

# Chapter 15

## Virtual Private LAN Service

Ethernet is an increasingly important component of a service provider's slate of service offerings. Many customers are requesting the ability to connect local area network (LAN) locations across the country and around the world. To fulfill customer desire, service providers have had to set up complex point-to-point Layer 2 virtual private networks (VPNs) or connect expensive Layer 2 switches to handle traffic.

In JUNOS Release 5.7 and later, an emerging service is available to meet the growing Ethernet needs of service providers and their customers. Virtual Private LAN Service (VPLS), based on the Internet Engineering Task Force (IETF) draft of the same name (draft-kompella-ppvnp-vpls-02.txt), is an Ethernet-based multipoint-to-multipoint Layer 2 virtual private network (VPN). With VPLS, multiple Ethernet LAN sites can be connected to each other across an MPLS backbone. To the customer, all sites interconnected by VPLS appear to be on the same Ethernet LAN (even though traffic travels across a service provider network).

This guide explains the background knowledge you need to understand VPLS and provides detailed steps for you to follow to implement it in your network.

This VPLS feature guide covers these topics:

Overview on page 489

System Requirements on page 492

Terms and Acronyms on page 492

Configure VPLS on page 493

Example: VPLS Configuration on page 496

Check Your Work on page 502

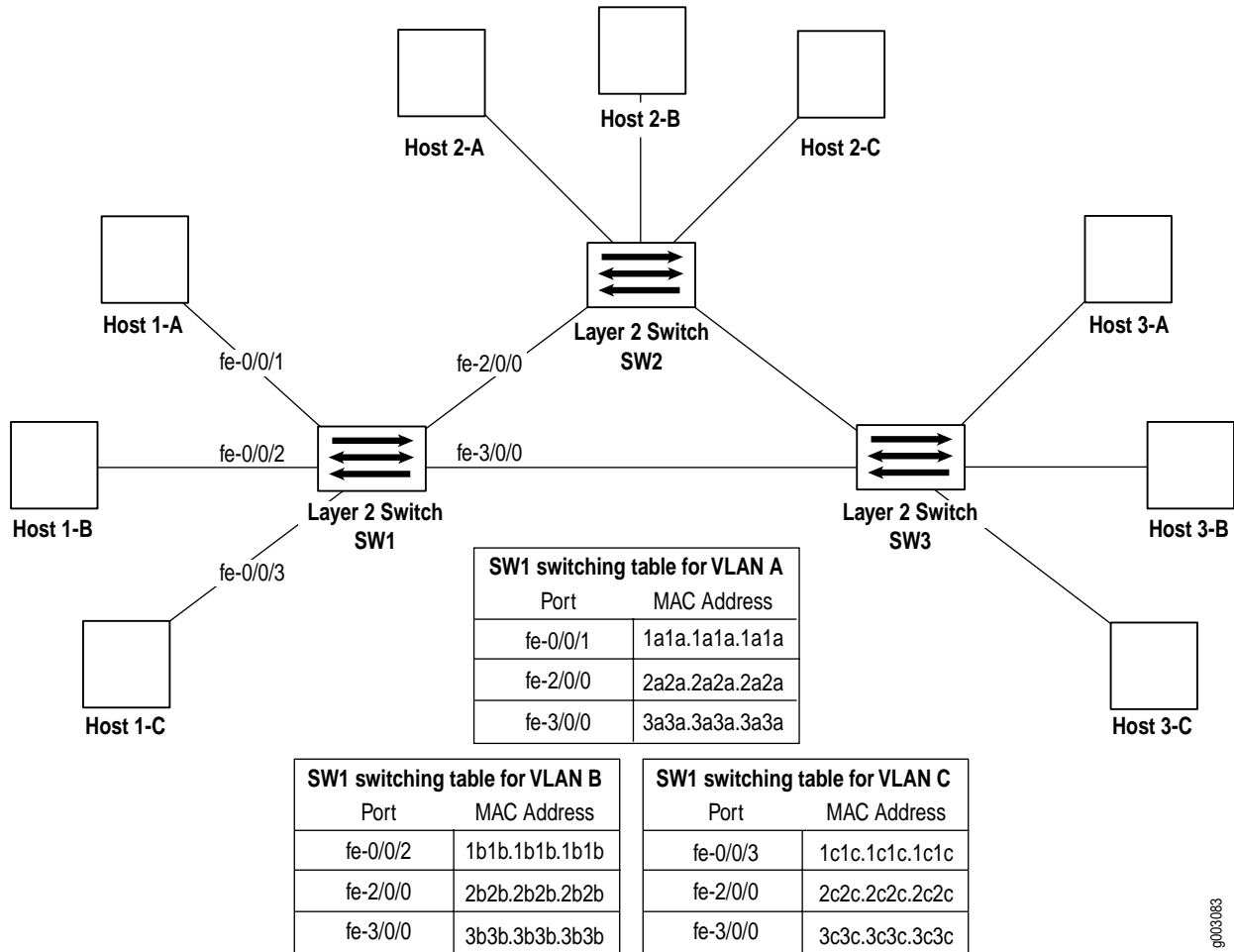
For More Information on page 506

Revision History on page 506

### Overview

Before VPLS, the only way you could connect Ethernet LAN sites together was to set up a Layer 2 VPN or install multiple Layer 2 Ethernet switches. Figure 53 on page 490 shows how three switches can be connected to each other.

