MPLS/VPN Configuration on IOS Platforms

Overview

This module covers MPLS/VPN configuration on Cisco IOS platforms.

Upon completion of this module, the learner will be able to perform the following tasks:

- Configure Virtual Routing and Forwarding tables
- Configure Multi-protocol BGP in MPLS/VPN backbone
- Configure PE-CE routing protocols
- Configure advanced MPLS/VPN features
- Monitor MPLS/VPN operations
- Troubleshoot MPLS/VPN implementation

Outline

The module contains the following lessons:

- MPLS/VPN Mechanisms in Cisco IOS
- Configuring Virtual Routing and Forwarding Tables
- Configuring a Multi-Protocol BGP Session between the PE Routers
- Configuring Routing Protocols between PE and CE Routers
- Monitoring an MPLS/VPN Operation
- Troubleshooting MPLS/VPN
- Advanced VRF Import/Export Features
- Advanced PE-CE BGP Configuration

MPLS/VPN Mechanisms in Cisco IOS

Overview

This lesson describes mechanisms that are used to implement MPLS VPN in Cisco IOS.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Describe the concept of Virtual Routing and Forwarding tables
- Describe the concept of routing protocol contexts
- Describe the interaction between PE-CE routing protocols, backbone MP-BGP, and virtual routing and forwarding tables

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

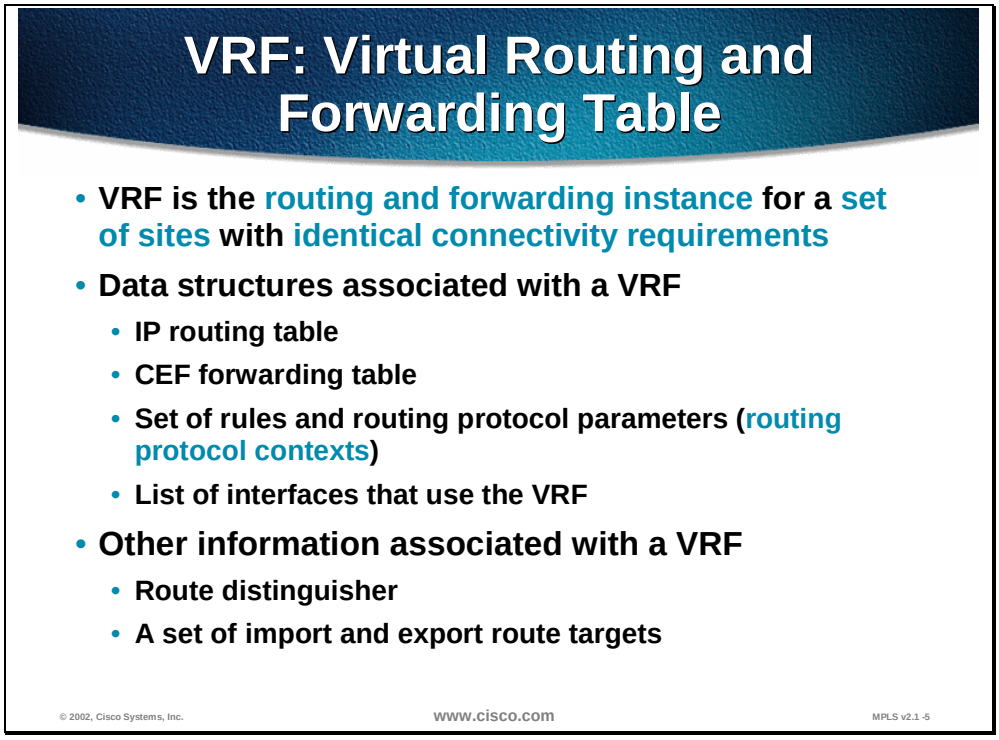
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- Virtual Routing and Forwarding Table
- Routing Protocol Contexts and Instances
- Interaction Between Routing Protocols and VRFs
- Summary
- Lesson Review

Virtual Routing and Forwarding Table



The major data structure associated with MPLS/VPN implementation in Cisco IOS is the **Virtual Routing** and **Forwarding** table (VRF). This data structure encompasses an IP routing table, identical in its function to the global IP routing table in IOS, a Cisco Express Forwarding (CEF) forwarding table, identical in its function to the global CEF forwarding table (Forwarding Information Base or FIB) and specifications for routing protocols running inside the VRF.

A VRF is thus a routing and forwarding instance that can be used for a single VPN site or for many sites connected to the same PE router **as long as these sites share exactly the same connectivity requirements**.

Other MPLS/VPN attributes associated with a VRF are:

- The route distinguisher which is prepended to all routes exported from the VRF into the global VPNv4 BGP table
- A set of export route targets which are attached to any route exported from the VRF
- A set of import route targets, which are used to select VPNv4 routes that are to be imported into the VRF

Practice

Q1) Which data structures are associated with a VRF? (Select all that apply)

A) LIB table.

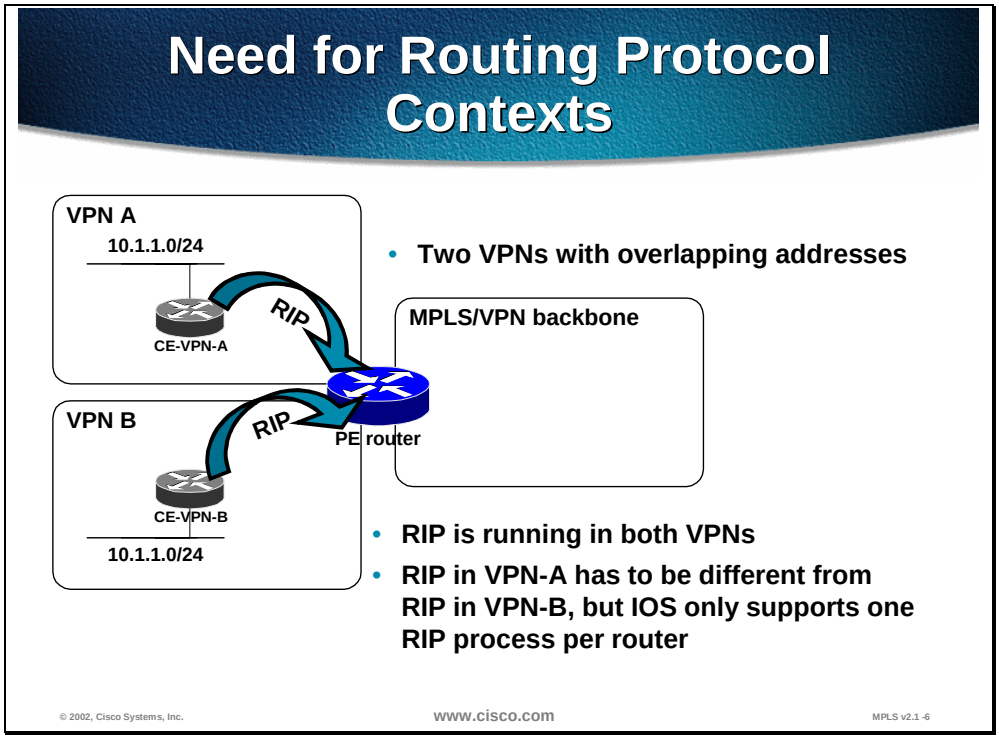
B) FIB table.

C) Routing table.

D) BGP table.

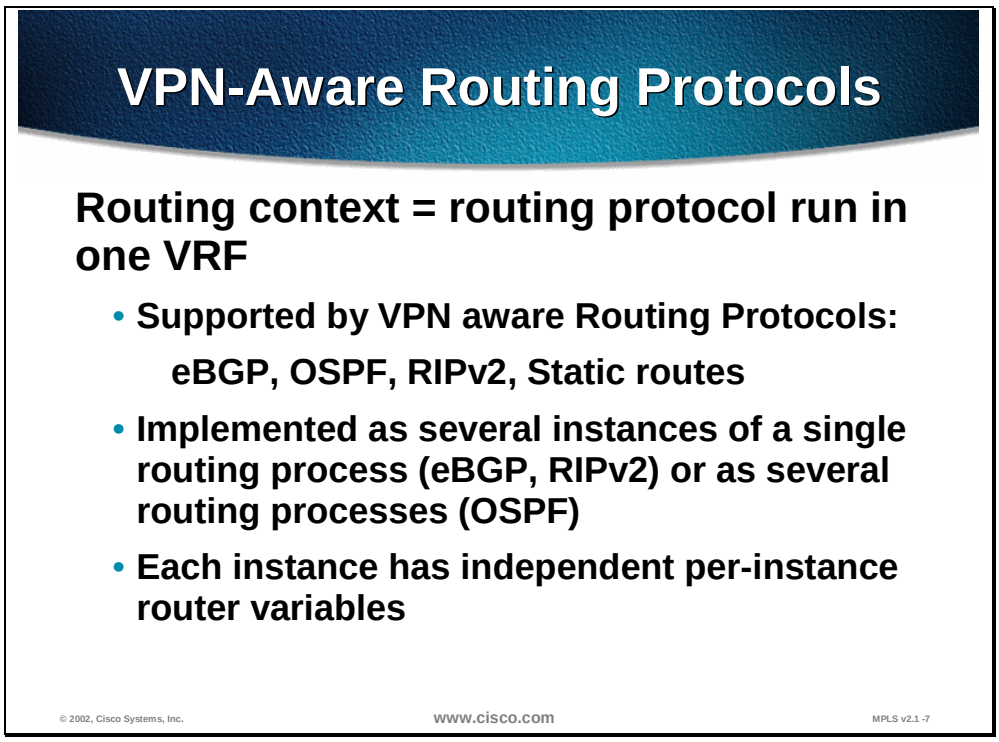
E) LFIB table.

Routing Protocol Contexts and Instances



Traditional Cisco IOS can support a number of different routing protocols; in some cases even several completely isolated copies of the same routing protocol (for example, several OSPF or EIGRP processes).

For several important routing protocols (for example, RIP or BGP), IOS supports only a single copy of the protocol running in the router. These protocols cannot be used directly between PE and CE routers in VPN environments, as each VPN (or, more precisely, each VRF) needs a separate, isolated copy of the routing protocol to prevent undesired route leakage between VPNs. Furthermore, VPNs can use overlapping IP address space (for example, each VPN could use subnets of network 10.0.0.0), which would also lead to routing confusions if all VPNs share the same copy of the routing protocol.

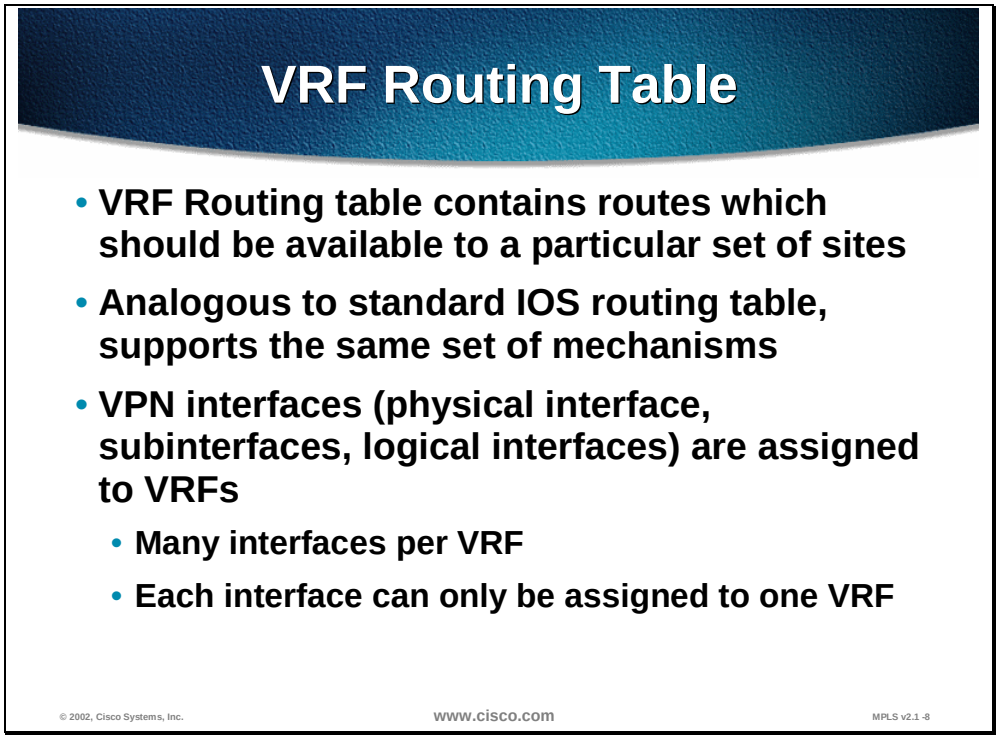


Routing Contexts were introduced in Cisco IOS to support the need for separate isolated copies of VPN routing protocols. The routing contexts can be implemented as separate routing processes (OSPF), similar to traditional IOS implementation, or as separate isolated *instances* of the same routing protocol.

