/0/3.2 with its dot1q tag 2, but that does not cause a problem because GigabitEthernet0/1/0/3.2 is also configured with the dot1q tag 2.
- That broadcast is also flooded out of GigabitEthernet0/1/0/3.3, which keeps its original tag 2 because there is no **rewrite** command on GigabitEthernet0/1/0/3.3. The **encapsulation dot1q 3** command on GigabitEthernet0/1/0/3.3 is not checked in the egress direction.
- The result is that, for one broadcast received with tag 2 on GigabitEthernet0/1/0/39, there are two broadcasts with tag 2 going out of GigabitEthernet0/1/0/3. That duplicated traffic can cause some application issues.
- The solution is configuration of ethernet egress-filter strict in order to ensure that packets leave the subinterface with the correct VLAN tags. Otherwise, the packets are not forwarded and are dropped.

<#root>

```
interface GigabitEthernet0/1/0/3.2 l2transport

ethernet egress-filter strict

!
interface GigabitEthernet0/1/0/3.3 l2transport

ethernet egress-filter strict

!
```