Figure 53: Ethernet Switching Example

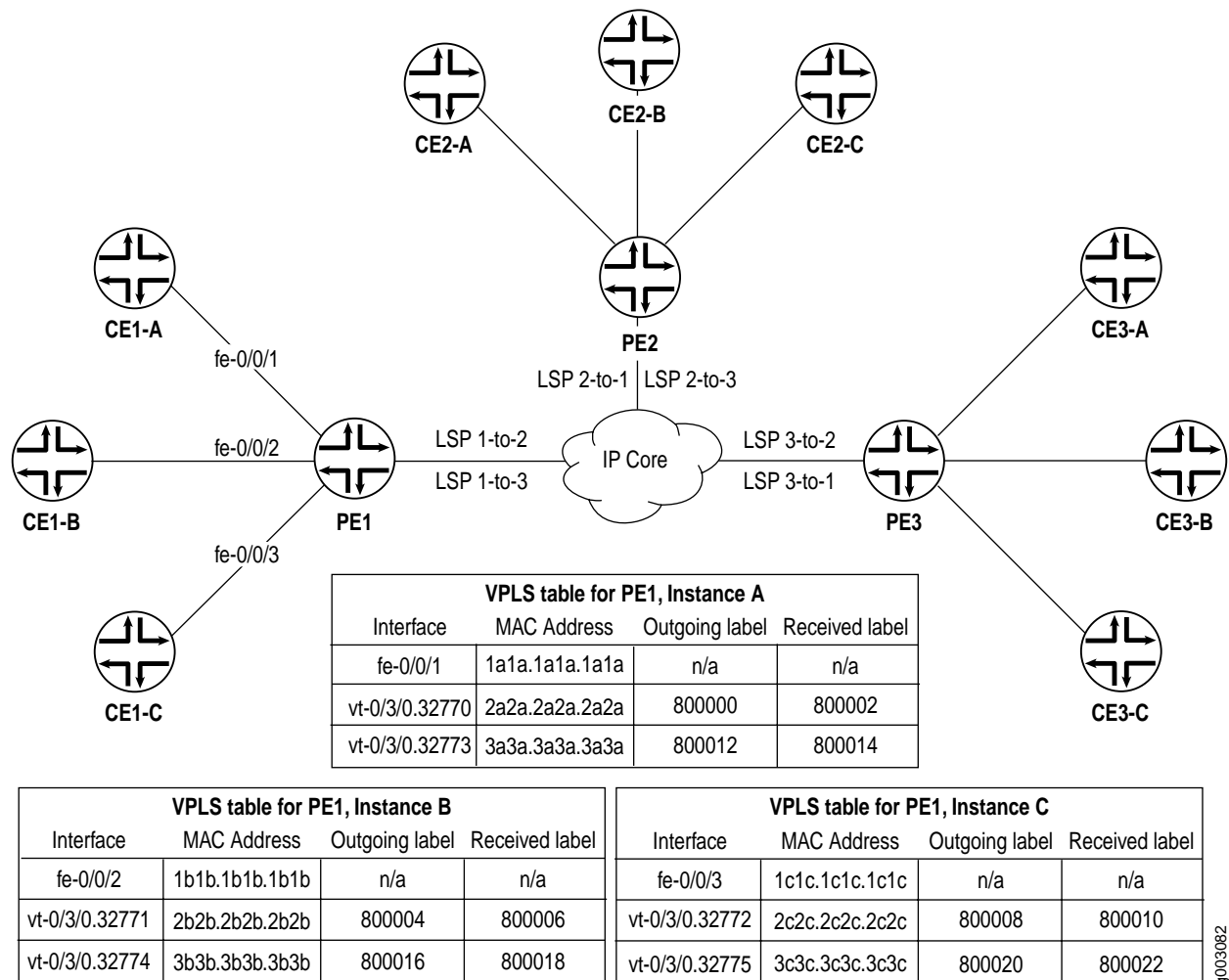


g003083

A typical switch builds its Layer 2 switching table with MAC address and interface information learned from other switches. If a switch does not know how to reach a particular destination, it floods traffic for that destination to all ports except the one where the traffic originated. When a reply for an unknown destination is received, this information is added to the switching table. If a destination is known, the switch sends the traffic directly to the intended recipient through the associated port listed in the switching table.

Figure 54 on page 491 shows a VPLS network comparable to the switch example and explains how VPLS functions similarly to Ethernet switches.