If the routing contexts are implemented as instances of the same routing protocol, each instance contains its own independent routing-protocol parameters (for example, networks over which the routing protocol is run, timers, authentication parameters, passive interfaces, neighbors etc.), giving the network designer maximum flexibility in implementing routing protocols between PE and CE routers.

VPN aware routing protocols are protocols that support routing context. In IOS 12.1 external BGP, OSPF, RIP version 2 and static routes are VPN aware. For BGP and RIP, the routing context is implemented as several instances (address families) of a single routing process, while in OSPF the routing context is implemented by using several routing processes, each running completely in its own context.

Interaction between Routing Protocols and VRFs

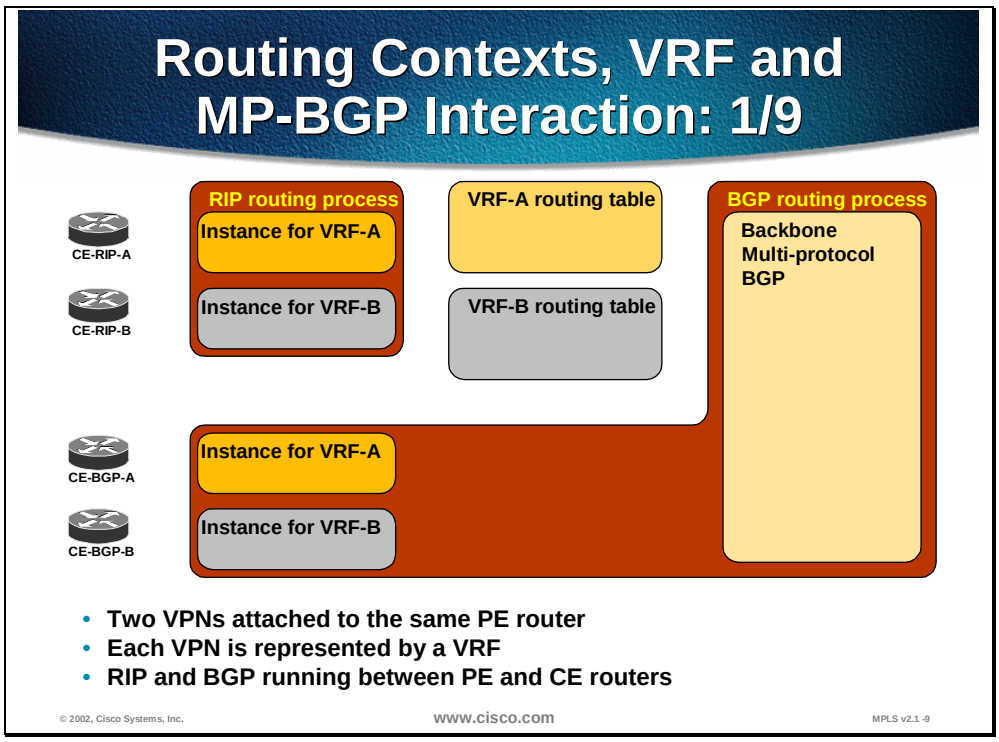


The routes received from VRF routing protocol instances or from dedicated VRF routing processes are inserted into the IP routing table contained within the VRF. This IP routing table supports exactly the same set of mechanisms as the standard IOS routing table, including filtering mechanisms (distribute lists or prefix lists) and inter-protocol route selection mechanisms (administrative distances).

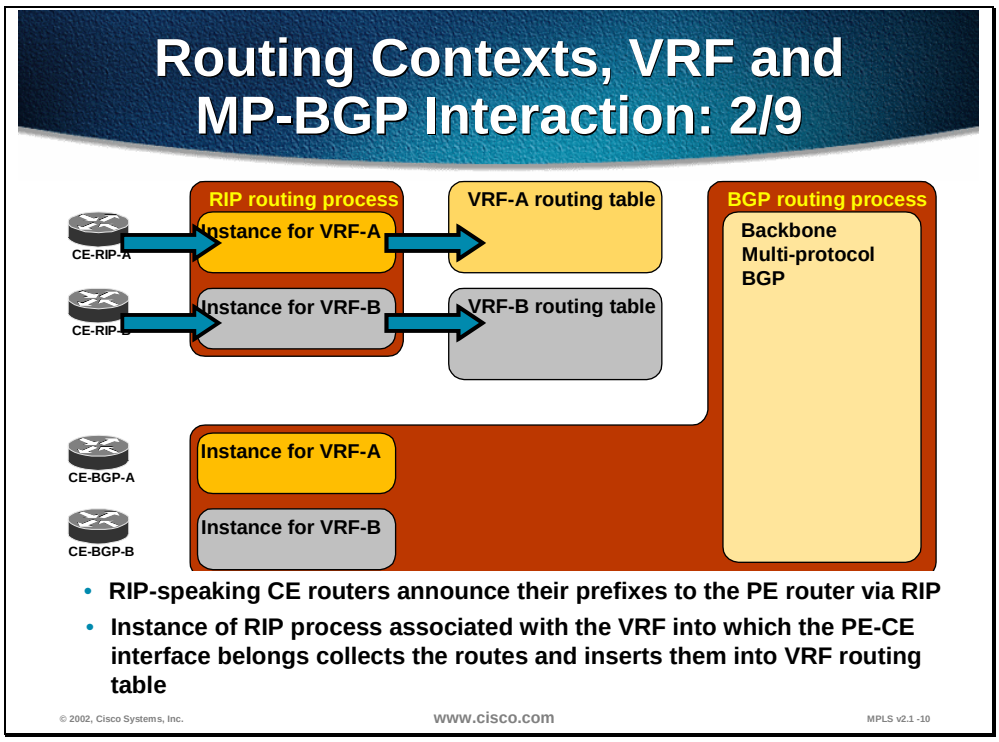
The per-VRF forwarding table (FIB) is built from the per-VRF routing table and is used to forward all the packets received through the interfaces associated with the VRF. Any interface can be associated with a VRF, be it physical interface, subinterface, or a logical interface, as long as it supports CEF switching.

Note The requirement to support CEF switching on inbound VRF interfaces prevents certain media or encapsulation types from being used for VPN connectivity. More notable examples in mainstream Cisco IOS 12.1 include dialer interfaces, ISDN interfaces, and Switched Multimegabit Data Service (SMDS) interfaces. Some restrictions are already lifted in IOS 12.1T releases, please refer to the release notes of the IOS release you're using for the details of interfaces and media types supporting CEF switching.

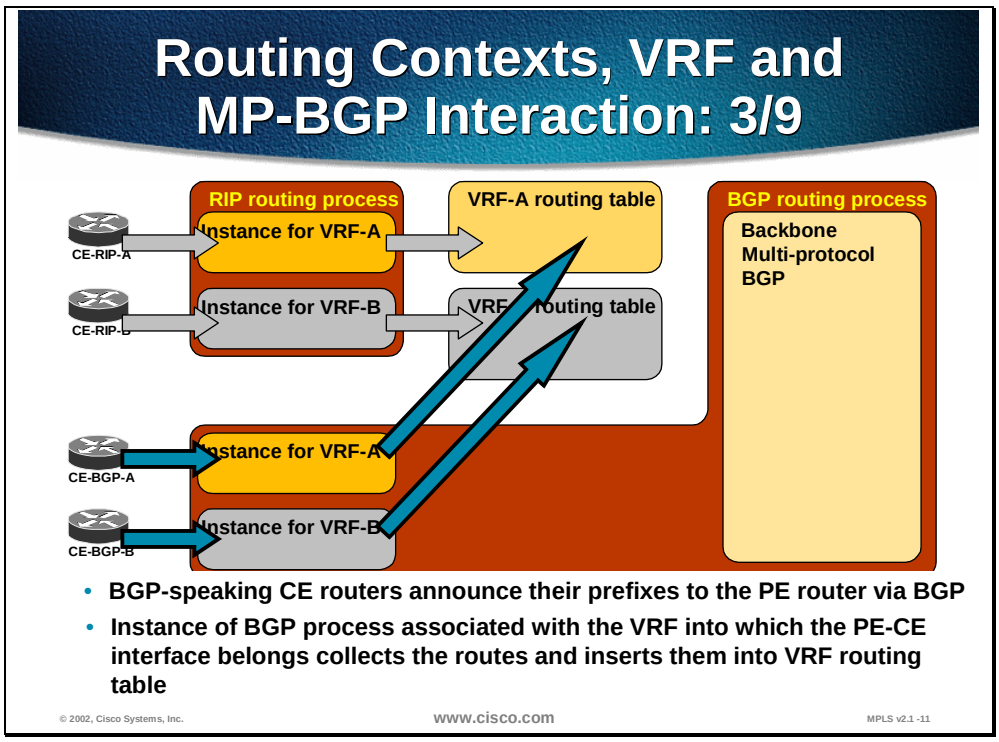
There is no limit to the number of interfaces associated with one VRF (the only limit is the number of interfaces supported by the router), however, each interface can be associated only with one VRF, because the router needs to uniquely identify the forwarding table to be used for packets received over an interface.



This and the following slides will illustrate the interactions between VRF instances of routing processes, VRF routing tables, and the global VPNv4 BGP routing process. A simple MPLS/VPN network will be used throughout the example. The network contains two VPN customers (called VPN-A and VPN-B). The customer sites are connected to a number of Provider Edge (PE) routers, but in the example we'll focus only on a single PE router, which contains two VRFs – one for each customer. Two sites of each customer are connected to the PE router, one site running BGP, the other site running RIP as the PE-CE routing protocol.

