



BGP Support for the L2VPN Address Family

Last Updated: November 21, 2012

BGP support for the Layer 2 Virtual Private Network (L2VPN) address family introduces a BGP-based autodiscovery mechanism to distribute L2VPN endpoint provisioning information. BGP uses a separate L2VPN Routing Information Base (RIB) to store endpoint provisioning information, which is updated each time any Layer 2 virtual forwarding instance (VFI) is configured. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to set up a pseudowire mesh to support L2VPN-based services.

- [Finding Feature Information, page 1](#)
- [Prerequisites for BGP Support for the L2VPN Address Family, page 1](#)
- [Restrictions for BGP Support for the L2VPN Address Family, page 2](#)
- [Information About BGP Support for the L2VPN Address Family, page 2](#)
- [How to Configure BGP Support for the L2VPN Address Family, page 3](#)
- [Configuration Examples for BGP Support for the L2VPN Address Family, page 9](#)
- [Where to Go Next, page 12](#)
- [Additional References, page 12](#)
- [Feature Information for BGP Support for the L2VPN Address Family, page 13](#)

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.

Prerequisites for BGP Support for the L2VPN Address Family

The BGP Support for the L2VPN Address Family feature assumes prior knowledge of VPN, Virtual Private LAN Service (VPLS), and Multiprotocol Layer Switching (MPLS) technologies.



Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Restrictions for BGP Support for the L2VPN Address Family

- For route maps used within BGP, all commands related to prefix processing, tag processing, and automated tag processing are ignored when used in L2VPN address family configuration mode. All other route map commands are supported.
- BGP multipaths and confederations are not supported in the L2VPN address family.

Information About BGP Support for the L2VPN Address Family

- [L2VPN Address Family, page 2](#)
- [VPLS ID, page 3](#)

L2VPN Address Family

In Cisco IOS Release 12.2(33)SRB and later releases, support for the L2VPN address family is introduced. L2VPN is defined as a secure network that operates inside an unsecured network by using an encryption technology such as IP security (IPsec) or generic routing encapsulation (GRE). The L2VPN address family is configured in BGP routing configuration mode, and within the L2VPN address family the VPLS subsequent address family identifier (SAFI) is supported.

BGP support for the L2VPN address family introduces a BGP-based autodiscovery mechanism to distribute L2VPN endpoint provisioning information. BGP uses a separate L2VPN Routing Information Base (RIB) to store endpoint provisioning information, which is updated each time any Layer 2 VFI is configured. Prefix and path information is stored in the L2VPN database, allowing BGP to make best-path decisions. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to set up a pseudowire mesh to support L2VPN-based services.

The BGP autodiscovery mechanism facilitates the setting up of L2VPN services, which are an integral part of the Cisco IOS Virtual Private LAN Service (VPLS) feature. VPLS enables flexibility in deploying services by connecting geographically dispersed sites as a large LAN over high-speed Ethernet in a robust and scalable IP MPLS network. For more details about VPLS, see the [VPLS Autodiscovery: BGP Based](#) feature.

In L2VPN address family, the following BGP commands are supported:

- **bgp nexthop**
- **bgp scan-time**
- **neighbor activate**
- **neighbor advertisement-interval**
- **neighbor allowas-in**
- **neighbor capability**
- **neighbor inherit**
- **neighbor maximum-prefix**
- **neighbor next-hop-self**
- **neighbor next-hop-unchanged**
- **neighbor peer-group**
- **neighbor remove-private-as**
- **neighbor route-map**

- **neighbor route-reflector-client**
- **neighbor send-community**
- **neighbor soft-reconfiguration**
- **neighbor soo**
- **neighbor weight**

**Note**

For route reflectors using L2VPNs, the **neighbor next-hop-self** and **neighbor next-hop-unchanged** commands are not supported.

For route maps used within BGP, all commands related to prefix processing, tag processing, and automated tag processing are ignored when used in L2VPN address family configuration. All other route map commands are supported.

BGP multipaths and confederations are not supported in the L2VPN address family.

VPLS ID

A VPLS ID is a BGP extended community value that identifies the VPLS domain. Manual configuration of this ID is optional because a default VPLS ID is generated using the BGP autonomous system number and the configured VPN ID. A VPLS ID can be composed in one of two ways: with an autonomous system number and an arbitrary number or with an IP address and an arbitrary number.