Figure 54: VPLS Introductory Example



g003082

Notice that Layer 2 information gathered by a switch (for example, MAC addresses and interface ports) is included in the VPLS instance table. However, instead of requiring all VPLS interfaces to be physical switch ports, the router allows remote traffic for a VPLS instance to be delivered across an MPLS label-switched path (LSP) and arrive on a virtual port. The virtual port emulates a local, physical port. Traffic can be learned, forwarded, or flooded to the virtual port almost identically to the way traffic is sent to a local port.

The VPLS table learns MAC address and interface information for both physical and virtual ports. If no activity is seen for a particular MAC address, it is purged from the table over time.

As shown in Figure 54, the main difference between a physical port and a virtual port is that the router captures additional information from the virtual port—an outgoing MPLS label used to reach the remote site and an incoming MPLS label for VPLS traffic received from the remote site.

When you configure VPLS on the router, the virtual port is generated dynamically if you have installed a PIC that supports virtual tunnels. Consequently, you must install at least one Tunnel PIC or Link Services PIC in each VPLS provider edge (PE) router.

One restriction to flooding behavior in VPLS is that traffic received from remote PE routers is never forwarded to other PE routers. This restriction helps prevent loops in the core network. However, if a customer edge (CE) Ethernet switch has two connections or more to the same PE router, you must enable the Spanning Tree Protocol on the CE switch to prevent loops. (Spanning tree is not supported directly on M-series routers.)



**Note**

Juniper Networks routers support transmission of standard Bridge Protocol Data Unit (BPDU) frames across Layer 2 VPNs, Layer 2 circuits, and VPLS instances. However, customer edge (CE) Ethernet switches that generate proprietary BPDU frames might not be able to run the Spanning Tree Protocol across Juniper Networks routers configured for these emulated Layer 2 connections.

## System Requirements

To implement VPLS, your system must meet these minimum requirements:

JUNOS software Release 6.0 or later for Ethernet VPLS over ATM LLC interface encapsulation

JUNOS software Release 5.7 or later for Ethernet VPLS, VLAN VPLS, and extended VLAN VPLS interface encapsulations

Two Juniper Networks M5, M10, M20, M40, or M40e routers for the provider edge (PE)

One Tunnel Services PIC or Link Services PIC per router

One Fast Ethernet or Gigabit Ethernet PIC per router (from this list):

Four-port Fast Ethernet PIC with 10/100 Base-TX interfaces

One-port Gigabit Ethernet PIC

Two-port Gigabit Ethernet PIC

Four-port, quad-wide Gigabit Ethernet PIC

## Terms and Acronyms

**VPLS (Virtual Private LAN Service)**—An Ethernet-based multipoint-to-multipoint Layer 2 VPN service used for interconnecting multiple Ethernet LANs across an MPLS backbone. VPLS is specified in the IETF draft “*Virtual Private LAN Service*” (draft-kompella-ppvnp-vpls-01.txt). For more information about VPLS, see the *JUNOS Internet Software Configuration Guide: VPNs*.

**virtual port**—A special virtual loopback tunnel interface that is generated dynamically when you configure VPLS on a PE router. Virtual ports send and receive VPLS traffic for remote PE routers as if the remote VPLS sites had Ethernet-based interfaces directly connected to the local PE router. (To generate virtual ports, VPLS PE routers require a PIC that can generate tunnel interfaces— such as the Tunnel Services PIC or Link Services PIC.)

## Configure VPLS

To implement VPLS, you must configure the following:

Configure BGP, MPLS, RSVP, and an IGP on the PE and Core Routers on page 493

Configure VPLS Encapsulation on CE-Facing Interfaces on page 493

Configure a VPLS Routing Instance on page 494

To apply your knowledge, visit these sections:

Example: VPLS Configuration on page 496

Check Your Work on page 502

### **Configure BGP, MPLS, RSVP, and an IGP on the PE and Core Routers**

At a fundamental level, VPLS is a type of Layer 2 VPN. All forms of Layer 2 VPNs require you to configure network protocols to handle *intradomain routing* (an interior gateway protocol, or IGP, such as Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS) ), *interdomain routing* (Border Gateway Protocol, or BGP), *label switching* (Multiprotocol Label Switching, or MPLS), and *path signaling* (Resource Reservation Protocol, or RSVP). For more information about these protocols and examples of how to configure these protocols to support a Layer 2 VPN, see the *JUNOS Internet Software Configuration Guide: VPNs*.



**Note**

The 12-port, 24-port, and 48-port dense Fast Ethernet Physical Interface Cards (PICs) cannot push more than two labels onto an MPLS packet. Because of this limitation, we do not recommend that you configure these PICs as core-facing interfaces or equivalent interfaces that need to push more than two labels onto an MPLS packet.