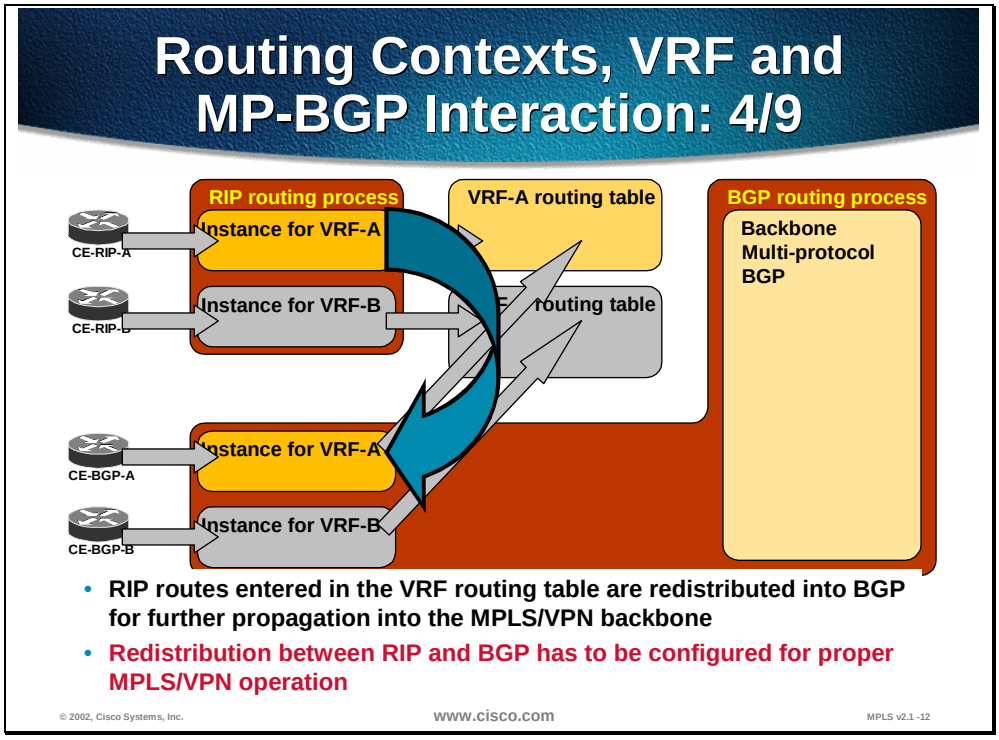


RIP-speaking CE routers announce their networks to the PE router. These updates are received by appropriate instances of RIP routing process (the correct instance is identified through the association of inbound PE interface to a VRF) and inserted into the per-VRF IP routing tables.



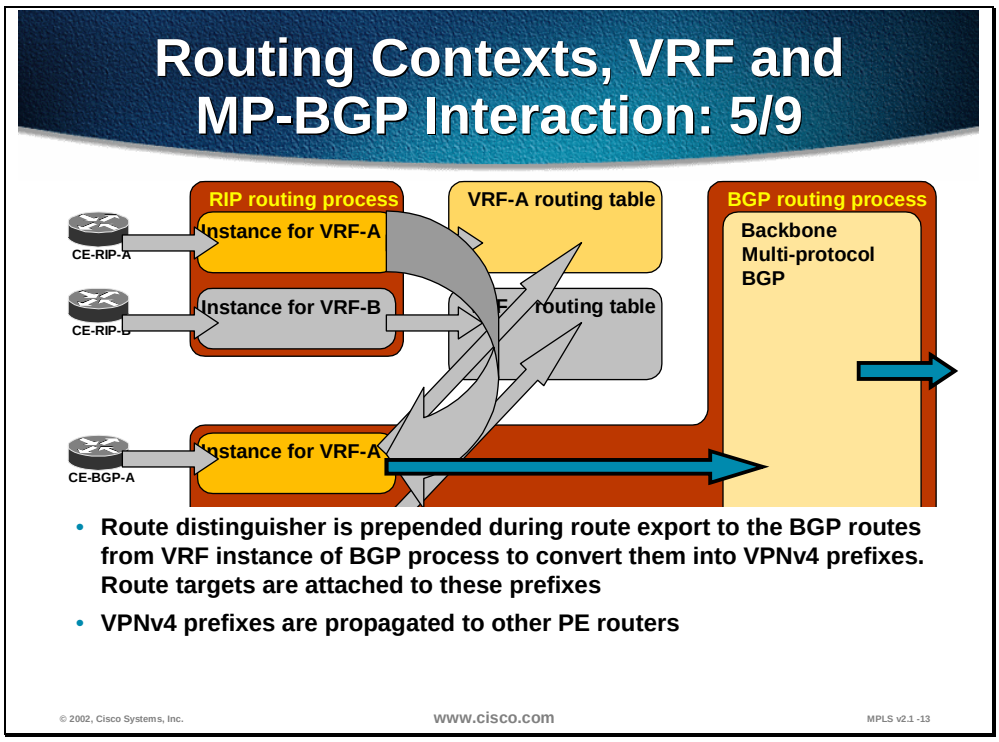
Similar to RIP-speaking routers, the BGP-speaking CE routers announce their networks via EBGP sessions to the PE router. The Customer Edge BGP neighbors of the PE router are associated with individual VRFs, which enable the various instances of the BGP routing process to put the received routing updates into the proper per-VRF routing table.

Should there be an overlap between an inbound RIP update and an inbound EBGP update, the standard route selection mechanism (administrative distance) is used in the per-VRF IP routing table and the EBGP route takes precedence over the RIP route, as the administrative distance of EBGP routes (20) is better than the administrative distance of RIP routes (120).



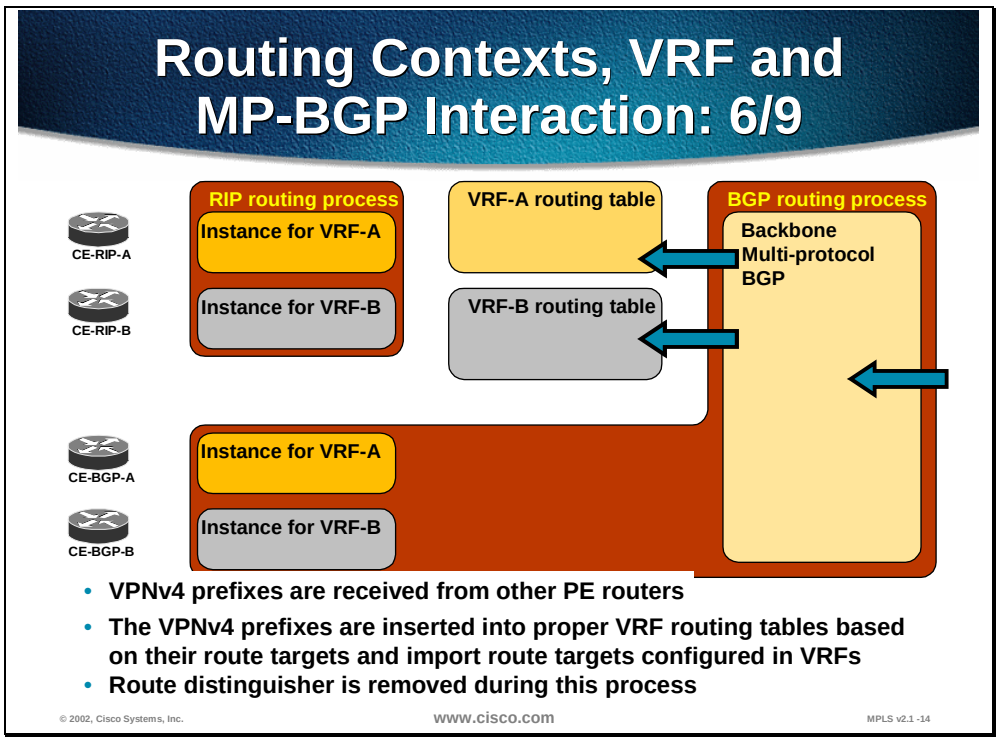
Multi-protocol BGP is used in the MPLS/VPN backbone to carry VPN routes (prefixed with route distinguisher) as 96-bit VPNv4 routes between the PE routers. The backbone BGP process looks exactly like a standard IBGP setup from the VRF's perspective. The per-VRF RIP routes therefore **have to be redistributed** into the per-VRF instance of the BGP process to allow them to be propagated through the backbone MP-BGP process to other PE routers.

Failure to redistribute non-BGP routes into per-VRF instance of BGP is one of the most common MPLS/VPN configuration failures.

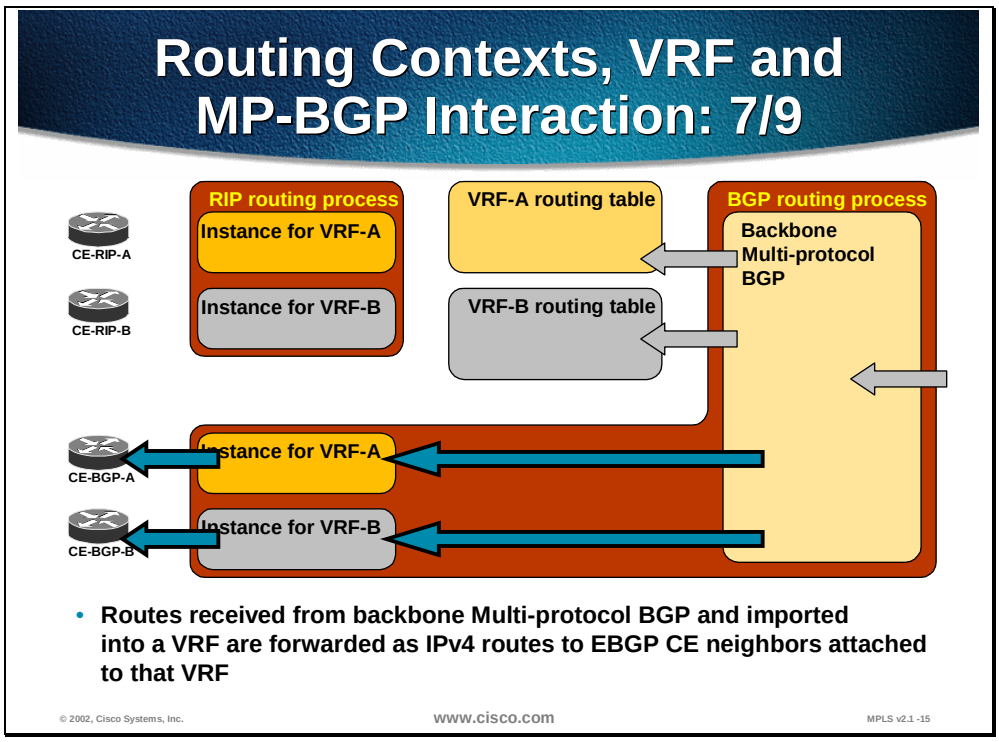


The RIP routes redistributed into the per-VRF instance of the BGP process as well as the BGP routes received from BGP-speaking CE routers are copied into the multi-protocol BGP table for further propagation to other PE routers. The IP prefixes are prepended with the Route Distinguisher (RD) and the set of route targets (extended BGP communities) configured as **export route targets** for the VRF is attached to the resulting VPNv4 route.

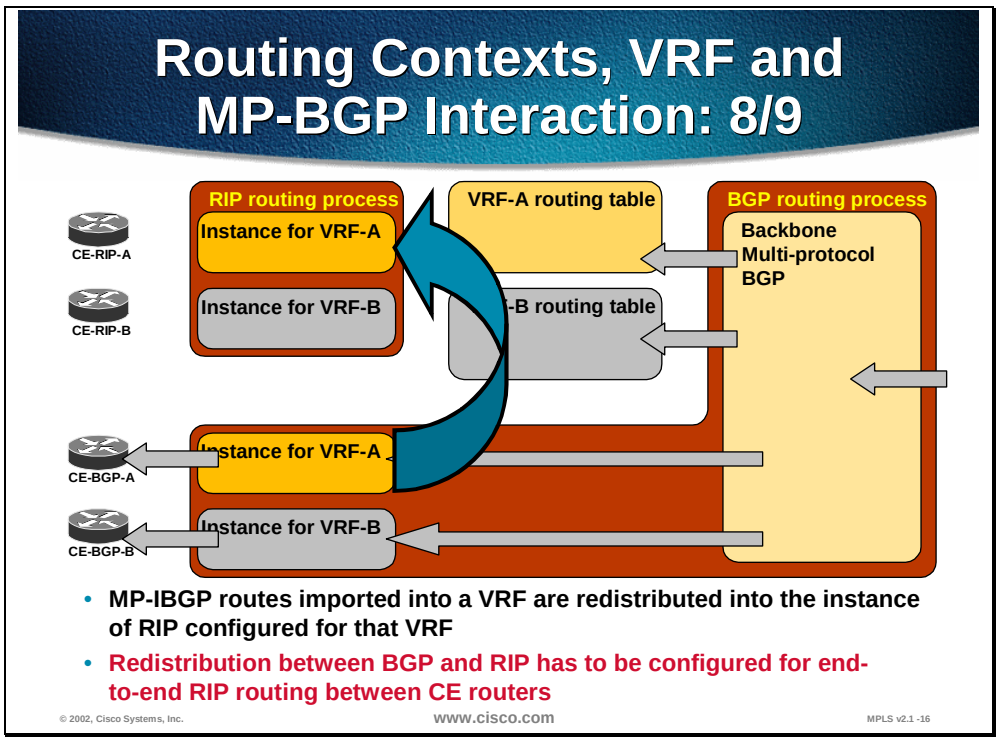
Note The difference between per-VRF BGP table and global MP-BGP table holding VPNv4 routes is displayed only to illustrate the steps in the route propagation process. In reality, there is no separate per-VRF BGP table in the Cisco IOS.



