Configuring Virtual Private LAN Service (VPLS)

• Finding Feature Information, on page 1
• Configuring VPLS, on page 1

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Configuring VPLS

Information About VPLS

VPLS Overview

VPLS (Virtual Private LAN Service) enables enterprises to link together their Ethernet-based LANs from multiple sites via the infrastructure provided by their service provider. From the enterprise perspective, the service provider's public network looks like one giant Ethernet LAN. For the service provider, VPLS provides an opportunity to deploy another revenue-generating service on top of their existing network without major capital expenditures. Operators can extend the operational life of equipment in their network.

Virtual Private LAN Services (VPLS) uses the provider core to join multiple attachment circuits together to simulate a virtual bridge that connects the multiple attachment circuits together. From a customer point of view, there is no topology for VPLS. All of the CE devices appear to connect to a logical bridge emulated by the provider core.
Full-Mesh Configuration

The full-mesh configuration requires a full mesh of tunnel label switched paths (LSPs) between all the PEs that participate in the VPLS. With full-mesh, signaling overhead and packet replication requirements for each provisioned VC on a PE can be high.

You set up a VPLS by first creating a virtual forwarding instance (VFI) on each participating PE router. The VFI specifies the VPN ID of a VPLS domain, the addresses of other PE devices in the domain, and the type of tunnel signaling and encapsulation mechanism for each peer PE router.

The set of VFIs formed by the interconnection of the emulated VCs is called a VPLS instance; it is the VPLS instance that forms the logic bridge over a packet switched network. The VPLS instance is assigned a unique VPN ID.

The PE devices use the VFI to establish a full-mesh LSP of emulated VCs to all the other PE devices in the VPLS instance. PE devices obtain the membership of a VPLS instance through static configuration using the Cisco IOS CLI.

The full-mesh configuration allows the PE router to maintain a single broadcast domain. Thus, when the PE router receives a broadcast, multicast, or unknown unicast packet on an attachment circuit, it sends the packet out on all other attachment circuits and emulated circuits to all other CE devices participating in that VPLS instance. The CE devices see the VPLS instance as an emulated LAN.

To avoid the problem of a packet looping in the provider core, the PE devices enforce a "split-horizon" principle for the emulated VCs. That means if a packet is received on an emulated VC, it is not forwarded on any other emulated VC.

After the VFI has been defined, it needs to be bound to an attachment circuit to the CE device.

The packet forwarding decision is made by looking up the Layer 2 virtual forwarding instance (VFI) of a particular VPLS domain.

A VPLS instance on a particular PE router receives Ethernet frames that enter on specific physical or logical ports and populates a MAC table similarly to how an Ethernet switch works. The PE router can use the MAC address to switch those frames into the appropriate LSP for delivery to the another PE router at a remote site.

If the MAC address is not in the MAC address table, the PE router replicates the Ethernet frame and floods it to all logical ports associated with that VPLS instance, except the ingress port where it just entered. The PE router updates the MAC table as it receives packets on specific ports and removes addresses not used for specific periods.
Restrictions for VPLS

- Protocol-based CLI Method (interface pseudowire configuration) is not supported. Only VFI and Xconnect mode are supported.
- Flow-Aware Transport Pseudowire (FAT PW) is not supported.
- IGMP Snooping is not Supported. Multicast traffic floods with IGMP Snooping disabled.
- L2 Protocol Tunneling is not supported.
- Integrated Routing and Bridging (IRB) not supported.
- Virtual Circuit Connectivity Verification (VCCV) ping with explicit null is not supported.
- The switch is supported only as spoke in H-VPLS but not as hub.
- L2 VPN Interworking is not supported.
- `ip unnumbered` command is not supported in MPLS configuration.
- VC statistics are not displayed for flood traffic in the output of show mpls l2 vc vcid detail command.
- dot1q tunnel is not supported in the attachment circuit.