### **Configure VPLS Encapsulation on CE-Facing Interfaces**

There are four types of VPLS interface encapsulation: Ethernet VPLS, Ethernet VPLS over ATM LLC, VLAN VPLS, and extended VLAN VPLS. When one of these encapsulations is applied to an interface, a family type of VPLS is enabled by default. The encapsulation types are:

**ether-vpls-over-atm-llc**—Use Ethernet VPLS over ATM LLC encapsulation on ATM 2 logical interfaces. This encapsulation type enables a VPLS instance to support bridging between Ethernet interfaces and ATM interfaces, as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. When you use this encapsulation type, you can configure it on logical interfaces only and you cannot configure multipoint interfaces.

**ethernet-vpls**—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and must accept packets carrying standard Tag Protocol ID (TPID) values.

**extended-vlan-vpls**—Use extended VLAN VPLS encapsulation on Ethernet interfaces that have VLAN 802.1Q tagging and VPLS enabled and that must accept packets carrying TPIDs 0x8100, 0x9100, and 0x9901.

**vlan-vpls**—Use VLAN VPLS encapsulation on Ethernet interfaces with VLAN tagging enabled. VLAN VPLS encapsulation supports TPID 0x8100 only. You must configure this encapsulation type on both the physical interface and the logical interface.

Use the following guidelines to configure a VPLS interface:

For encapsulation type **vlan-vpls**, VLAN IDs 1 through 511 are reserved for normal Ethernet VLANs, IDs 512 through 1023 are reserved for VPLS VLANs on Fast Ethernet interfaces, and IDs 512 through 4094 are reserved for VPLS VLANs on Gigabit Ethernet interfaces. For encapsulation type **extended-vlan-vpls**, all VLAN IDs from 1 through 4094 are valid for VPLS VLANs. VLAN ID 0 is reserved for priority tagging.

For VLAN-based VPLS, you can configure only one VLAN ID per VPLS instance.

To configure VPLS interface encapsulation for an Ethernet interface, include the encapsulation statement at the [edit interfaces *interface-fpc/pic/port*] hierarchy level and select **ethernet-vpls**, **vlan-vpls**, or **extended-vlan-vpls** as the encapsulation type. If you select the VLAN VPLS encapsulation, also include the **vlan-vpls** statement at the [edit interfaces *ethernet-interface-fpc/pic/port* unit *unit-number* encapsulation] logical interface hierarchy level. When using either VLAN VPLS or extended VLAN VPLS encapsulations, include the **vlan-tagging** statement at the [edit interfaces *ethernet-interface-fpc/pic/port*] hierarchy level.

To configure VPLS interface encapsulation for an ATM 2 interface, include the encapsulation statement at the [edit interfaces *interface-fpc/pic/port*] hierarchy level and select **ether-vpls-over-atm-llc** as the encapsulation type.

```
[edit]
interfaces {
  fe-0/1/0 {
    vlan-tagging;
    encapsulation vlan-vpls;
    unit 0 {
      encapsulation vlan-vpls;
      vlan-id 600;
    }
  }
}
```

## Configure a VPLS Routing Instance

Like other Layer 2 VPNs, you must enable a routing instance to isolate VPLS traffic from other network traffic. To configure, include the **instance-type vpls** statement at the [edit routing-instances *instance-name*] hierarchy level.

Within the instance, you can define the maximum number of sites that can participate in this VPLS instance, the size of the MAC address table, a local site name, and a local site identifier. To configure the maximum number of sites, include the **site-range** statement at the [edit routing-instances *instance-name* protocols vpls] hierarchy level. To configure the size of the MAC address table, include the **mac-table-size** statement at the [edit routing-instances *instance-name* protocols vpls] hierarchy level. The default size is 512 addresses, the minimum is 16 addresses, and the maximum is 65,536 addresses.

To configure a site name, include the site statement at the [edit routing-instances *instance-name* protocols vpls] hierarchy level. To configure the site ID, include the site-identifier *number* statement at the [edit routing-instances *instance-name* protocols vpls site *name*] hierarchy level.

```
[edit]
routing-instances
  green {
    instance-type vpls;
    interface fe-0/1/0.0;
    route-distinguisher 10.245.14.218:1;
    vrf-target target:11111:1;
    protocols {
      vpls {
        site-range 10;
        mac-table-size 1024;
        site greenPE1 {
          site-identifier 1;
        }
      }
    }
  }
}
```

### **Option: Select an LSP for the VPLS Routing Instance to Traverse**

If you have two or more equal-cost-path LSPs between your VPLS PE router sites, you can select an LSP over which the VPLS traffic will travel. You can assign the VPLS routing instance to a BGP community, define a policy that directs community traffic over a specified LSP, and then apply the policy to the forwarding table.

To configure a BGP community, include the community *community-name* statement at the [edit policy-options] hierarchy level. Be sure to specify the vrf-export or vrf-target values from the VPLS routing instance as community identifiers with the members *community-ids* statement at the [edit policy-options community *community-name*] hierarchy level.