You can enter a VPLS ID in either of these formats:

- Enter a 16-bit autonomous system number, a colon, and a 32-bit number. For example:

45000:3

- Enter a 32-bit IP address, a colon, and a 16-bit number. For example:

192.168.10.15:1

How to Configure BGP Support for the L2VPN Address Family

- [Configuring VPLS Autodiscovery Using BGP and the L2VPN Address Family](#), page 3

Configuring VPLS Autodiscovery Using BGP and the L2VPN Address Family

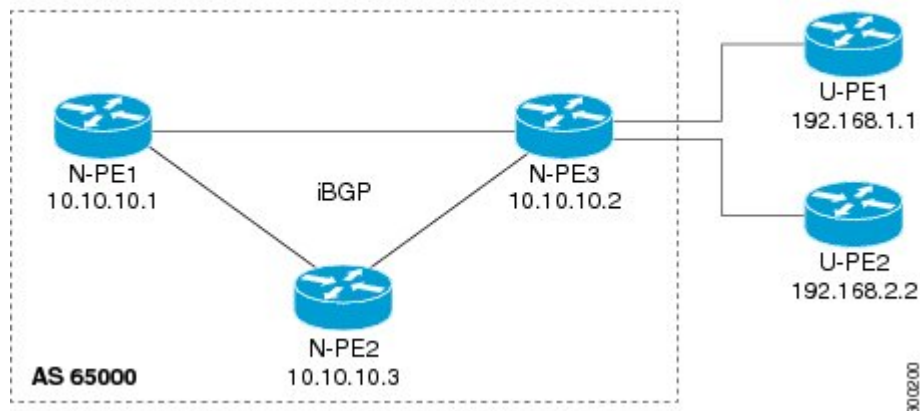
Perform this task to implement VPLS autodiscovery of each provider edge (PE) router that is a member of a specific VPLS. In Cisco IOS Release 12.2(33)SRB, the BGP L2VPN address family was introduced with a separate L2VPN RIB that contains endpoint provisioning information. BGP learns the endpoint provisioning information from the L2VPN database, which is updated each time any Layer 2 (L2) VFI is configured. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to set up a pseudowire mesh to support L2VPN-based services.

BGP-based VPLS autodiscovery eliminates the need to manually provision a VPLS neighbor. After a PE router configures itself to be a member of a particular VPLS, information needed to set up connections to remote routers in the same VPLS is distributed by a discovery process. When the discovery process is complete, each member of the VPLS will have the information needed to set up VPLS pseudowires to form the full mesh of pseudowires needed for the VPLS.

This task is configured at router N-PE3 in the figure below and must be repeated at routers N-PE1 and N-PE2 with the appropriate changes such as different IP addresses. For a full configuration of these routers, see “Configuring VPLS Autodiscovery Using BGP” and the “L2VPN Address Family Example”.

Figure 1 Network Diagram for BGP Autodiscovery Using the L2VPN Address Family

N-PE - Network-facing PE router
U-PE - User-facing PE router



In this task, the PE router N-PE3 in the figure above is configured with a Layer 2 router ID, a VPN ID, a VPLS ID, and is enabled to automatically discover other PE routers that are part of the same VPLS domain. A BGP session is created to activate BGP neighbors in the L2VPN address family. Finally, two optional **show** commands are entered to verify the steps in the task.

If a route reflector (RR) node is provisioned for a new virtual forwarding instance (VFI), BGP announces the entire current table from the L2VPN address family identifier (AFI) to the L2VPN xconnect database to make sure that the virtual circuits (VC) are active.

This task assumes that MPLS is configured with VPLS options. For more details, see the VPLS Autodiscovery: BGP Based feature.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2 router-id** *ip-address*
4. **l2 vfi** *vfi-name* **autodiscovery**
5. **vpn id** *vpn-id*
6. **vpls-id** *vpls-id*
7. **exit**
8. Repeat Step 4 through Step 6 to configure other L2 VFIs and associated VPN and VPLS IDs.
9. **router bgp** *autonomous-system-number*
10. **no bgp default ipv4-unicast**
11. **bgp log-neighbor-changes**
12. **bgp update-delay** *seconds*
13. **neighbor** { *ip-address* | *peer-group-name* } **remote-as** *autonomous-system-number*
14. **neighbor** { *ip-address* | *peer-group-name* } **update-source** *interface-type interface-number*
15. Repeat Step 13 and Step 14 to configure other BGP neighbors.
16. **address-family l2vpn** [**vpls**]
17. **neighbor** { *ip-address* | *peer-group-name* } **activate**
18. **neighbor** { *ip-address* | *peer-group-name* } **send-community**[**both** | **standard** | **extended**]
19. Repeat Step 17 and Step 18 to activate other BGP neighbors in L2VPN address family.
20. **end**
21. **show vfi**
22. **show ip bgp l2vpn vpls** { **all** | **rd** *vpn-rd* }