Configuring PE Layer 2 Interfaces to CEs

Configuring 802.1Q Trunks for Tagged Traffic from a CE

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `no ip address ip_address mask [secondary ]`
5. `switchport`
6. `switchport trunk encapsulation dot1q`
7. `switchport trunk allow vlan vlan_ID`
8. `switchport mode trunk`
9. `end`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device&gt; <code>enable</code></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 3** interface interface-id | Defines the interface to be configured as a trunk, and enters interface configuration mode. |
| Example: Device(config)# interface TenGigabitEthernet1/0/24 | |

| **Step 4** no ip address ip_address mask [secondary ] | Disables IP processing and enters interface configuration mode. |
| Example: Device(config-if)# no ip address | |

| **Step 5** switchport | Modifies the switching characteristics of the Layer 2-switched interface. |
| Example: Device(config-if)# switchport | |

| **Step 6** switchport trunk encapsulation dot1q | Sets the switch port encapsulation format to 802.1Q. |
| Example: Device(config-if)# switchport trunk encapsulation dot1q | |

| **Step 7** switchport trunk allow vlan vlan_ID | Sets the list of allowed VLANs. |
| Example: Device(config-if)# switchport trunk allow vlan 2129 | |

| **Step 8** switchport mode trunk | Sets the interface to a trunking VLAN Layer 2 interface. |
| Example: Device(config-if)# switchport mode trunk | |

| **Step 9** end | Returns to privileged EXEC mode. |
| Example: Device(config)# end | |
Configuring 802.1Q Access Ports for Untagged Traffic from a CE

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `no ip address ip_address mask [secondary ]`
5. `switchport`
6. `switchport mode access`
7. `switchport access vlan vlan_ID`
8. `end`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1 | `enable`  
**Example:**  
Device> enable | Enables privileged EXEC mode.  
* Enter your password if prompted. |
| Step 2 | `configure terminal`  
**Example:**  
Device# configure terminal | Enters global configuration mode. |
| Step 3 | `interface interface-id`  
**Example:**  
Device(config)# interface TenGigabitEthernet1/0/24 | Defines the interface to be configured as a trunk, and enters interface configuration mode. |
| Step 4 | `no ip address ip_address mask [secondary ]`  
**Example:**  
Device(config-if)# no ip address | Disables IP processing and enters interface configuration mode. |
| Step 5 | `switchport`  
**Example:**  
Device(config-if)# switchport | Modifies the switching characteristics of the Layer 2-switched interface. |
| Step 6 | `switchport mode access`  
**Example:**  
Device(config-if)# switchport mode access | Sets the interface type to nontrunking, nontagged single VLAN Layer 2 interface. |
**Command or Action**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 7</strong></td>
<td><code>switchport access vlan vlan_ID</code>&lt;br&gt;<code>Example:</code>&lt;br&gt;<code>Device(config-if)# switchport access vlan 2129</code></td>
<td>Sets the VLAN when the interface is in access mode.</td>
</tr>
<tr>
<td><strong>Step 8</strong></td>
<td><code>end</code>&lt;br&gt;<code>Example:</code>&lt;br&gt;<code>Device(config)# end</code></td>
<td>Returns to privileged EXEC mode.</td>
</tr>
</tbody>
</table>

### Configuring Layer 2 VLAN Instances on a PE

Configuring the Layer 2 VLAN interface on the PE enables the Layer 2 VLAN instance on the PE router to the VLAN database to set up the mapping between the VPLS and VLANs.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `vlan vlan-id`
4. `interface vlan vlan-id`
5. `end`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>enable</code>&lt;br&gt;<code>Example:</code>&lt;br&gt;<code>Device&gt; enable</code></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>configure terminal</code>&lt;br&gt;<code>Example:</code>&lt;br&gt;<code>Device# configure terminal</code></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><code>vlan vlan-id</code>&lt;br&gt;<code>Example:</code>&lt;br&gt;<code>Device(config)# vlan 2129</code></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><code>interface vlan vlan-id</code>&lt;br&gt;<code>Example:</code>&lt;br&gt;<code>Device(config)# interface vlan 2129</code></td>
</tr>
</tbody>
</table>
Configuring MPLS in the PE

To configure MPLS in the PE, you must provide the required MPLS parameters.