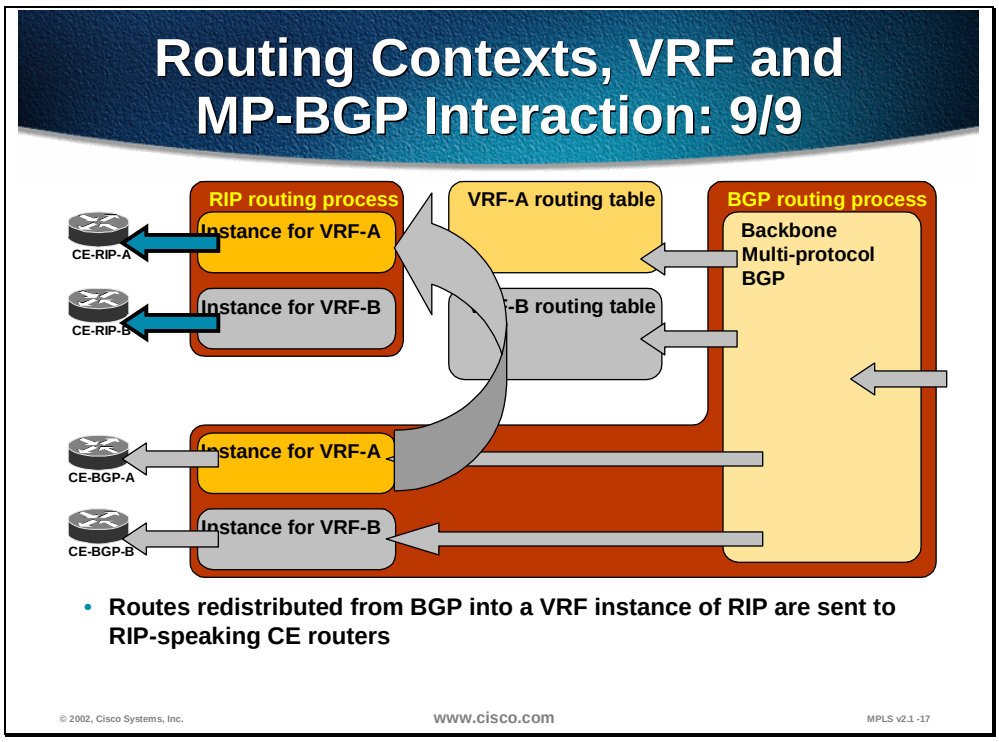
As the other PE routers start originating VPNv4 routes, the MP-BGP process in our PE router will receive these routes. The routes are filtered based on route target attributes attached to them and inserted into the proper per-VRF IP routing tables based on the **import route targets** configured for individual VRF. The route distinguisher that was prepended by the originating PE router is removed before the route is inserted into IPv4 the per-VRF IP routing table.



The MP-IBGP VPNv4 routes received from other PE routers and selected by the import route targets of a VRF are automatically propagated as 32-bit IPv4 routes to all BGP-speaking CE neighbors of the PE router.



The same routes, although they are inserted in the per-VRF IP routing table, are **not** propagated to RIP-speaking CE routers automatically. To propagate these routes (which appear as standard BGP routes in the per-VRF IP routing table) to the RIP-speaking CE routers, redistribution between per-VRF instance of BGP and per-VRF instance of RIP needs to be manually configured.



When the IBGP routes from the per-VRF IP routing table are successfully redistributed into the per-VRF instance of RIP process, the RIP process announces these routes to RIP-speaking CE routers, thus achieving transparent end-to-end connectivity between the CE routers.

Practice

Q1) How is a RIP route propagated into MP-BGP?

A) It is redistributed into the appropriate address family in BGP and exported.

B) Network statements are used to forward RIP routing information across MP-BGP.

C) MP-BGP automatically takes and advertises all entries in the VRF routing table.

D) Distribute lists are used to allow RIP routing information into MP-BGP.

Summary

After completing this section, you should be able to perform the following tasks:

- Describe the concept of Virtual Routing and Forwarding table
- Describe the concept of routing protocol contexts
- Describe the interaction between PE-CE routing protocols, backbone MP-BGP and virtual routing and forwarding tables

Next Steps

After completing this lesson, go to:

- Configuring Virtual Routing and Forwarding Table

Lesson Review

Instructions

Answer the following questions:

1. Which data structures are associated with a VRF?
2. How many interfaces can be associated with a VRF?
3. How many VRFs can be associated with an interface?
4. What is a routing protocol context?
5. How are routing protocol contexts implemented in RIP?
6. How are routing protocol contexts implemented in OSPF?
7. How is a RIP route propagated into MP-BGP?
8. When is a MP-BGP route inserted into a VRF?

Configuring Virtual Routing and Forwarding Table

Overview

This lesson describes the basic configuration tasks that are needed to configure VRF on an MPLS VPN PE router.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Create a Virtual Routing and Forwarding Table
- Specify Routing Distinguisher and Route Targets for the created VRF
- Associate interfaces with the VRF

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

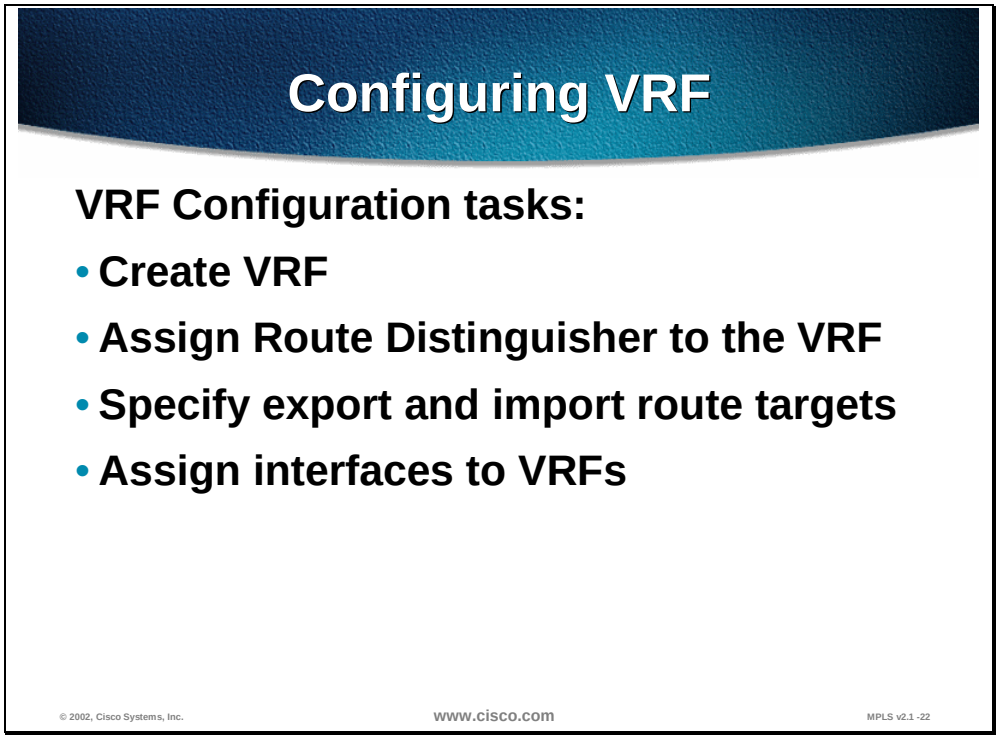
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- Creating a VRF
- Specifying Route Distinguisher
- Specifying Route Targets
- Associating Interfaces with the VRF
- Sample VPN Network
- Summary
- Lesson Review

Creating a VRF



Configuring VRF and starting deployment of an MPLS/VPN service for a customer consists of four mandatory steps:

- Creating a new VRF
- Assigning a unique route distinguisher to the VRF

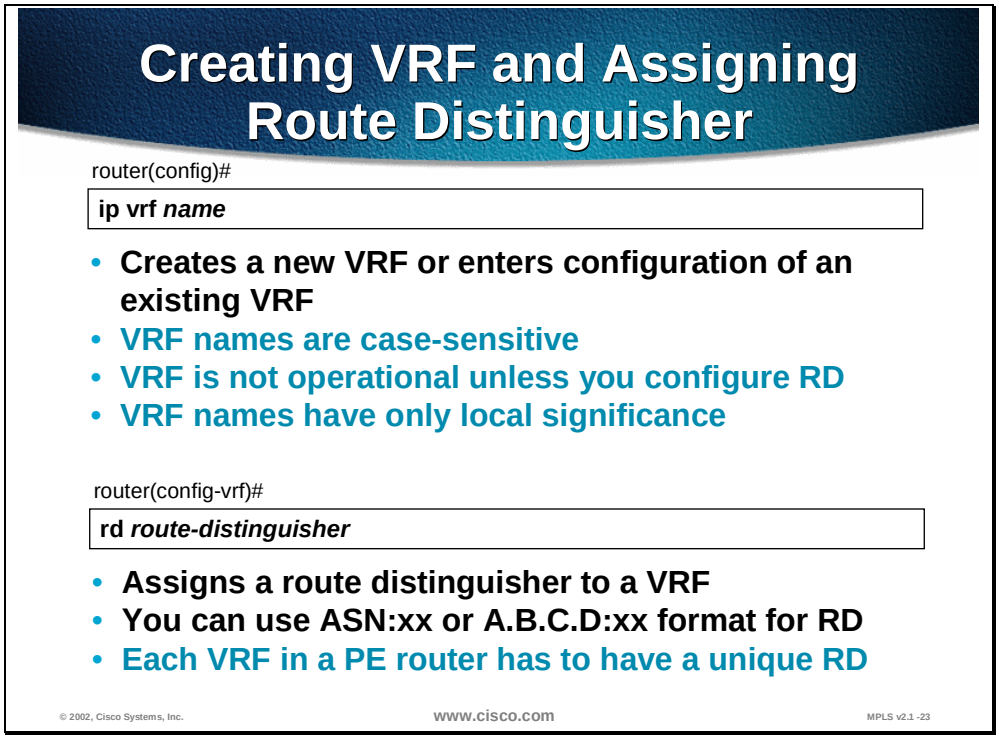
Note	A unique route distinguisher needs to be assigned to every VRF created in a PE router. The same route distinguisher **might** be used in multiple PE routers, based on customer connectivity requirements. The same route distinguisher **should** be used on all PE routers for simple VPN service. Please refer to Chapter #1 of the **SS_MPLS_VPN** lesson for more details on route distinguisher assignment for different VPN topologies.

- Specifying import and export route targets for a VRF

Note	Import and export route target should be equal to route distinguisher for simple VPN service. For other options, please refer to Chapter #1 of the **SS_MPLS_VPN** lesson.

- Assign the PE-CE interfaces to the new VRF.

Specifying Route Distinguisher



ip vrf

To configure a VRF routing table, use the **ip vrf** command in global configuration mode. To remove a VRF routing table, use the **no** form of this command.

ip vrf *vrf-name*
no ip vrf *vrf-name*

Syntax Description

vrf-name	Name assigned to a VRF.

Defaults

No VRFs are defined. No import or export lists are associated with a VRF. No route maps are associated with a VRF.

rd

To create routing and forwarding tables for a VRF, use the **rd** command in VRF submode.

rd *route-distinguisher*

Syntax Description

route-distinguisher	Adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix.