To create a policy that sends community traffic over a specific LSP, include the community *community-name* statement at the [edit policy-options policy-statement *policy-name* term *term-name* from] hierarchy level and the install-nexthop lsp *lsp-name* statement at the [edit policy-options policy-statement *policy-name* term *term-name* then] hierarchy level. To apply the policy to the forwarding table, include the export *policy-name* statement at the [edit routing-options forwarding-table] hierarchy level.

```
[edit]
routing-options {
  autonomous-system 69;
  forwarding-table {
    export LSP-policy;
  }
}
policy-options {
  policy-statement LSP-policy {
    term a {
      from community gold;
      then {
        install-nexthop lsp pe1-to-pe2;
        accept;
      }
    }
  }
}
```

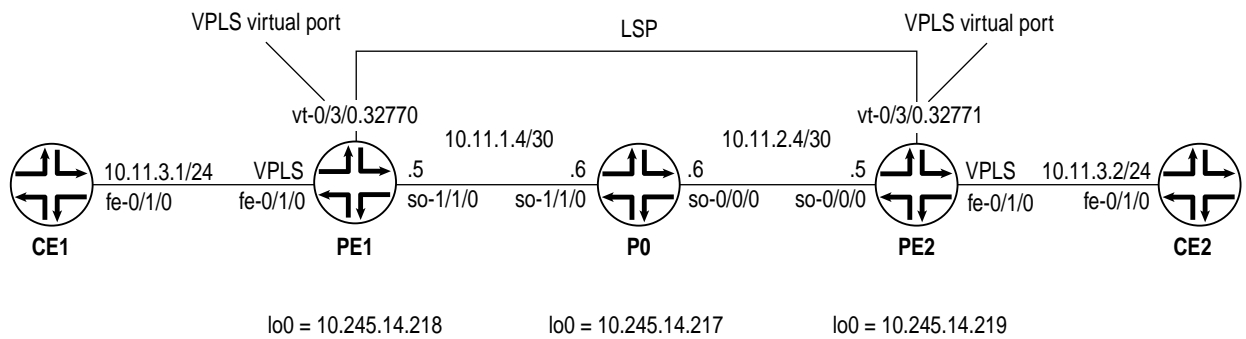
```

    }
  }
  community gold members target:11111:1;
}

```

### Example: VPLS Configuration

Figure 55: VPLS Topology Diagram



VPLS table for PE1			
Interface	MAC Address	Outgoing label	Received label
fe-0/1/0	aaaa.aaaa.aaaa	n/a	n/a
vt-0/3/0.32770	bbbb.bbbb.bbbb	800000	800002

g003084

In Figure 55, a simple VPLS topology is enabled between routers PE1 and PE2. CE routers CE1 and CE2 use Ethernet-based interfaces to connect VLAN 600 to their local PE router. The PE routers PE1 and PE2 are connected to one another by LSPs enabled across a service provider backbone running MPLS, BGP, RSVP, and OSPF.

In a VPLS routing instance named green, PE1 has a local interface fe-0/1/0 and a virtual port of vt-0/3/0.32770 (the virtual port is created dynamically on the Tunnel PIC when VPLS is configured). PE2 has a local interface fe-0/1/0 and a virtual port of vt-0/3/0.32771 in the same green instance. As a result, routers CE1 and CE2 can send Ethernet traffic to one another as if they are physically connected to each other on a LAN.

On router CE1, the only item you need to configure is the Fast Ethernet interface that connects to PE1. Be sure to write down the VLAN identifier and IP address, so you can match them later on CE2.

```

Router CE1 [edit]
interfaces {
  fe-0/1/0 {
    vlan-tagging;          # Configure VLAN tagging for VLAN VPLS or extended VLAN VPLS.
    unit 0 {
      vlan-id 600;         # The Ethernet-based interface on CE2 must use the same VLAN ID.
      family inet {
        address 10.11.3.1/24; # The interface on CE2 must use the same network prefix.
      }
    }
  }
}

```

On router PE1, prepare the router for VPLS by configuring BGP, MPLS, OSPF, and RSVP (These protocols are the basis for most Layer 2 VPN-related applications, including VPLS). Include the family l2vpn statement at the [edit protocols bgp group *group-name*] hierarchy level, because VPLS uses the same infrastructure for internal BGP as used for Layer 2 VPNs.

Next, configure VLAN tagging on the Fast Ethernet interface connected to router CE1. Include VLAN VPLS encapsulation at both the physical and logical interface levels. Be sure to use the same VLAN ID for all Ethernet interfaces that are part of a single VPLS instance. Finally, add the Fast Ethernet interface into a VPLS routing instance and specify the site range, site ID number, and site name.

```

Router PE1 [edit]
interfaces {
  fe-0/1/0 {
    vlan-tagging;           # Configure VLAN tagging for VLAN VPLS or extended VLAN VPLS.
    encapsulation vlan-vpls; # Configure VPLS encapsulation on the physical interface.
    unit 0 {
      encapsulation vlan-vpls; # This encapsulation is also configured on the logical interface.
      vlan-id 600;           # The VLAN ID is the same one used by the CE routers.
    }                       # No IP address is needed on the CE-facing interface.
  }
  so-1/1/0 {
    unit 0 {
      family inet {
        address 10.11.1.5/30;
      }
      family mpls;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.245.14.218/32;
      }
    }
  }
}
routing-options {
  autonomous-system 69;
  forwarding-table {
    export exp-to-fwd; # This applies the policy that selects an LSP for the VPLS instance.
  }
}

