VPLS over GRE

Virtual Private LAN Service (VPLS) enables geographically separate LAN segments to be interconnected as a single bridged domain over an MPLS network (VPLS can only be enabled on an MPLS network).

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for VPLS over GRE

- Load-balancing at the VPLS ingress or at the core is not supported for flood or multicast traffic.
- Interior Gateway Protocol (IGP) load balance and flow aware transport of MPLS pseudowires (FAT PW) are not supported for EoMPLS.
- Virtual circuit connection verification (VCCV) over FAT PW is not supported, neither will IGP load balance work for VCCV.

- Configuring scheme 2 of VPLS over GRE by using the **platform vpls gre favor-performance** command is not supported for VPLS/EoMPLS over GRE on MPLS cloud. MPLS should not be enabled on the underlying physical interface that carries the GRE traffic.

Information About VPLS over GRE

VPLS over GRE Overview

Virtual Private LAN Service (VPLS) enables geographically separate LAN segments to be interconnected as a single bridged domain over an MPLS network (VPLS can only be enabled on an MPLS network). Generic routing encapsulation (GRE) is a tunneling protocol developed by Cisco that can encapsulate a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link to Cisco routers at remote points over an IP internetwork. VPLS over GRE then enables VPLS across an IP network. The provider edge (PE) routers for VPLS over GRE must support VPLS and additional GRE encapsulation/decapsulation. The PE routers can be placed in customer sites. For example, different sites of a data center (DC) can have L2 service across an IP network. The PE router can also be placed at the edge of an IP core cloud if a service provider wants to provide L2 service for customers.

A VPLS instance must be configured on each PE router. GRE tunnels are configured to connect PEs across an IP network. MPLS virtual circuit (VC) labels are transported by the MPLS-enabled GRE tunnels. To support the fully meshed pseudowires (PW), GRE tunnels must be fully meshed among PE routers. A pseudowire defines a VLAN and its corresponding pseudoport.

VPLS over GRE Data Plane

In the data plane, the L2 Ethernet frames arrive at the ingress interface on the PE1 router. A VC label is imposed on the Ethernet frame and then the GRE header is encapsulated. An explicit null label could be imposed if the **mpls ldp explicit-null** command is issued on this router. The PE1 router switches the packets to the appropriate interface, which will route the packets to the egress PE2/PE3 routers. When packets arrive at the egress PE2/PE3 routers, the PE2/PE3 routers must decapsulate the GRE header of the IP packets, perform label disposition, L2 lookup, and forward the frame to the appropriate egress interface.

VPLS over GRE Encapsulation

VPLS over GRE requires at least one recirculation at the ingress router (Because of a hardware limitation, hardware cannot encapsulate the VC label + MPLS label + GRE header + L2 rewrite in one packet pass. Packets may travel back to the data path to finish the encapsulation). Packet recirculation is a specific means for packets to travel back to the data path. Two schemes to achieve these recirculations exist.

In scheme one, for remote unicast, two recirculations are required. The first pass handles VC label and MPLS label encapsulation. The hardware must do the recirculation with a shim header indicating the destination index of the GRE tunnel encapsulation adjacency entry. An MTU check is performed in the first pass. The second pass handles GRE encapsulation. In this pass, the GRE header and IP header are added. In addition, the egress features on the GRE tunnel, such as ACL and QoS, are handled in this pass. The hardware must do a second recirculation with a shim header indicating the destination index of the L2 rewrite adjacency entry. The third pass handles L2 rewrite. In the third pass, IPv4 lookup is performed and hits an adjacency that programs a new L2 MAC address.

In scheme two, one recirculation is required. The first pass handles VC label and MPLS label encapsulation. The hardware must do the recirculation. The second pass performs IP + GRE encapsulation and provides a new destination media access control (DMAC). The egress logical interface (LIF) is the physical outgoing interface LIF.

The advantage of scheme two is that scheme two has better performance because of one less pass than scheme one in EARL. The disadvantage of scheme two is that the GRE egress QoS and ACL features are sacrificed.