The route distinguisher can be specified in one of two formats:

- 16-bit AS-number followed by a 32-bit decimal number (AS:nn)
- 32-bit IP address followed by a 16-bit decimal number (A.B.C.D:nn)

Defaults

There is no default. An RD must be configured for a VRF to be functional.

Practice

Q1) Which VRF parameters must be specified for a VRF to become operational?

A) Import map.

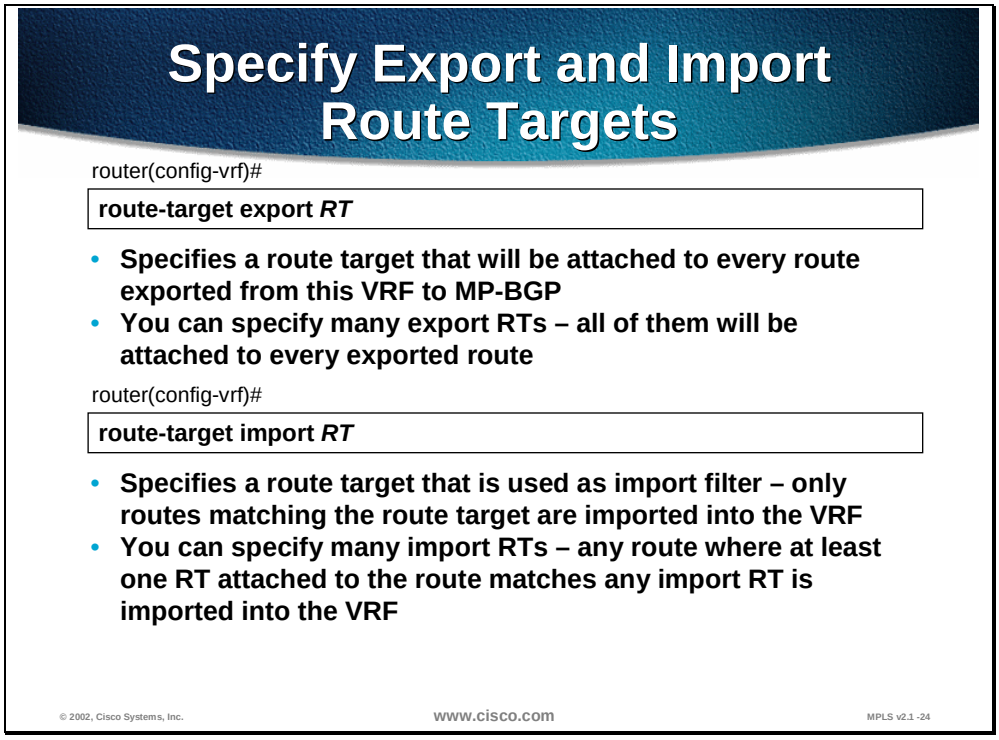
B) Route distinguisher.

C) Export map.

D) Route target.

E) VPN label.

Specifying Route Targets



route-target

To create a route-target extended community for a VRF, use the **route-target** command in VRF submode. To disable the configuration of a route-target community option, use the **no** form of this command.

route-target {import | export | both} *route-target-ext-community*
no route-target {import | export | both} *route-target-ext-community*

Syntax Description

import	Imports routing information from the target VPN extended community.
export	Exports routing information to the target VPN extended community.
both	Imports both import and export routing information to the target VPN extended community.
route-target-ext-community	Adds the route-target extended community attributes to the VRF's list of import, export, or both (import and export) route-target extended communities.

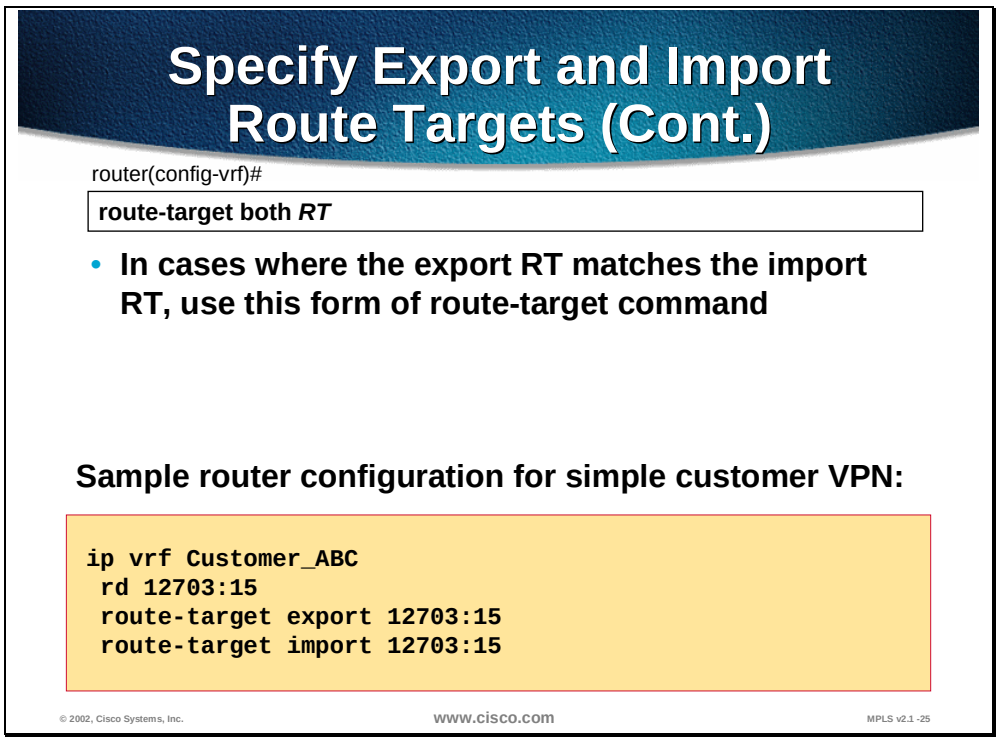
Similar to route distinguisher, the route targets can be specified in one of two formats:

- 16-bit AS-number followed by a 32-bit decimal number (AS:nn)

- 32-bit IP address followed by a 16-bit decimal number (A.B.C.D:nn)

Defaults

There are no defaults. A VRF has no route-target extended community attributes associated with it until specified by the **route-target** command.



Whenever a route target is both an import and an export route target for a VRF; you can use the **route-target both** command to simplify the configuration. For example, the two **route-target** configuration lines in the sample router configuration above could be reduced into a single command – **route-target both 12703:15**.

Practice

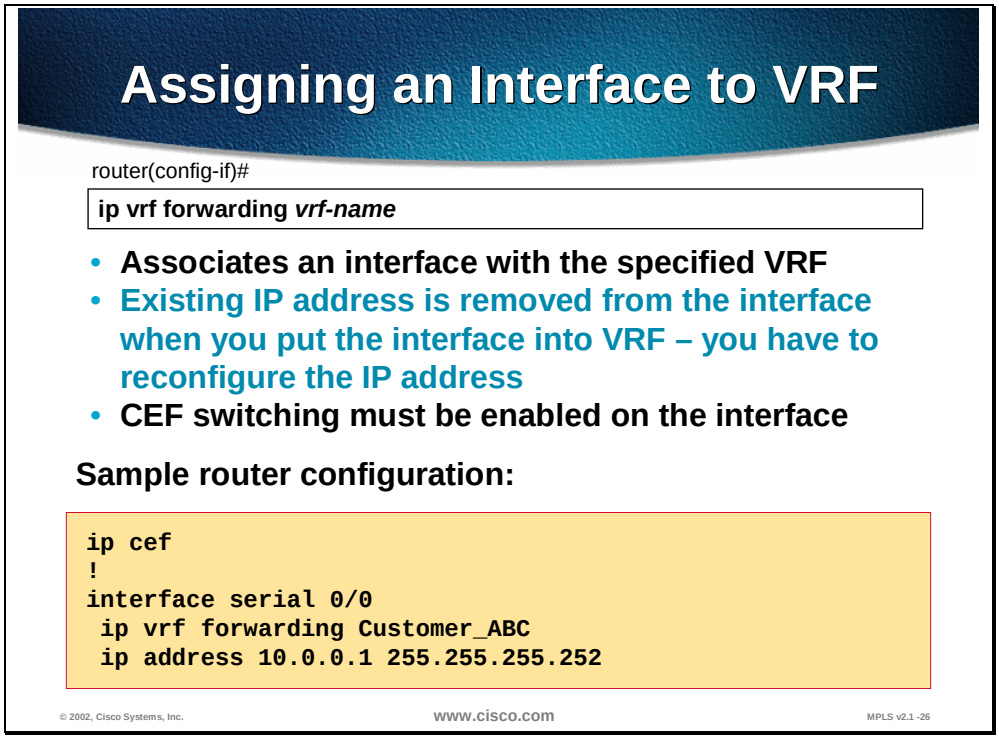
Q1) How many route targets can you configure on a VRF?

A) Exactly one route target has to be configured on a VRF (one import and one export)

B) Any number of route targets can be configured on a VRF

C) At most one route target has to be configured on a VRF (one import and one export)

Associating Interfaces with the VRF



ip vrf forwarding

To associate a VRF with an interface or subinterface, use the **ip vrf forwarding** command in interface configuration mode. To disassociate a VRF, use the **no** form of this command.

ip vrf forwarding *vrf-name*
no ip vrf forwarding *vrf-name*

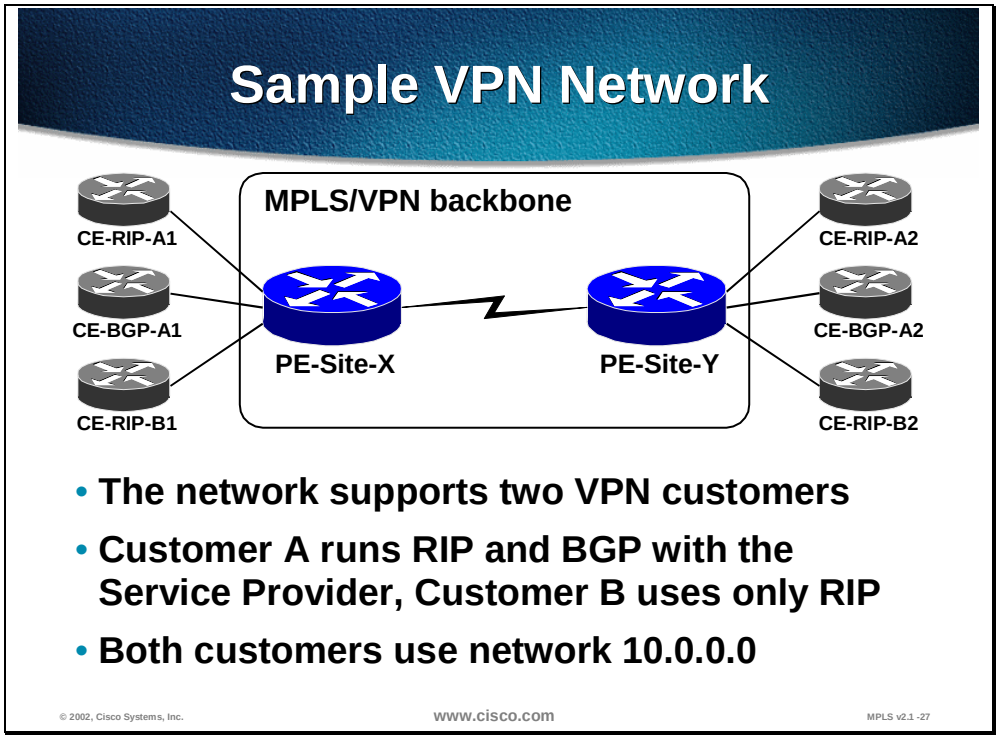
Syntax Description

vrf-name	Name assigned to a VRF.