```

```

protocols {
  rsvp {
    interface all {
      aggregate;
    }
  }
  mpls {
    label-switched-path pe1-to-pe2 { # Configure an LSP to reach other VPLS PE routers.
      to 10.245.14.219;
    }
    interface all;
  }
  bgp {
    group vpls-pe {
      type internal;
      local-address 10.245.14.218;
      family l2vpn { # VPLS uses the same infrastructure as Layer 2 VPNs for internal BGP
        unicast;
      }
      neighbor 10.245.14.217;
      neighbor 10.245.14.219;
    }
  }
  ospf {
    traffic-engineering;
    area 0.0.0.0 {
      interface 10.11.1.5 {
        metric 11;
      }
      interface lo0.0 {
        passive;
      }
    }
  }
}
policy-options {
  policy-statement exp-to-fwd {
    term a {
      from community grn-com; # This matches the BGP community carrying the VPLS instance.
      then {
        install-nexthop lsp pe1-to-pe2; # If there are multiple LSPs that exist between VPLS PE
        accept; # routers, this statement sends VPLS traffic over a specific LSP
      }
    }
  }
  community grn-com members target:11111:1; # This adds the instance into a BGP community.
}

```

```

routing-instances
green {
  instance-type vpls;           # Configure a VPLS routing instance.
  interface fe-0/1/0.0;
  route-distinguisher 10.245.14.218:1;
  vrf-target target:11111:1;# This value is important when you configure the BGP community.
  protocols {
    vpls {                     # Configure a VPLS site range, site name, and site identifier.
      site-range 10;
      site greenPE1 {
        site-identifier 1;
      }
    }
  }
}

```

On router P0, configure BGP, MPLS, OSPF, and RSVP to interconnect PE1 and PE2.

```

Router P0 [edit]
interfaces {
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.11.2.6/30;
      }
      family mpls;
    }
  }
  so-1/1/0 {
    unit 0 {
      family inet {
        address 10.11.1.6/30;
      }
      family mpls;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.245.14.217/32;
      }
    }
  }
}
protocols {
  rsvp {
    interface all {
      aggregate;
    }
  }
  mpls {
    interface all;
  }
}

```

```

bgp {
  group vpls-pe {
    type internal;
    local-address 10.245.14.217;
    family l2vpn {      # VPLS uses the same infrastructure as Layer 2 VPNs for internal BGP
      unicast;
    }
    neighbor 10.245.14.218;
    neighbor 10.245.14.219;
  }
}
ospf {
  traffic-engineering;
  area 0.0.0.0 {
    interface 10.11.1.6 {
      metric 11;
    }
    interface 10.11.2.6 {
      metric 15;
    }
    interface lo0.0 {
      passive;
    }
  }
}
}

```

On router PE2, configure BGP, MPLS, OSPF, and RSVP to complement the configuration on PE1. Next, configure VLAN tagging on the Fast Ethernet interface connected to router CE2. Include VLAN VPLS encapsulation at both the physical and logical interface levels. Be sure to use the same VLAN ID for all Ethernet interfaces that are part of a single VPLS instance. Finally, add the Fast Ethernet interface into a VPLS routing instance and specify the site range, site ID number, and site name.

```

Router PE2 [edit]
interfaces {
  fe-0/1/0 {
    vlan-tagging;          # Configure VLAN tagging for VLAN VPLS or extended VLAN VPLS.
    encapsulation vlan-vpls; # Configure VPLS encapsulation on the physical interface.
    unit 0 {
      encapsulation vlan-vpls;# This encapsulation is also configured on the logical interface.
      vlan-id 600;         # The VLAN ID is the same one used by the CE routers.
    }
    # No IP address is needed on the CE-facing interface.
  }
  so-0/0/0 {
    unit 0 {
      family inet {
        address 10.11.2.5/30;
      }
      family mpls;
    }
  }
}

```

```

lo0 {
  unit 0 {
    family inet {
      address 10.245.14.219/32;
    }
  }
}
}
routing-options {
  autonomous-system 69;
  forwarding-table {
    export exp-to-fwd; # This applies the policy that selects an LSP for the VPLS instance.
  }
}
protocols {
  rsvp {
    interface all {
      aggregate;
    }
  }
  mpls {
    label-switched-path pe2-to-pe1 { # Configure an LSP to other VPLS PE routers.
      to 10.245.14.218;
    }
    interface all;
  }
  bgp {
    group vpls-pe {
      type internal;
      local-address 10.245.14.219;
      family l2vpn { # VPLS uses the same infrastructure as Layer 2 VPNs for internal BGP
        unicast;
      }
      neighbor 10.245.14.217;
      neighbor 10.245.14.218;
    }
  }
  ospf {
    traffic-engineering;
    area 0.0.0.0 {
      interface 10.11.2.5 {
        metric 15;
      }
      interface lo0.0 {
        passive;
      }
    }
  }
}
}
policy-options {
  policy-statement exp-to-fwd {
    term a {
      from community grn-com; # This matches the BGP community carrying the VPLS instance.
      then {
        install-nexthop lsp pe2-to-pe1; # If there are multiple LSPs that exist between VPLS PE
        accept; # routers, this statement sends VPLS traffic over a specific LSP
      }
    }
  }
}
}
community grn-com members target:11111:1; # This adds the instance into a BGP community.
}