SUMMARY STEPS

1. enable
2. configure terminal
3. mpls ip
4. mpls label protocol ldp
5. end
6. mpls ldp logging neighbor-changes

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>mpls ip</td>
<td>Configures MPLS hop-by-hop forwarding.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# mpls ip</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>mpls label protocol ldp</td>
<td>Specifies the default Label Distribution Protocol for a platform.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-vlan)# mpls label protocol ldp</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring VFI in the PE

The virtual switch instance (VFI) specifies the VPN ID of a VPLS domain, the addresses of other PE devices in this domain, and the type of tunnel signaling and encapsulation mechanism for each peer (This is where you create the VFI and associated VCs.). Configure a VFI as follows:

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `l2 vfi vfi-name manual`
4. `vpn id vpn-id`
5. `neighbor remote-router-id {encapsulation mpls}`
6. `end`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Step 1</strong> Example:</td>
<td></td>
</tr>
<tr>
<td><code>Device&gt;</code> <code>enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong> Example:</td>
<td></td>
</tr>
<tr>
<td><code>Device# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>l2 vfi vfi-name manual</code></td>
<td>Enables the Layer 2 VFI manual configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong> Example:</td>
<td></td>
</tr>
<tr>
<td><code>Device(config)# l2 vfi 2129 manual</code></td>
<td></td>
</tr>
</tbody>
</table>
### Purpose

**Command or Action**

**Step 4**

```
vpn id vpn-id
```

**Example:**

```
Device(config-vfi)# vpn id 2129
```

Configures a VPN ID for a VPLS domain. The emulated VCs bound to this Layer 2 VRF use this VPN ID for signaling.

**Note**  
`vpn-id` is the same as `vlan-id`.

**Step 5**

```
neighbor remote-router-id {encapsulation mpls}
```

**Example:**

```
Device(config-vfi)# neighbor remote-router-id {encapsulation mpls}
```

Specifies the remote peering router ID and the tunnel encapsulation type or the pseudo-wire property to be used to set up the emulated VC.

**Step 6**

```
end
```

**Example:**

```
Device(config)# end
```

Returns to privileged EXEC mode.

### Associating the Attachment Circuit with the VFI at the PE

After defining the VFI, you must bind it to one or more attachment circuits.

#### SUMMARY STEPS

1. enable
2. configure terminal
3. interface vlan vlan-id
4. no ip address
5. xconnect vfi vfi-name
6. end

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

<p>| <strong>Step 2</strong> configure terminal | Enters global configuration mode. |
| Example:                     |                                   |
| Device# configure terminal   |                                   |</p>
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3** | **interface vlan vlan-id**<br>**Example:**<br>Device(config)# interface vlan 2129 | Creates or accesses a dynamic switched virtual interface (SVI).  
**Note**: `vlan-id` is the same as `vpn-id`. |
| **Step 4** | **no ip address**<br>**Example:**<br>Device(config-if)# no ip address | Disables IP processing. (You configure a Layer 3 interface for the VLAN if you configure an IP address.) |
| **Step 5** | **xconnect vfi vfi-name**<br>**Example:**<br>Device(config-if)# xconnect vfi 2129 | Specifies the Layer 2 VFI that you are binding to the VLAN port. |
| **Step 6** | **end**<br>**Example:**<br>Device(config)# end | Returns to privileged EXEC mode. |

**Configuration Examples for VPLS**

*Figure 2: VPLS Topology*
<table>
<thead>
<tr>
<th>PE1 Configuration</th>
<th>PE2 Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>pseudowire-class vpls2129</td>
<td>pseudowire-class vpls2129</td>
</tr>
<tr>
<td>encapsulation mpls</td>
<td>encapsulation mpls</td>
</tr>
<tr>
<td>!</td>
<td>no control-word</td>
</tr>
<tr>
<td>12 vfi 2129 manual</td>
<td>12 vfi 2129 manual</td>
</tr>
<tr>
<td>vpn id 2129</td>
<td>vpn id 2129</td>
</tr>
<tr>
<td>neighbor 44.254.44.44 pw-class vpls2129</td>
<td>neighbor 1.1.1.72 pw-class vpls2129</td>
</tr>
<tr>
<td>neighbor 188.98.89.98 pw-class vpls2129</td>
<td>neighbor 188.98.89.98 pw-class vpls2129</td>
</tr>
<tr>
<td>!</td>
<td>!</td>
</tr>
<tr>
<td>interface TenGigabitEthernet1/0/24</td>
<td>interface TenGigabitEthernet1/0/47</td>
</tr>
<tr>
<td>switchport trunk allowed vlan 2129</td>
<td>switchport trunk allowed vlan 2129</td>
</tr>
<tr>
<td>switchport mode trunk</td>
<td>switchport mode trunk</td>
</tr>
<tr>
<td>!</td>
<td>!</td>
</tr>
<tr>
<td>interface Vlan2129</td>
<td>interface Vlan2129</td>
</tr>
<tr>
<td>no ip address</td>
<td>no ip address</td>
</tr>
<tr>
<td>xconnect vfi 2129</td>
<td>xconnect vfi 2129</td>
</tr>
</tbody>
</table>

The `show mpls 12transport vc detail` command provides information about the virtual circuits.