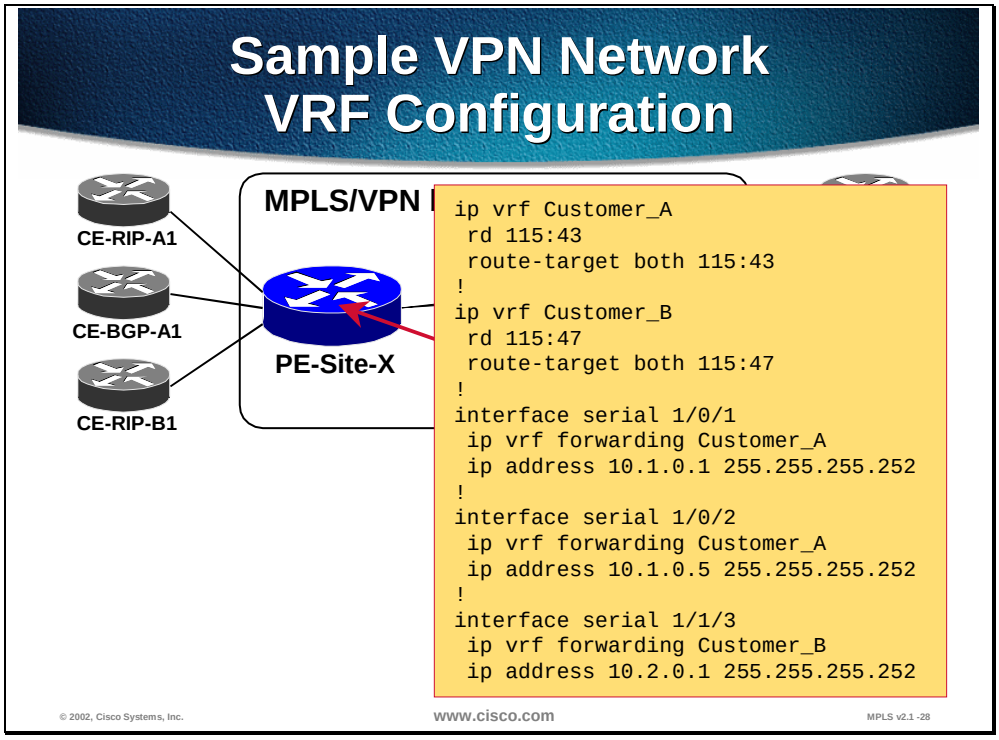
Defaults

The default for an interface is the global routing table.

Sample VPN Network



To illustrate the use of MPLS/VPN configuration commands, we'll configure the PE router in a sample network with two VPN customers. Customer A with four sites is using BGP and RIP as the PE-CE routing protocol and customer B (with two sites) is only using RIP. Both customers use private IP address space (subnets of network 10.0.0.0)



The configuration steps we can perform on the PE router so far include:

- **Configuring VRF** for Customer A and Customer B
- **Assigning route distinguishers and route targets to the VRFs.** As these customers only require simple VPN connectivity, one route distinguisher per customer is used on all PE routers in the MPLS/VPN backbone. To simplify the configuration and troubleshooting process, the route targets are made equal to route distinguishers.
- **Assigning PE-CE interfaces to individual VRFs**

Practice

Q1) What happens to existing interface configuration when you associate the interface with a VRF?

A) The ip address configuration of the interface is removed.

B) The interface enters the administratively down state.

C) All configuration on the interface is completly removed.

D) Nothing happens to the xisting interface configuration.

Summary

After completing this section, you should be able to perform the following tasks:

- Create a Virtual Routing and Forwarding Table
- Specify Route Distinguisher and Route Targets for the created VRF
- Associate interfaces with the VRF

Next Steps

After completing this lesson, go to:

- Configuring a Multi-Protocol BGP Session Between the PE Routers

Lesson Review

Instructions

Answer the following questions:

1. Which commands do you use to create a VRF?
2. Which VRF parameters must be specified for a VRF to become operational?
3. How do you associate an interface with a VRF?
4. What happens to an existing interface configuration when you associate the interface with a VRF?
5. How many formats can you use to specify RD and RT? What are these formats?
6. How many route targets can you configure on a VRF?
7. How many import route targets have to match a route for the route to be imported into the VRF?

Configuring a Multi-Protocol BGP Session Between the PE Routers

Overview

This lesson describes configuration tasks needed to establish MP-BGP connectivity between the PE routers.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Configure BGP address families
- Configure MP-BGP neighbors
- Configure inter-AS MP-BGP neighbors
- Configure additional mandatory parameters on MP-BGP neighbors
- Configure propagation of standard and extended BGP communities
- Selectively enable IPv4 and MP-BGP sections between BGP neighbors

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

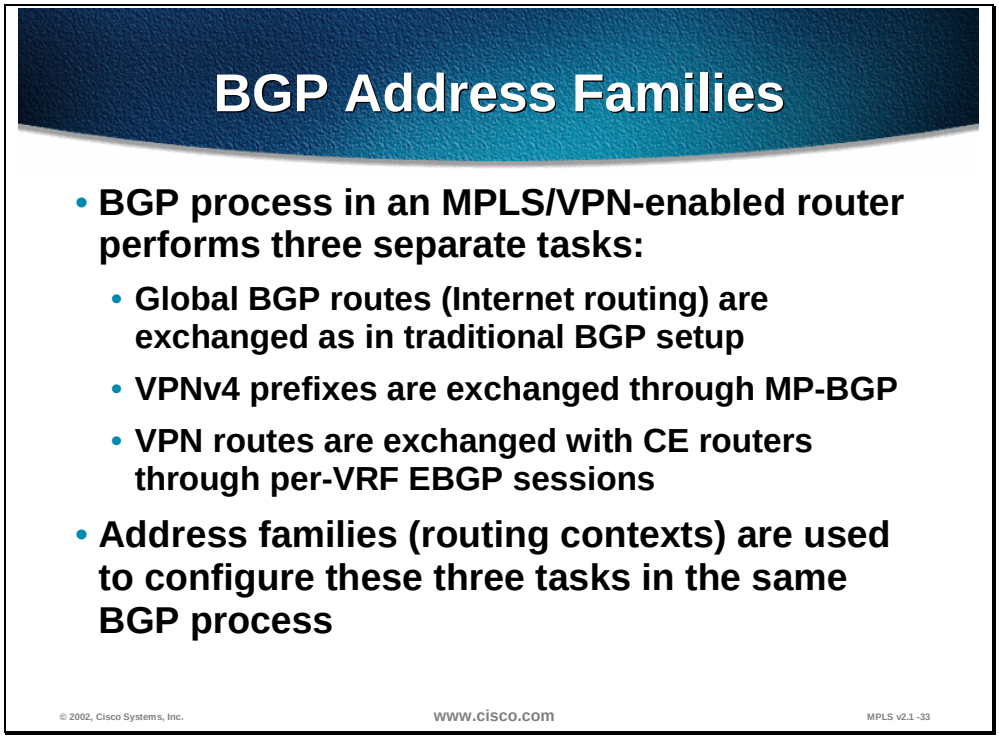
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- Configuring BGP Address Family
- Configuring MP-BGP
- Configuring MP-IBGP
- Configuring BGP Communities Propagation
- Sample VPN Network MP-IBGP Configuration
- Configuring Inter-AS MP-BGP Neighbors
- Sample VPN Network MP-EBGP Configuration
- LSP Stitching
- Next-Hop-Self Option
- Intraconfederation MP-eBGP Sessions
- Selective Activation of MP-BGP Sessions
- Summary
- Lesson Review

Configuring BGP Address Family

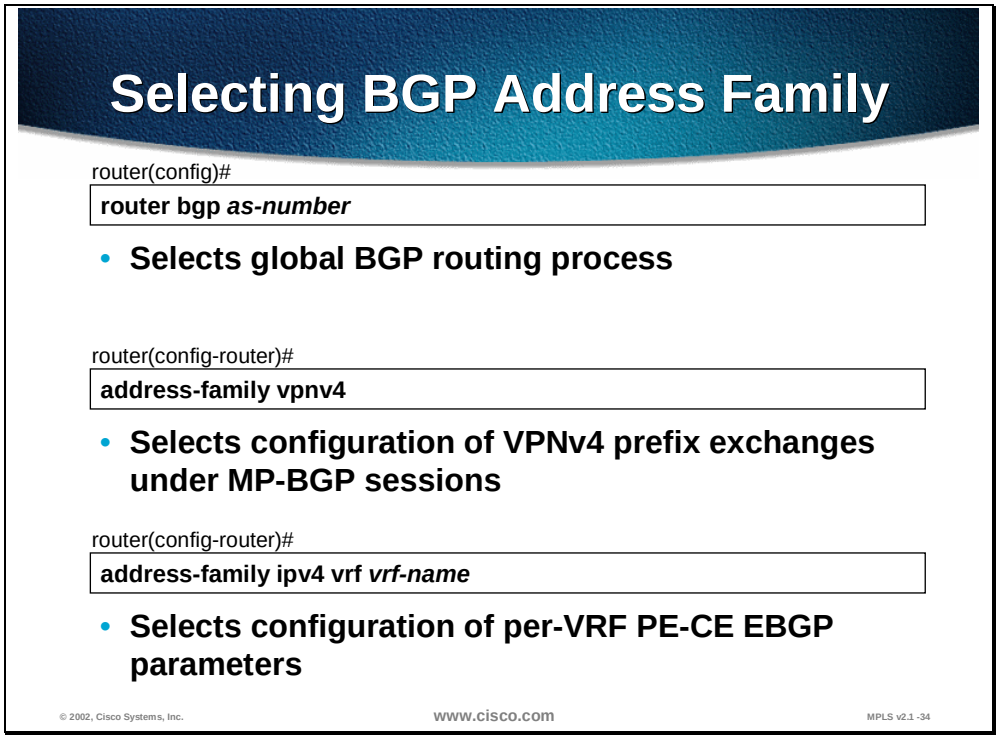


The MPLS/VPN architecture uses BGP routing protocol in two different ways:

- VPNv4 routes are propagated across a MPLS/VPN backbone using multi-protocol BGP between the PE routers
- BGP can be used as the PE-CE routing protocol to exchange VPN routes between the provider edge routers and the customer edge routers

Independently from MPLS/VPN, the PE router can also use BGP to receive and propagate Internet routes in scenarios where the PE routers are also used to provide Internet connectivity to the customers.

All three route exchange mechanisms take place in one BGP process (as you can only configure one BGP process per router) and the routing contexts (called **address families** from router configuration perspective) are used to configure all three independent route exchange mechanisms.



The **address-family** router configuration command is used to select the routing context that you'd like to configure:

- Internet routing (global IP routing table) is the default address family that you configure when you start configuring the BGP routing process;
- To configure multi-protocol BGP sessions between the PE routers, use the **vpnv4** address family
- To configure BGP between the PE routers and the CE routes within individual VRF, use the **ipv4 vrf *name*** address family

router bgp

To configure the Border Gateway Protocol (BGP) routing process, use the **router bgp** global configuration command. To remove a routing process, use the **no** form of this command.

router bgp *autonomous-system*
no router bgp *autonomous-system*

Syntax Description

autonomous-system	Number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along.

Default

No BGP routing process is enabled by default.

address-family

To enter the address family submode for configuring routing protocols, such as BGP, RIP and static routing, use the **address-family** command in address family configuration submode. To disable the address family submode for configuring routing protocols, use the **no** form of this command.