```

```

routing-instances
green {
  instance-type vpls;           # Configure a VPLS routing instance.
  interface fe-0/1/0.0;
  route-distinguisher 10.245.14.219:1;
  vrf-target target:11111:1;# This value is important when you configure the BGP community.
  protocols {
    vpls {                      # Configure a VPLS site range, site name, and site identifier.
      site-range 10;
      site greenPE2 {
        site-identifier 2;
      }
    }
  }
}

```

On router CE2, complete your VPLS network by configuring the Fast Ethernet interface that connects to PE2. Use the same VLAN identifier and IP address prefix used on router CE1.

```

Router CE2 [edit]
interfaces {
  fe-0/1/0 {
    vlan-tagging;              # Configure VLAN tagging for VLAN VPLS or extended VLAN VPLS.
    unit 0 {
      vlan-id 600;             # The Ethernet-based interface on CE1 must use the same VLAN ID.
      family inet {
        address 10.11.3.2/24;   # The interface on CE1 must use the same network prefix.
      }
    }
  }
}

```

## Check Your Work

To verify proper operation of VPLS, use the following commands:

```

show vpls connections

show route forwarding-table family mpls

show route forwarding-table family vpls

show interfaces terse

show route instance (detail)

show system statistics vpls

```

The following section shows the output of these commands on router PE1 as a result of the configuration example:

```

user@PE1> show interfaces terse
Interface          Admin Link Proto Local                               Remote
so-1/1/0           up    up
so-1/1/0.0         up    up   inet  10.11.1.5/30
                               mpls

so-1/1/1           up    up
so-1/1/2           up    up
so-1/1/3           up    up
fe-0/1/0           up    up
fe-0/1/0.0        up   up   vpls   # This is the local Fast Ethernet interface.
fe-0/1/1           up    up
fe-0/1/2           up    up
fe-0/1/3           up    up
gr-0/3/0           up    up
ip-0/3/0           up    up
mt-0/3/0           up    up
pd-0/3/0           up    up
pe-0/3/0           up    up
vt-0/3/0           up    up
vt-0/3/0.32770   up   up   # This is the dynamically generated virtual port.
dsc                up    up
fxp0               up    up
fxp0.0             up    up   inet  192.186.14.218/24
fxp1               up    up
fxp1.0             up    up   tnp   4
gre                up    up
ipip               up    up
lo0                up    up
lo0.0              up    up   inet  10.245.14.218      --> 0/0
                               127.0.0.1         --> 0/0
                               inet6 fe80::2a0:a5ff:fe28:13e0
                               feee::10:245:14:218

lsi                up    up
mtun               up    up
pimd               up    up
pime               up    up
tap                up    up

user@PE1> show system statistics vpls
vpls:
    0 total packets received
    0 with size smaller than minimum
    0 with incorrect version number
    0 packets for this host

    0 packets with no logical interface
    0 packets with no family
    0 packets with no route table
    0 packets with no auxiliary table
    0 packets with no corefacing entry
    0 packets with no CE-facing entry

    6 mac route learning requests   # This indicates that VPLS is working.
    6 mac routes learnt
    0 mac routes aged
    0 mac routes moved