Local interface: VFI 2129 vfi up
   Interworking type is Ethernet
   Destination address: 44.254.44.44, VC ID: 2129, VC status: up
   Output interface: Gi1/0/9, imposed label stack (18 17)
   Preferred path: not configured
   Default path: active
   Next hop: 177.77.177.2
   Create time: 19:09:33, last status change time: 09:24:14
   Last label FSM state change time: 09:24:14
   Signaling protocol: LDP, peer 44.254.44.44:0 up
   Targeted Hello: 1.1.1.72(LDP Id) -> 44.254.44.44, LDP is UP
   Graceful restart: configured and enabled
   Non stop routing: not configured and not enabled
   Status TLV support (local/remote) : enabled/supported
   LDP route watch : enabled
   Label/status state machine : established, LruRru
   Last local dataplane status rcvd: No fault
   Last BFD dataplane status rcvd: Not sent
   Last BFD peer monitor status rcvd: No fault
   Last local AC circuit status rcvd: No fault
   Last local AC circuit status sent: No fault
   Last local PW i/f circ status rcvd: No fault
   Last local LDP TLV status sent: No fault
   Last remote LDP TLV status rcvd: No fault
   Last remote LDP ADJ status rcvd: No fault
MPLS VC labels: local 512, remote 17
   Group ID: local n/a, remote 0
   MTU: local 1500, remote 1500
   Remote interface description:
   Sequencing: receive disabled, send disabled
   Control Word: Off
SSO Descriptor: 44.254.44.44/2129, local label: 512
Dataplane:
  SSM segment/switch IDs: 20498/20492 (used), PWID: 2
VC statistics:
  transit packet totals: receive 0, send 0
  transit byte totals: receive 0, send 0
  transit packet drops: receive 0, seq error 0, send 0

The `show l2vpn atom vc` shows that ATM over MPLS is configured on a VC.

pseudowire100005 is up, VC status is up PW type: Ethernet
Create time: 19:25:56, last status change time: 09:40:37
Last label FSM state change time: 09:40:37
Destination address: 44.254.44.44 VC ID: 2129
  Output interface: Gi1/0/9, imposed label stack {18 17}
  Preferred path: not configured
  Default path: active
  Next hop: 177.77.177.2
Member of vfi service 2129
  Bridge-Domain id: 2129
  Service id: 0x32000003
Signaling protocol: LDP, peer 44.254.44.44:0 up
  Targeted Hello: 1.1.1.72(LDP Id) -> 44.254.44.44, LDP is UP
Graceful restart: configured and enabled
Non stop routing: not configured and not enabled
PWid FEC (128), VC ID: 2129
  Status TLV support (local/remote): enabled/supported
    LDP route watch: enabled
    Label/status state machine: established, LruRru
    Local dataplane status received: No fault
    BFD dataplane status received: Not sent
    BFD peer monitor status received: No fault
    Status received from access circuit: No fault
    Status sent to access circuit: No fault
    Status received from pseudowire i/f: No fault
  Status sent to network peer: No fault
    Status received from network peer: No fault
    Adjacency status of remote peer: No fault
Sequencing: receive disabled, send disabled

Bindings

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Local</th>
<th>Remote</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>512</td>
<td>17</td>
</tr>
<tr>
<td>Group ID</td>
<td>n/a</td>
<td>0</td>
</tr>
<tr>
<td>Interface</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTU</td>
<td>1500</td>
<td>1500</td>
</tr>
<tr>
<td>Control word</td>
<td>off</td>
<td>off</td>
</tr>
<tr>
<td>PW type</td>
<td>Ethernet</td>
<td>Ethernet</td>
</tr>
<tr>
<td>VCCV CV type</td>
<td>0x02</td>
<td>0x02</td>
</tr>
</tbody>
</table>

VCCV CC type 0x06 0x06
Status TLV enabled supported
SSO Descriptor: 44.254.44.44/2129, local label: 512
Dataplane:
SSM segment/switch IDs: 20498/20492 (used), PWID: 2
Rx Counters
0 input transit packets, 0 bytes
0 drops, 0 seq err
Tx Counters
0 output transit packets, 0 bytes
0 drops