VPN-IPv4 unicast

address-family vpnv4 [unicast]
no address-family vpnv4 [unicast]

IPv4 unicast

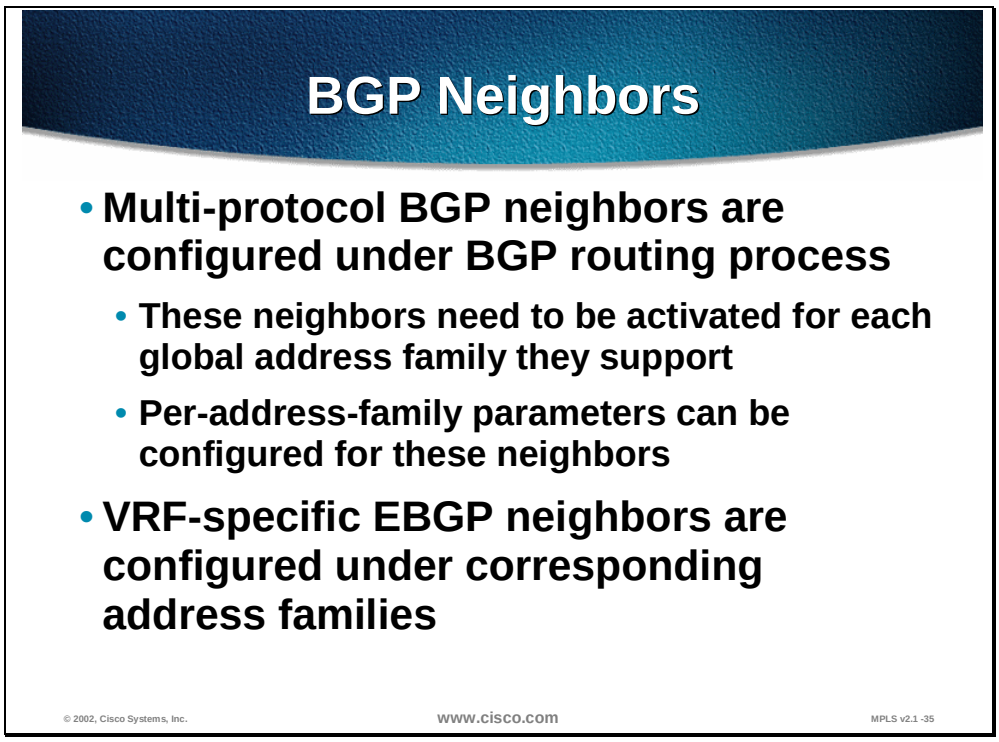
address-family ipv4 [unicast]
no address-family ipv4 [unicast]

IPv4 unicast with CE router

address-family ipv4 [unicast] vrf *vrf-name*
no address-family ipv4 [unicast] vrf *vrf-name*

Syntax Description

ipv4	Configures sessions that carry standard IPv4 address prefixes.
vpnv4	Configures sessions that carry customer VPN-IPv4 prefixes, each of which has been made globally unique by adding an 8-byte route distinguisher.
unicast	(Optional) Specifies unicast prefixes.
vrf *vrf-name*	Specifies the name of a VPN routing/forwarding instance (VRF) to associate with submode commands.



MPLS/VPN architecture defines two types of BGP neighbors:

- Global BGP neighbors (other PE routers), with which the PE router can exchange multiple types of routes. These neighbors are defined in the global BGP definition and only have to be **activated** for individual address families
- Per-VRF BGP neighbors (the CE routers) which are configured and activated within the **ipv4 vrf *name*** address family

Practice

Q1) What is a BGP address family?

A) All the BGP neighbors addresses.

B) The IETF working group for the BGP protocol.

C) The routing protocol context.

D) The BGP version.

Q2) How many BGP address families do you have to configure on a PE router?

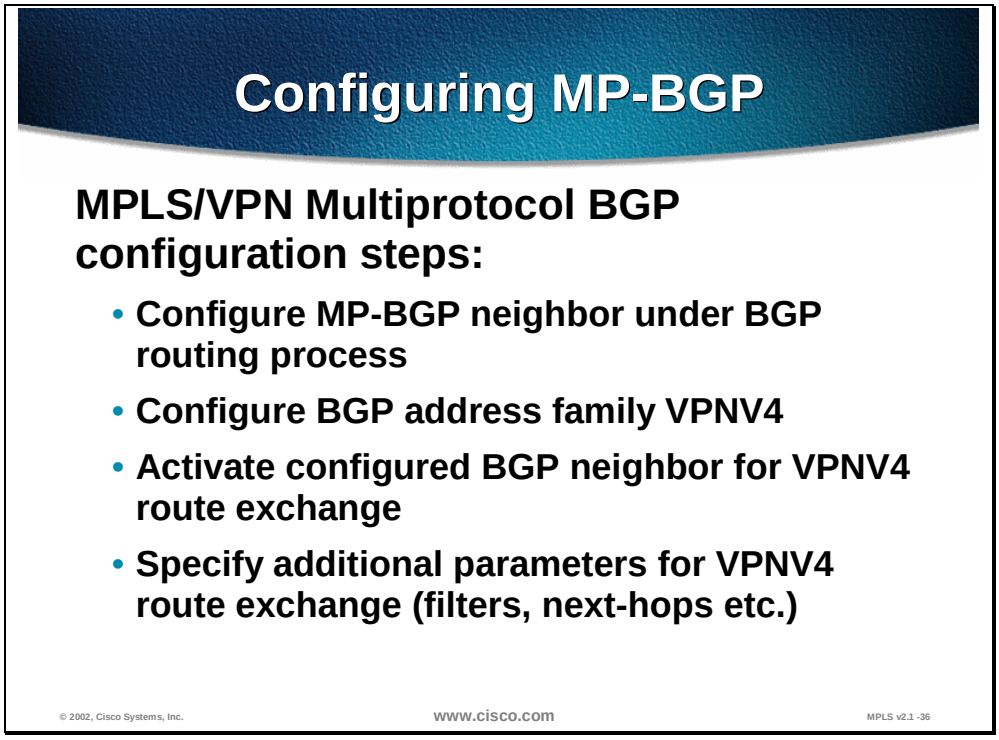
A) PE routers do not use the address family concept.

B) One for each MP-IBGP session and PE-CE sessions.

C) One for all the MP-IBGP sessions and one for each vrf.

D) Only one address family is required in any PE router.

Configuring MP-BGP

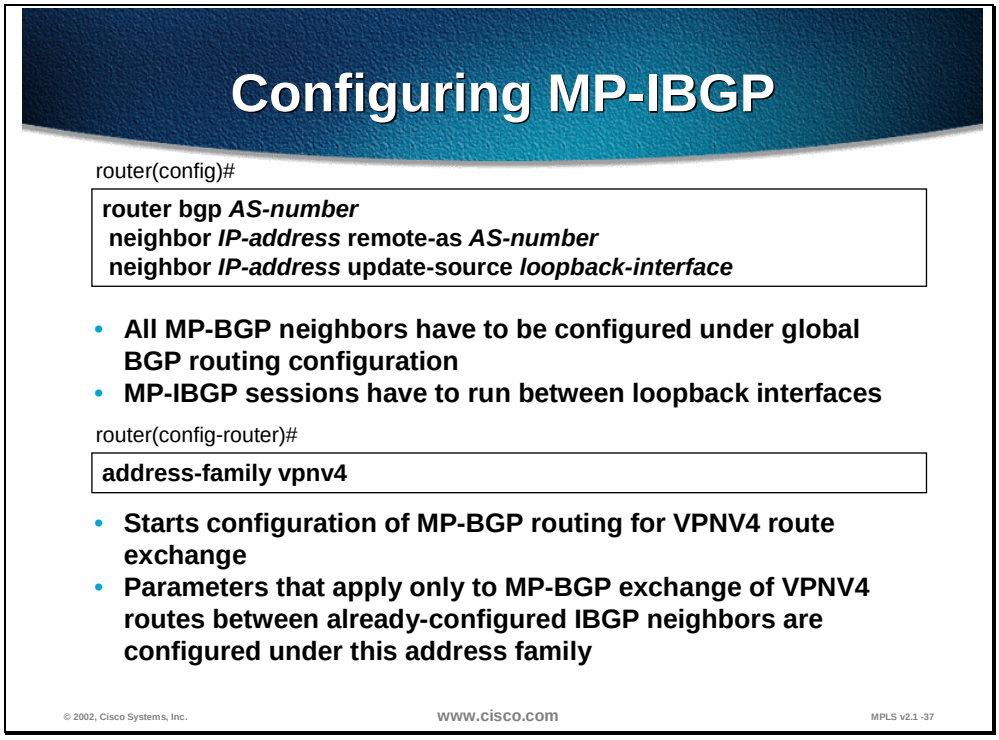


BGP connectivity between two PE routers is configured in four steps:

- The remote PE router is configured as global BGP neighbor under BGP router configuration mode
- Parameters that affect the BGP session itself (for example, source address for the TCP session) are defined on the global BGP neighbor
- VPNv4 address family is selected and the BGP neighbor is activated for VPNv4 route exchange
- Additional VPNv4-specific BGP parameters that affect the VPNv4 routing updates (filters, next-hop processing, route-maps) are configured within the VPNv4 address family

Note IPv4-specific BGP parameters that affect the IPv4 routing updates are still configured under the BGP router configuration mode – there is no special IPv4 address family.

Configuring MP-IBGP



The initial commands that are needed to configure MP-IBGP session between PE routers are:

- **neighbor *address* remote-as *as-number*** command configures the neighboring PE-router
- **neighbor *address* update-source *interface*** command configures the source address used for TCP session carrying BGP updates as well as the IP address used as the BGP next-hop for VPNv4 routes
- **address-family vpnv4** enters the VPNv4 configuration mode where the additional VPNv4-specific parameters have to be configured on the BGP neighbor.

neighbor remote-as

To add an entry to the BGP neighbor table, use the **neighbor remote-as** router configuration command. To remove an entry from the table, use the **no** form of this command.

neighbor {*ip-address* | *peer-group-name*} **remote-as** *number*
no neighbor {*ip-address* | *peer-group-name*} **remote-as** *number*

Syntax Description

ip-address	Neighbor's IP address.
peer-group-name	Name of a BGP peer group.

number	Autonomous system to which the neighbor belongs.

Default

There are no BGP neighbor peers.

neighbor update-source

To have the Cisco IOS software allow internal BGP sessions to use any operational interface for TCP connections, use the **neighbor update-source** router configuration command. To restore the interface assignment to the closest interface, which is called the *best local address*, use the **no** form of this command

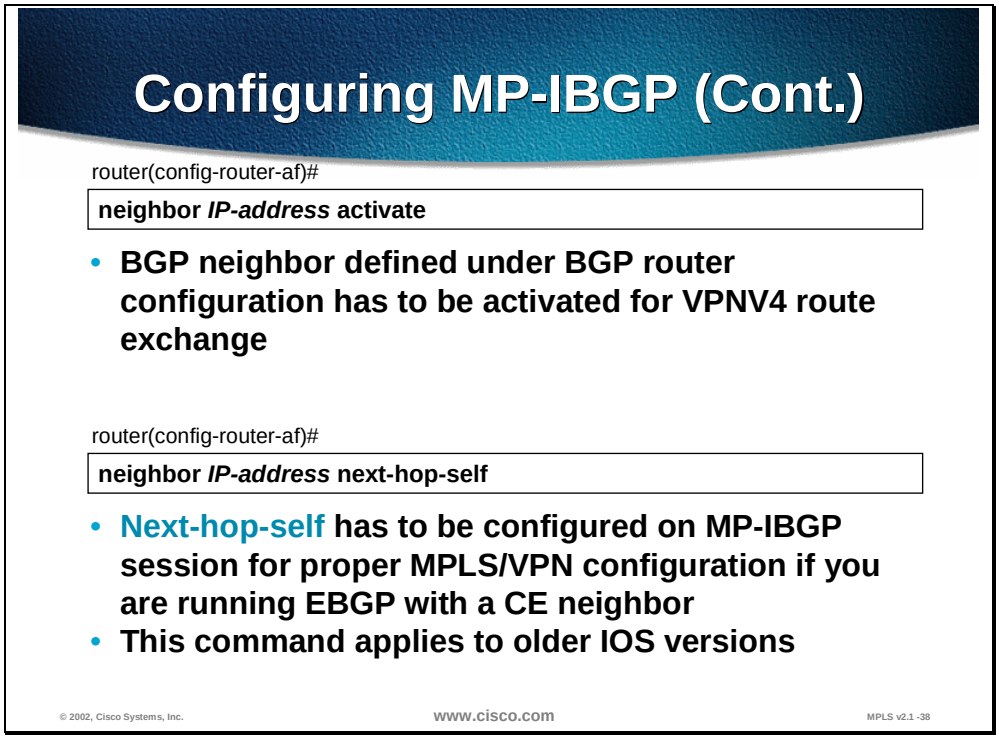
neighbor {*ip-address* | *peer-group-name*} **update-source** *interface*
no neighbor {*ip-address* | *peer-group-name*} **update-source** *interface*

Syntax Description

ip-address	IP address of the BGP-speaking neighbor.
peer-group-name	Name of a BGP peer group.
interface	Loopback interface.