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>

Command or Action	Purpose
<p>Step 3 <code>l2 router-id ip-address</code></p> <p>Example:</p> <pre>Router(config)# l2 router-id 10.1.1.3</pre>	<p>Specifies a router ID (in IP address format) for the PE router to use with VPLS autodiscovery pseudowires.</p> <ul style="list-style-type: none"> In this example, the L2 router ID is defined as 10.1.1.3.
<p>Step 4 <code>l2 vfi vfi-name autodiscovery</code></p> <p>Example:</p> <pre>Router(config)# l2 vfi customerA autodiscovery</pre>	<p>Creates an L2 VFI, enables the VPLS PE router to automatically discover other PE routers that are part of the same VPLS domain, and enters L2 VFI autodiscovery configuration mode.</p> <ul style="list-style-type: none"> In this example, the L2 VFI named customerA is created.
<p>Step 5 <code>vpn id vpn-id</code></p> <p>Example:</p> <pre>Router(config-vfi)# vpn id 100</pre>	<p>Specifies a VPN ID.</p> <ul style="list-style-type: none"> Use the same VPN ID for the PE routers that belong to the same VPN. Make sure that the VPN ID is unique for each VPN in the service provider network. Use the <i>vpn-id</i> argument to specify a number in the range from 1 to 4294967295. In this example, a VPN ID of 100 is specified.
<p>Step 6 <code>vpls-id vpls-id</code></p> <p>Example:</p> <pre>Router(config-vfi)# vpls-id 65000:100</pre>	<p>(Optional) Specifies a VPLS ID.</p> <ul style="list-style-type: none"> The VPLS ID is an identifier that is used to identify the VPLS domain. This command is optional because a default VPLS ID is automatically generated using the BGP autonomous system number and the VPN ID configured for the VFI. Only one VPLS ID can be configured per VFI, and the same VPLS ID cannot be configured in multiple VFIs on the same router. In this example, a VPLS ID of 65000:100 is specified.
<p>Step 7 <code>exit</code></p> <p>Example:</p> <pre>Router(config-vfi)# exit</pre>	<p>Exits L2 VFI autodiscovery configuration mode and returns to global configuration mode.</p>
<p>Step 8 Repeat Step 4 through Step 6 to configure other L2 VFIs and associated VPN and VPLS IDs.</p>	<p>—</p>
<p>Step 9 <code>router bgp autonomous-system-number</code></p> <p>Example:</p> <pre>Router(config)# router bgp 65000</pre>	<p>Enters router configuration mode for the specified routing process.</p>

Command or Action	Purpose
<p>Step 10 no bgp default ipv4-unicast</p> <p>Example:</p> <pre>Router(config-router)# no bgp default ipv4-unicast</pre>	<p>Disables the IPv4 unicast address family for the BGP routing process.</p> <p>Note Routing information for the IPv4 unicast address family is advertised by default for each BGP routing session configured with the neighbor remote-as router configuration command unless you configure the no bgp default ipv4-unicast router configuration command before configuring the neighbor remote-as command. Existing neighbor configurations are not affected.</p>
<p>Step 11 bgp log-neighbor-changes</p> <p>Example:</p> <pre>Router(config-router)# bgp log- neighbor-changes</pre>	<p>Enables logging of BGP neighbor resets.</p>
<p>Step 12 bgp update-delay <i>seconds</i></p> <p>Example:</p> <pre>Router(config-router)# bgp update- delay 1</pre>	<p>Sets the maximum initial delay period before a BGP-speaking networking device sends its first updates.</p> <ul style="list-style-type: none"> Use the <i>seconds</i> argument to set the delay period.
<p>Step 13 neighbor {<i>ip-address</i> <i>peer-group-name</i>} remote-as <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Router(config-router)# neighbor 10.10.10.1 remote-as 65000</pre>	<p>Adds the IP address or peer group name of the neighbor in the specified autonomous system to the IPv4 multiprotocol BGP neighbor table of the local router.</p> <ul style="list-style-type: none"> If the <i>autonomous-system-number</i> argument matches the autonomous system number specified in the router bgp command, the neighbor is an internal neighbor. If the <i>autonomous-system-number</i> argument does not match the autonomous system number specified in the router bgp command, the neighbor is an external neighbor. In this example, the neighbor at 10.10.10.1 is an internal BGP neighbor.
<p>Step 14 neighbor {<i>ip-address</i> <i>peer-group-name</i>} update-source <i>interface-type interface-number</i></p> <p>Example:</p> <pre>Router(config-router)# neighbor 10.10.10.1 update-source loopback 1</pre>	<p>(Optional) Configures a router to select a specific source or interface to receive routing table updates.</p> <ul style="list-style-type: none"> This example uses a loopback interface. The advantage to this configuration is that the loopback interface is not as susceptible to the effects of a flapping interface.
<p>Step 15 Repeat Step 13 and Step 14 to configure other BGP neighbors.</p>	<p>--</p>