Scheme one is the default setting in a Cat6k switch. A command is provided to globally change the default setting to scheme two if you want to have better performance. If you select to use scheme two, scheme two only applies to the VC created after the command is issued. If you want to have consistent hardware programming, existing VCs must be brought down and then brought back up.

VPLS over GRE Decapsulation

When packets arrive at the egress router, two recirculations are required. In the first pass, the GRE decapsulation is performed by the layer 3 (L3) module. After the GRE header is removed, the second pass performs EoMPLS decapsulation and the third pass performs L2 lookup and sends out the Ethernet frame to a proper outgoing interface.

VPLS over GRE MTU Requirements

In VPLS over GRE, the PEs are virtually connected by a GRE tunnel. At least one label (4 bytes) and a control word (4 bytes, optional) are added to each frame that is transported across the network. The transport frame is the Ethernet frame, the added 14 bytes are 6 bytes for each source and destination MAC address and 2 bytes for the Ethertype. Finally, 24 bytes are added for the GRE header and the outer IP header.

RFC preferences are to set the tunnel interface descriptor block (IDB) maximum transmission unit (MTU) to be the minimum MTU of all the egress interfaces that can be used by this tunnel to the remote tunnel endpoint. At the ingress router, the MTU size for the first pass should be at least 42 bytes less than the minimum MTU size (12 for MAC destination address [DA] and source address [SA], 2 for Ethertype, 4 for MPLS VC label stack, and 24 for GRE tunnel). 4 bytes for a control word and 4 bytes for an explicit null could be added for certain pseudowires.

EoMPLS over GRE

EoMPLS over GRE is conceptually the same as VPLS over GRE, but it is a peer-to-peer (P2P) service. The first pass decapsulates the GRE header, and the second pass performs EoMPLS decapsulation and sends the traffic to the proper interface.

How to Configure VPLS over GRE

Configuring VPLS over GRE

Perform these steps to configure VPLS over GRE on your Cisco network. If you would like to enable scheme two of VPLS over GRE, use the **platform vpls gre favor-performance** command at the end of these steps.

Note In scenarios where Generic Routing Encapsulation (GRE) is implemented over multiple Equal-Cost Multipath (ECMP) routes, and scheme two of VPLS over GRE is configured by using the **platform vpls gre favor-performance** command, the following should be considered. Scheme two of VPLS over GRE selects one of the ECMP routes as egress. Additional logic is executed on the supervisor engine or line card while selecting an ECMP route. Each supervisor engine or line card can select different ECMP routes as egress. For example, if GRE has two possible ECMP routes, the supervisor engine may select one route while the line card may select the other route as egress.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type/slot/port*
4. **ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]]
5. **exit**
6. **interface** *type/slot/port*
7. **ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]]
8. **mpls ip**
9. **tunnel source** {*ip-address* | *type/number*}
10. **tunnel destination** {*hostname* | *ip-address*}
11. **exit**
12. **interface** *type/slot/port*
13. **ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]]
14. **mpls ip**
15. **tunnel source** {*ip-address* | *type/number*}
16. **tunnel destination** {*hostname* | *ip-address* }
17. **exit**
18. **ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** |**track** *number*] [**tag** *tag*]
19. **ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]
20. **l2 vfi** *name* **manual**
21. **vpn id** *vpn-id*
22. **neighbor** *ip-address* [*vc-id*] {**encapsulation mpls** | **pw-class** *pw-class-name* } [*no-split-horizon*]
23. **neighbor** *ip-address* [*vc-id*] { **encapsulation mpls** | **pw-class** *pw-class-name* } [*no-split-horizon*]
24. **exit**
25. **interface** *type number*
26. **switchport mode access**
27. **switchport access vlan** *vlan-id*
28. **interface vlan** *vlan-id*
29. **xconnect vfi** *vfi-name*
30. **exit**
31. **platform vpls gre favor-performance**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enters privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type/slot/port* **Example:** `Device(config)# interface Loopback0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]] **Example:** `Device(config-if)# ip address 209.165.202.225 255.255.255.224`	Sets a primary or secondary IP address for an interface.
Step 5	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.
Step 6	**interface** *type/slot/port* **Example:** `Device(config)# interface Tunnel0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.
Step 7	**ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]] **Example:** `Device(config-if)# ip address 209.165.200.225 255.255.255.224`	Sets a primary or secondary IP address for an interface.
Step 8	**mpls ip** **Example:** `Device(config-if)# mpls ip`	Enables Multiprotocol Label Switching (MPLS) forwarding of IPv4 packets along normally routed paths for a particular interface.