```

```

user@PE1> show route forwarding-table family vpls extensive
Routing table: green.vpls [Index 2]
VPLS:

Destination: default
  Route type: dynamic          Route reference: 0
  Flags: sent to PFE
  Next-hop type: flood          Index: 353      Reference: 1

Destination: default
  Route type: permanent        Route reference: 0
  Flags: none
  Next-hop type: discard        Index: 298      Reference: 1

Destination: fe-0/1/0.0
  Route type: dynamic          Route reference: 0
  Flags: sent to PFE
  Next-hop type: flood          Index: 355      Reference: 1

Destination: bb:bb:bb:bb:bb:bb/48      # This MAC address belongs to remote CE2.
  Route type: dynamic          Route reference: 0
  Flags: sent to PFE, prefix load balance
  Next-hop type: indirect      Index: 351      Reference: 4
  Next-hop type: Push 800000, Push 100002(top)
  Next-hop interface: so-1/1/0.0

Destination: aa:aa:aa:aa:aa:aa/48      # This MAC address belongs to local CE1.
  Route type: dynamic          Route reference: 0
  Flags: sent to PFE, prefix load balance
  Next-hop type: unicast       Index: 354      Reference: 2
  Next-hop interface: fe-0/1/0.0

user@PE1> show route forwarding-table family vpls
Routing table: green.vpls
VPLS:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          dymn  0         next hop          flood  353   1
default          perm  0         next hop          dscd  298   1
fe-0/1/0.0       dymn  0         next hop          flood  355   1
bb:bb:bb:bb:bb:bb/48      # This MAC address belongs to remote CE2.
                  dymn  0         next hop          indr   351   4
                  Push 800000, Push 100002(top)
so-1/1/0.0
aa:aa:aa:aa:aa:aa/48      # This MAC address belongs to local CE1.
                  dymn  0         next hop          ucst   354   2 fe-0/1/0.0

user@PE1> show route forwarding-table family mpls
Routing table: mpls
MPLS:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0         next hop          dscd  19    1
0                user  0         next hop          recv  18    3
1                user  0         next hop          recv  18    3
2                user  0         next hop          recv  18    3
100000           user  0 10.11.1.6        swap  100001 so-1/1/0.0
800002           user  0         next hop          Pop           vt-0/3/0.32770
vt-0/3/0.32770 (VPLS)
                  user  0         next hop          indr   351   4
                  Push 800000, Push 100002(top)
so-1/1/0.0

```

```

user@PE1> show route instance green detail
green:
  Router ID: 0.0.0.0
  Type: vpls                               State: Active
Interfaces:
  fe-0/1/0.0                               # This is the local Fast Ethernet interface.
  vt-0/3/0.32770                           # This is the dynamically generated VPLS virtual port.
  Route-distinguisher: 10.245.14.218:1
  Vrf-import: [ __vrf-import-green-internal__ ]
  Vrf-export: [ __vrf-export-green-internal__ ]
  Vrf-import-target: [ target:11111:1 ]
  Vrf-export-target: [ target:11111:1 ]
  Tables:
    green.l2vpn.0                          : 2 routes (2 active, 0 holddown, 0 hidden)

```

```

user@PE1> show vpls connections

```

```

L2VPN Connections:

```

```

Legend for connection status (St)

```

```

OR -- out of range                WE -- intf encaps != instance encaps
EI -- encapsulation invalid       Dn -- down
EM -- encapsulation mismatch      VC-Dn -- Virtual circuit down
CM -- control-word mismatch       -> -- only outbound conn is up
CN -- circuit not present         <- -- only inbound conn is up
OL -- no outgoing label           Up -- operational
NC -- intf encaps not CCC/TCC     XX -- unknown
NP -- interface not present

```

```

Legend for interface status

```

```

Up -- operational
Dn -- down

```

```

Instance: green

```

```

Local site: greenPE1 (1)

```

```

connection-site      Type  St      Time last up      # Up trans
2                    rmt   Up      Jan 24 06:26:49 2003              1

```

```

Local interface: vt-0/3/0.32770, Status: Up, Encapsulation: VPLS

```

```

Remote PE: 10.245.14.219, Negotiated control-word: No

```

```

Incoming label: 800002, Outgoing label: 800000

```

```

user@PE1> show system statistics vpls

```

```

vpls:

```

```

0 total packets received
0 with size smaller than minimum
0 with incorrect version number
0 packets for this host

0 packets with no logical interface
0 packets with no family
0 packets with no route table
0 packets with no auxiliary table
0 packets with no corefacing entry
0 packets with no CE-facing entry

7 mac route learning requests
7 mac routes learnt
0 mac routes aged
0 mac routes moved

```

```
user@PE1> show route instance green detail
green:
  Router ID: 0.0.0.0
  Type: vpls                               State: Active
Interfaces:
  fe-0/1/0.0
  vt-0/3/0.32770
  Route-distinguisher: 10.245.14.218:1
  Vrf-import: [ __vrf-import-green-internal__ ]
  Vrf-export: [ __vrf-export-green-internal__ ]
  Vrf-import-target: [ target:11111:1 ]
  Vrf-export-target: [ target:11111:1 ]
  Tables:
    green.l2vpn.0                          : 2 routes (2 active, 0 holddown, 0 hidden)
```

## For More Information

For additional information about VPLS, see the following:

*JUNOS Internet Software Configuration Guide: VPNs*

*JUNOS Internet Software Configuration Guide: Network Interfaces and Class of Service*

*JUNOS Internet Software Configuration Guide: Routing and Routing Protocols*

K. Kompella, et. al., “Virtual Private LAN Service,” Internet draft, draft-kompella-ppvpn-vpls-02.txt, May 2003

D. Grossman and J. Heinanen, “Multiprotocol Encapsulation over ATM Adaptation Layer 5,” RFC 2684, September 1999

## Revision History

30 June 2003—Added the ether-vpls-over-atm-llc interface encapsulation type and LSP selection for VPLS instances, 6.0R1 Release. Elizabeth Lichtenberg and Richard Hendricks.

2 April 2003—Initial document written, 5.7R1. Richard Hendricks.