Command or Action	Purpose
<p>Step 16 <code>address-family l2vpn [vpls]</code></p> <p>Example:</p> <pre>Router(config-router)# address-family l2vpn vpls</pre>	<p>Specifies the L2VPN address family and enters address family configuration mode.</p> <ul style="list-style-type: none"> The optional vpls keyword specifies that VPLS endpoint provisioning information is to be distributed to BGP peers. In this example, an L2VPN VPLS address family session is created.
<p>Step 17 <code>neighbor {ip-address peer-group-name} activate</code></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 10.10.10.1 activate</pre>	<p>Enables the neighbor to exchange information for the L2VPN VPLS address family with the local router.</p> <p>Note If you have configured a BGP peer group as a neighbor, you do not use this step. BGP peer groups are activated when a BGP parameter is configured. For example, the neighbor send-community command in the next step will automatically activate a peer group.</p>
<p>Step 18 <code>neighbor {ip-address peer-group-name} send-community[both standard extended]</code></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 10.10.10.1 send-community extended</pre>	<p>Specifies that a communities attribute should be sent to a BGP neighbor.</p> <ul style="list-style-type: none"> In this example, an extended communities attribute is sent to the neighbor at 10.10.10.1.
<p>Step 19 Repeat Step 17 and Step 18 to activate other BGP neighbors in L2VPN address family.</p>	<p>--</p>
<p>Step 20 <code>end</code></p> <p>Example:</p> <pre>Router(config-router-af)# end</pre>	<p>Exits address family configuration mode and returns to privileged EXEC mode.</p>
<p>Step 21 <code>show vfi</code></p> <p>Example:</p> <pre>Router# show vfi</pre>	<p>(Optional) Displays information about the configured VFI instances.</p>
<p>Step 22 <code>show ip bgp l2vpn vpls {all rd vpn-rd}</code></p> <p>Example:</p> <pre>Router# show ip bgp l2vpn vpls all</pre>	<p>(Optional) Displays information about the L2 VPN VPLS address family.</p>

- [Examples, page 9](#)
- [What to Do Next, page 9](#)

Examples

The following is sample output from the **show vfi** command that shows two VFIs, CustomerA and CustomerB, with their associated VPN and VPLS IDs:

```
Router# show vfi

Legend: RT=Route-target, S=Split-horizon, Y=Yes, N=No
VFI name: customerA, state: down, type: multipoint
VPN ID: 100, VPLS-ID: 65000:100
RD: 65000:100, RT: 65000:100
Local attachment circuits:
Neighbors connected via pseudowires:
Peer Address      VC ID      Discovered Router ID  S
10.10.10.1        100        10.10.10.99           Y
VFI name: customerB, state: down, type: multipoint
VPN ID: 200, VPLS-ID: 65000:200
RD: 65000:200, RT: 65000:200
Local attachment circuits:
Neighbors connected via pseudowires:
Peer Address      VC ID      Discovered Router ID  S
10.10.10.3        200        10.10.10.98           Y
```