Step 9	**tunnel source** {*ip-address* | *type/number*}
	Example:
	`Device(config-if)# tunnel source 209.165.201.1`
		Configures the tunnel source.

	Command or Action	Purpose

Step 10	**tunnel destination** {*hostname* | *ip-address*}
	Example:
	`Device(config-if)# tunnel destination 209.165.201.2`
		Configures the tunnel destination.

Step 11	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.
Step 12	**interface** *type/slot/port* **Example:** `Device(config)# interface Tunnel1`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.
Step 13	**ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]] **Example:** `Device(config-if)# ip address 209.165.201.3 255.255.255.224`	Sets a primary or secondary IP address for an interface.
Step 14	**mpls ip** **Example:** `Device(config-if)# mpls ip`	Enables Multiprotocol Label Switching (MPLS) forwarding of IPv4 packets along normally routed paths for a particular interface.

Step 15	**tunnel source** {*ip-address* | *type/number*}
	Example:
	`Device(config-if)# tunnel source 209.165.201.4`
		Configures the tunnel source.

Step 16	**tunnel destination** {*hostname* | *ip-address* }
	Example:
	`Device(config-if)# tunnel destination 209.165.201.5`
		Configures the tunnel destination.

Step 17	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.
Step 18	**ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** |**track** *number*] [**tag** *tag*] **Example:** `Device(config)# ip route 209.165.201.6 255.255.255.224 Tunnel0`	Establishes a static route.

	Command or Action	Purpose
Step 19	**ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*] **Example:** `Device(config)# ip route 209.165.201.7 255.255.255.255 Tunnel1`	Establishes another static route.
Step 20	**l2 vfi** *name* **manual** **Example:** `Device(config)# l2 vfi green manual`	Creates a Layer 2 virtual forwarding instance (VFI) and enters Layer 2 manual configuration mode.
Step 21	**vpn id** *vpn-id* **Example:** `Device(config-vfi)# vpn id 100`	Configures a VPN ID for a VPLS domain. The emulated VCs bound to this Layer 2 VRF use this VPM ID for signaling.
Step 22	**neighbor** *ip-address* [*vc-id*] {**encapsulation mpls** | **pw-class** *pw-class-name* } [*no-split-horizon*] **Example:** `Device(config-vfi)# neighbor 209.165.201.7 encapsulation mpls`	Specifies the router that should form a point-to-point Layer 2 virtual forwarding interface (VFI) connection.
Step 23	**neighbor** *ip-address* [*vc-id*] { **encapsulation mpls** | **pw-class** *pw-class-name* } [*no-split-horizon*] **Example:** `Device(config-vfi)# neighbor 209.165.201.6 encapsulation mpls`	Specifies the router that should form a point-to-point Layer 2 virtual forwarding interface (VFI) connection.
Step 24	**exit** **Example:** `Device(config-if)# exit`	Exits Layer 2 manual configuration mode.
Step 25	**interface** *type number* **Example:** `Device(config)# interface gigabitEthernet 5/23`	Selects an interface to configure and enters interface configuration mode.
Step 26	**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	Purpose
Step 27	**switchport access vlan** *vlan-id* **Example:** `Device(config-if)# switchport access vlan 100`	Sets the VLAN when the interface is in access mode and enters Layer 2 manual configuration mode.
Step 28	**interface vlan** *vlan-id* **Example:** `Device(config-vfi)# interface vlan 100`	Creates or accesses a dynamic switched virtual interface (SVI).
Step 29	**xconnect vfi** *vfi-name* **Example:** `Device(config-vfi)# xconnect vfi green`	Specifies a Layer 2 VFI that you are binding to the VLAN port.
Step 30	**exit** **Example:** `Device(config-vfi)# exit`	Exits Layer 2 manual configuration mode and returns to global configuration mode.
Step 31	**platform vpls gre favor-performance** **Example:** `Device(config)# platform vpls gre favor-performance`	Optional step to enable scheme 2 of VPLS over GRE.