The following is sample output from the **show ip bgp l2vpn vpls all** command that shows two VFIs identified by their VPN route distinguisher:

```
Router# show ip bgp l2vpn vpls all

BGP table version is 5, local router ID is 10.10.10.2
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          Metric LocPrf Weight Path
Route Distinguisher: 65000:100
*> 65000:100:10.10.10.1/96
                   0.0.0.0                                32768 ?
*>i65000:100:192.168.1.1/96
                   10.10.10.2                                0 100 0 ?
Route Distinguisher: 65000:200
*> 65000:200:10.10.10.3/96
                   0.0.0.0                                32768 ?
*>i65000:200:192.168.2.2/96
                   10.10.10.2                                0 100 0 ?
```

What to Do Next

To configure more VPLS features, see the main VPLS documentation in the “VPLS Autodiscovery: BGP Based” module in the *MPLS Layer 2 VPNs Configuration Guide*.

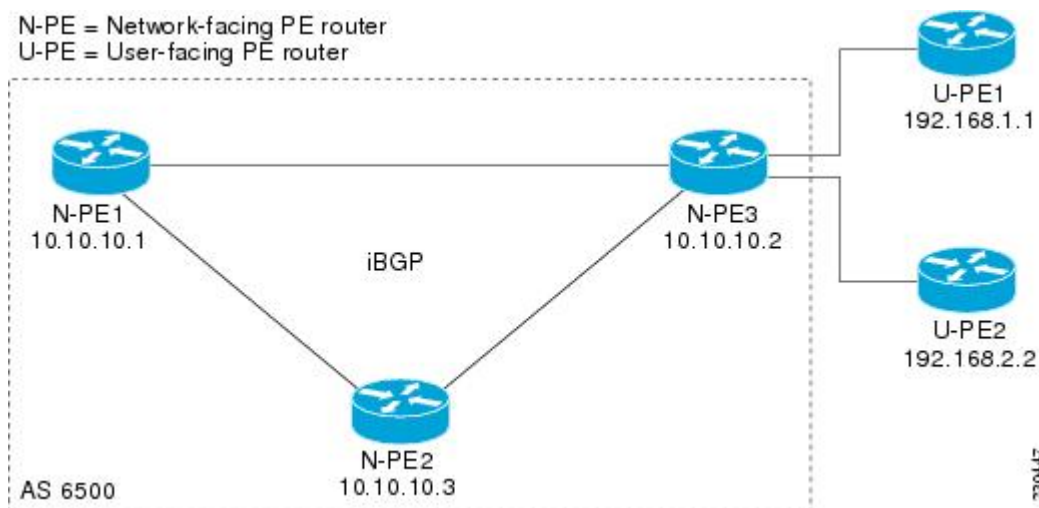
Configuration Examples for BGP Support for the L2VPN Address Family

- [Example: Configuring VPLS Autodiscovery Using BGP and the L2VPN Address Family, page 10](#)

Example: Configuring VPLS Autodiscovery Using BGP and the L2VPN Address Family

In this configuration example, all the routers in autonomous system 65000 in the figure below are configured to provide BGP support for the L2VPN address family. VPLS autodiscovery is enabled and L2 VFI and VPN IDs are configured. BGP neighbors are configured and activated in L2VPN address family to ensure that the VPLS endpoint provisioning information is saved to a separate L2VPN RIB and then distributed to the other BGP peers in BGP update messages. When the endpoint information is received by the BGP peers, a pseudowire mesh is set up to support L2VPN-based services.

Figure 2 Network Diagram for VPLS Autodiscovery Using BGP and the L2VPN Address Family



Router N-PE1

```
ip subnet-zero
ip cef
no ip dhcp use vrf connected
!
no mpls traffic-eng auto-bw timers frequency 0
mpls label range 1000 2000
mpls label protocol ldp
l2 router-id 10.1.1.1
l2 vfi auto autodiscovery
  vpn id 100
!
pseudowire-class mpls
  encapsulation mpls
!
interface Loopback1
  ip address 10.1.1.1 255.255.255.255
!
interface Ethernet0/0
  description Backbone interface
  ip address 10.0.0.1 255.255.255.0
  mpls ip
!
router ospf 1
  log-adjacency-changes
  network 10.10.1.0 0.0.0.255 area 0
  network 192.168.0.0 0.0.0.255 area 0
!
```

```

router bgp 65000
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  bgp update-delay 1
  neighbor 10.10.10.2 remote-as 65000
  neighbor 10.10.10.2 update-source Loopback 1
  neighbor 10.10.10.3 remote-as 65000
  neighbor 10.10.10.3 update-source Loopback 1
  !
  address-family l2vpn vpls
  neighbor 10.10.10.2 activate
  neighbor 10.10.10.2 send-community extended
  neighbor 10.10.10.3 activate
  neighbor 10.10.10.3 send-community extended
  exit-address-family
  !
ip classless

```

Router N-PE2

```

ip subnet-zero
ip cef
no ip dhcp use vrf connected
!
no mpls traffic-eng auto-bw timers frequency 0
mpls label range 2000 3000
mpls label protocol ldp
l2 router-id 10.1.1.2
l2 vfi auto autodiscovery
  vpn id 100
  !
pseudowire-class mpls
  encapsulation mpls
  !
interface Loopback1
  ip address 10.1.1.2 255.255.255.255
  !
interface Ethernet0/0
  description Backbone interface
  ip address 10.0.0.2 255.255.255.0
  mpls ip
  !
router ospf 1
  log-adjacency-changes
  network 10.10.1.0 0.0.0.255 area 0
  network 192.168.0.0 0.0.0.255 area 0
  !
router bgp 65000
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  bgp update-delay 1
  neighbor 10.10.10.1 remote-as 65000
  neighbor 10.10.10.1 update-source Loopback 1
  neighbor 10.10.10.3 remote-as 65000
  neighbor 10.10.10.3 update-source Loopback 1
  !
  address-family l2vpn vpls
  neighbor 10.10.10.1 activate
  neighbor 10.10.10.1 send-community extended
  neighbor 10.10.10.3 activate
  neighbor 10.10.10.3 send-community extended
  exit-address-family
  !
ip classless

```

Router N-PE3

```

ip subnet-zero
ip cef
no ip dhcp use vrf connected

```

```

!
no mpls traffic-eng auto-bw timers frequency 0
mpls label range 2000 3000
mpls label protocol ldp
l2 router-id 10.1.1.3
l2 vfi auto autodiscovery
  vpn id 100
!
pseudowire-class mpls
  encapsulation mpls
!
interface Loopback1
  ip address 10.1.1.3 255.255.255.255
!
interface Ethernet0/0
  description Backbone interface
  ip address 10.0.0.3 255.255.255.0
  mpls ip
!
router ospf 1
  log-adjacency-changes
  network 10.10.1.0 0.0.0.255 area 0
  network 192.168.0.0 0.0.0.255 area 0
!
router bgp 65000
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  bgp update-delay 1
  neighbor 10.10.10.1 remote-as 65000
  neighbor 10.10.10.1 update-source Loopback 1
  neighbor 10.10.10.2 remote-as 65000
  neighbor 10.10.10.2 update-source Loopback 1
!
  address-family l2vpn vpls
  neighbor 10.10.10.1 activate
  neighbor 10.10.10.1 send-community extended
  neighbor 10.10.10.2 activate
  neighbor 10.10.10.2 send-community extended
  exit-address-family
!
ip classless

```

Where to Go Next

For more details about configuring VPLS autodiscovery, see the “VPLS Autodiscovery: BGP Based” module in the *MPLS Layer 2 VPNS Configuration Guide*.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
BGP commands	Cisco IOS IP Routing: BGP Command Reference
VPLS Autodiscovery BGP Based	“VPLS Autodiscovery BGP Based” module in the <i>MPLS Layer 2 VPNS Configuration Guide</i>

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
—	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

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 BGP Support for the L2VPN Address Family

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 BGP Support for the L2VPN Address Family**

Feature Name	Releases	Feature Information
BGP Support for the L2VPN Address Family	12.2(33)SRB 15.1(1)SG Cisco IOS XE 3.3SG	<p>BGP support for the L2VPN address family introduces a BGP-based autodiscovery mechanism to distribute L2VPN endpoint provisioning information. BGP uses a separate L2VPN RIB to store endpoint provisioning information, which is updated each time any Layer 2 VFI is configured. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to set up a pseudowire mesh to support L2VPN-based services.</p> <p>In Cisco IOS Release 12.2(33)SRB, this feature was introduced on the Cisco 7600 platform.</p> <p>The following commands were introduced or modified by this feature: address-family l2vpn, clear ip bgp l2vpn, show ip bgp l2vpn.</p>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2012 Cisco Systems, Inc. All rights reserved.