Configuration Examples for VPLS over GRE

Example: Configuring VPLS over GRE

The following example enables scheme one of VPLS over GRE, which is the default. To enable scheme two, use the **platform vpls gre favor-performance** command after all these commands.

PE1

```
Device(config)# interface Loopback0
Device(config-if)# ip address 209.165.202.225 255.255.255.224

Device(config)# interface Tunnel0
Device(config-if)# ip address 209.165.200.225 255.255.255.224
Device(config-if)# mpls ip
Device(config-if)# tunnel source 209.165.201.1
Device(config-if)# tunnel destination 209.165.201.2

Device(config)# interface Tunnel1
Device(config-if)# ip address 209.165.201.3 255.255.255.224
Device(config-if)# mpls ip
Device(config-if)# tunnel source 209.165.201.4
Device(config-if)# tunnel destination 209.165.201.5

Device(config)# ip route 209.165.201.6 255.255.255.224 Tunnel0
Device(config)# ip route 209.165.201.7 255.255.255.224 Tunnel1


Device(config)# l2 vfi green manual
Device(config-vfi)# vpn id 100
Device(config-vfi)# neighbor 209.165.201.7 encapsulation mpls
Device(config-vfi)# neighbor 209.165.201.6 encapsulation mpls


Device(config)# int gigabitEthernet 5/23
Device(config-if)# switchport mode access
Device(config-if)# switchport access vlan 100


Device(config-vfi)# interface Vlan 100
Device(config-if)# xconnect vfi green
```

PE2

```
Device2(config-if)# interface Loopback0
Device2(config-if)# ip address 209.165.201.6 255.255.255.224

Device2(config-if)# interface Tunnel0
Device2(config-if)# ip address 209.165.201.8 255.255.255.224
Device2(config-if)# mpls ip
Device2(config-if)# tunnel source 209.165.201.2
Device2(config-if)# tunnel destination 209.165.201.1

Device2(config-if)# interface Tunnel1
Device2(config-if)# ip address 209.165.201.9 255.255.255.224
Device2(config-if)# mpls ip
Device2(config-if)# tunnel source 209.165.201.10
Device2(config-if)# tunnel destination 209.165.201.11

Device2(config)# ip route 209.165.202.224 255.255.255.224 Tunnel0
Device2(config)# ip route 209.165.201.7 255.255.255.255 Tunnel1

Device2(config)# l2 vfi green manual
Device2(config-vfi)# vpn id 100
Device2(config-vfi)# neighbor 209.165.202.225 encapsulation mpls
Device2(config-vfi)# neighbor 209.165.201.7 encapsulation mpls

Device2(config)# int gigabitEthernet 5/23
Device2(config-if)# switchport mode access
Device2(config-if)# switchport access vlan 100

Device2(config)# interface Vlan 100
Device2(config-if)# xconnect vfi green
```

Additional References for VPLS over GRE

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Related Topic	Document Title
MPLS commands	Multiprotocol Label Switching Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 4762	*Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for VPLS over GRE

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 1: Feature Information for Configuring Scheme Two of VPLS over GRE

Feature Name	Releases	Feature Information
VPLS over GRE	15.1(1)SY	The VPLS over GRE feature. The following commands were introduced or modified: **platform vpls gre favor-performance**