Default

Best local address



After the remote PE router has been defined as a global BGP neighbor, it has to be activated for VPNv4 route exchange. In older IOS versions, the default IBGP next-hop processing needed to be disabled for VPNv4 route exchange with **next-hop-self** command.

Note	If you the default next-hop processing would not be disabled, the VPN IP address of a BGP-speaking CE router might become VPNv4 BGP next hop and the connectivity across the MPLS/VPN backbone is broken.

neighbor activate

To enable the exchange of information with a BGP neighboring router, use the **neighbor activate** router configuration command. To disable the exchange of an address with a neighboring router, use the **no** form of this command.

neighbor {*ip-address* | *peer-group-name*} **activate**
no neighbor {*ip-address* | *peer-group-name*} **activate**

Syntax Description

ip-address	IP address of the neighboring router.
peer-group-name	Name of BGP peer group.

Defaults

The exchange of addresses with neighbors is enabled by default for the IPv4 address family. For all other address families, address exchange is disabled by

default. You can explicitly activate the default command using the appropriate address family submode.

neighbor next-hop-self

To disable next-hop processing of BGP updates on the router, use the **neighbor next-hop-self** router configuration command. To disable this feature, use the **no** form of this command.

neighbor {*ip-address* | *peer-group-name*} **next-hop-self**
no neighbor {*ip-address* | *peer-group-name*} **next-hop-self**

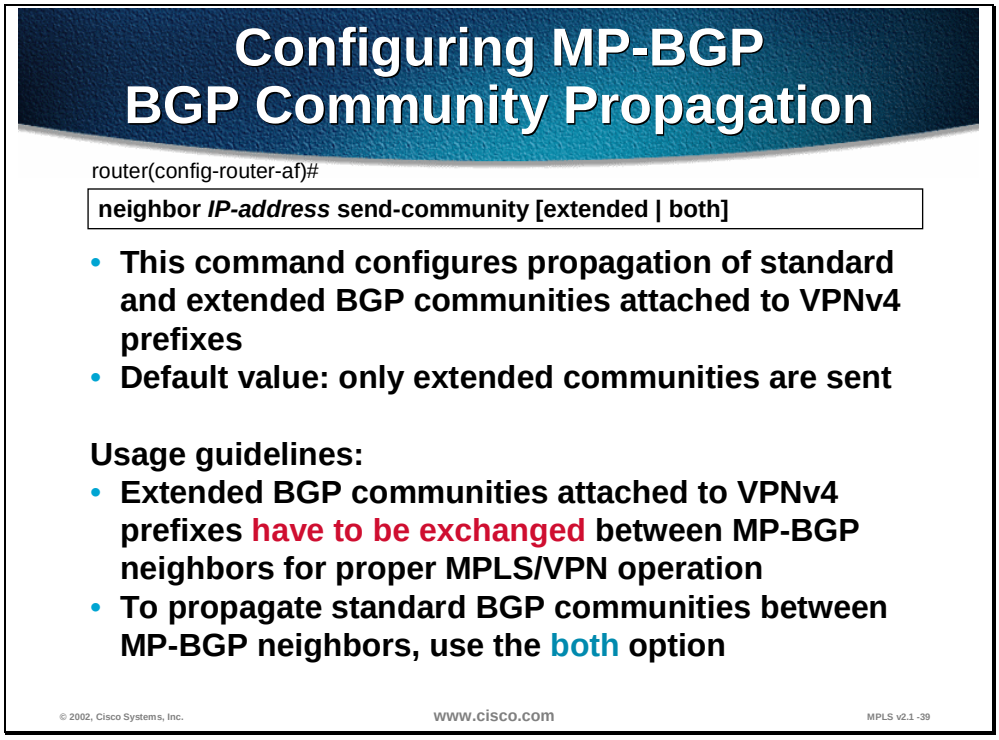
Syntax Description

ip-address	IP address of the BGP-speaking neighbor.
peer-group-name	Name of a BGP peer group.

Default

Disabled

Configuring BGP Communities Propagation



MPLS/VPN architecture has introduced the **extended community** BGP attribute. BGP still supports the **standard community** attribute, which has not been superseded with the extended communities. The default community propagation behavior for standard BGP communities has not changed – community propagation still needs to be configured manually. Extended BGP communities are propagated by default, because their propagation is mandatory for successful MPLS/VPN operation.

The **neighbor send-community** command was extended to support standard and extended communities. You should use this command to configure propagation of standard and extended communities if your BGP design relies on usage of standard communities (for example, to propagate Quality of Service information across the network).

neighbor send-community

To specify that a COMMUNITIES attribute should be sent to a BGP neighbor, use the **neighbor send-community** router configuration command. To remove the entry, use the **no** form of this command.

neighbor {*ip-address* | *peer-group-name*} **send-community [extended | both]**
no neighbor {*ip-address* | *peer-group-name*} **send-community**

Syntax Description

ip-address	Neighbor's IP address.

peer-group-name Name of a BGP peer group.

Default

No COMMUNITIES attribute is sent to any neighbor.

Practice

Q1) How do you enable propagation of standard communities for VPNv4 MP-BGP sessions?

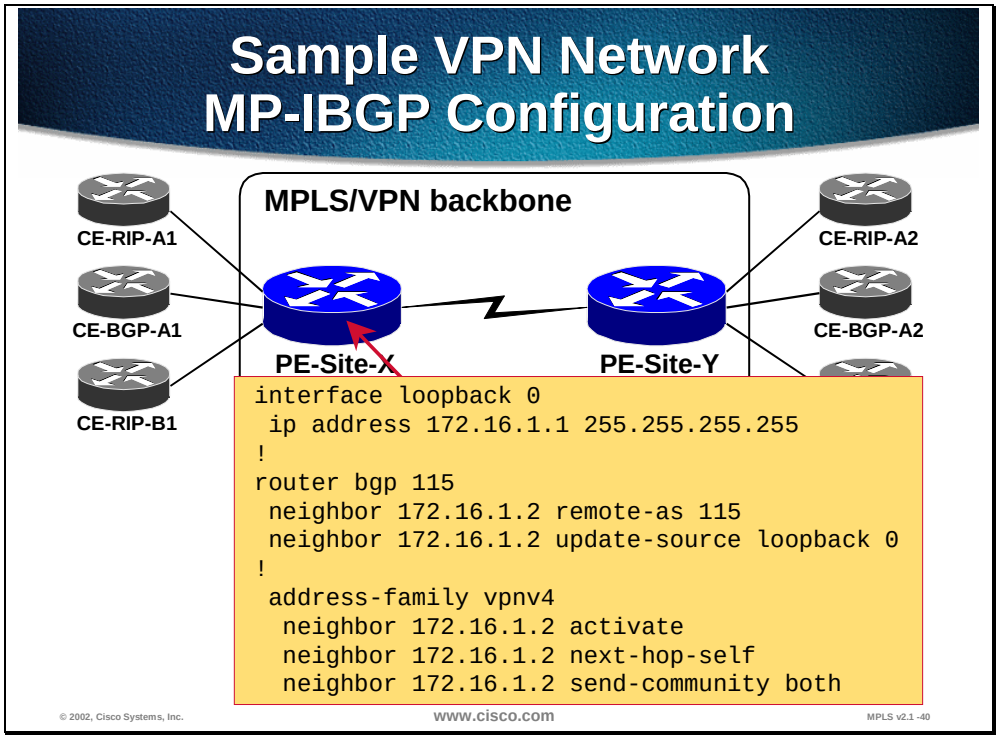
A) neighbor ip send-community both

B) neighbor ip send-community standard

C) neighbor ip send-community

D) neighbor ip send-community extended

Sample VPN Network MP-IBGP Configuration



The configuration example from page 31 continues with the configuration of multi-protocol IBGP sessions on the PE router. The following steps need to be performed:

Step 1 A loopback interface is defined that will serve as the BGP next-hop for VPNv4 routes and as the source address for IBGP session

Step 2 The remote PE router is configured as global BGP neighbor

Step 3 The source address for the TCP session is specified

Step 4 VPNv4 address family is selected

Step 5 The remote PE router is activated for VPNv4 route exchange

Step 6 Next-hop processing is disabled for VPNv4 route exchange in order to guarantee that the **loopback 0** interface will always be the BGP next-hop for VPNv4 routes propagated by this router to its MP-IBGP neighbors

Step 7 Propagation of standard and extended communities is configured

Practice

Q1) Why would you want to disable propagation of IPv4 routing updates between MP-BGP neighbors?

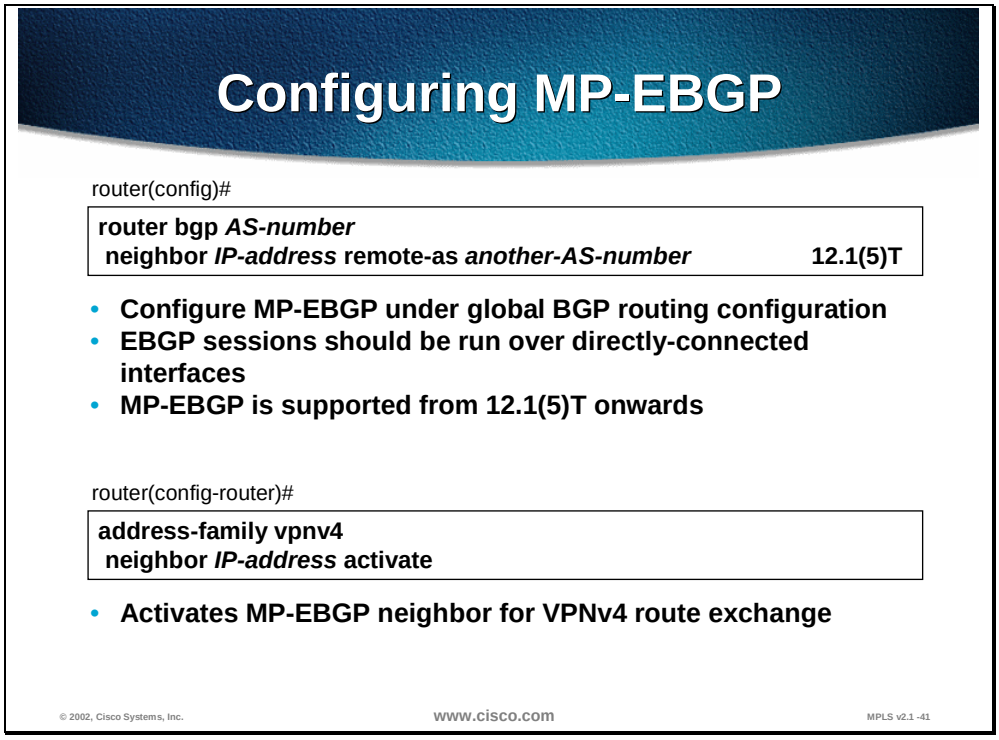
A) To avoid unnecessary memory, bandwidth and CPU consumtion by sending all Internet routes to those PE routers that dont use them, IPv4 route propagation can be selectively disabled.

B) IPv4 routes are never used by PE routers so propagation of them should always be disabled.

C) IPv4 route propagation must never be disabled.

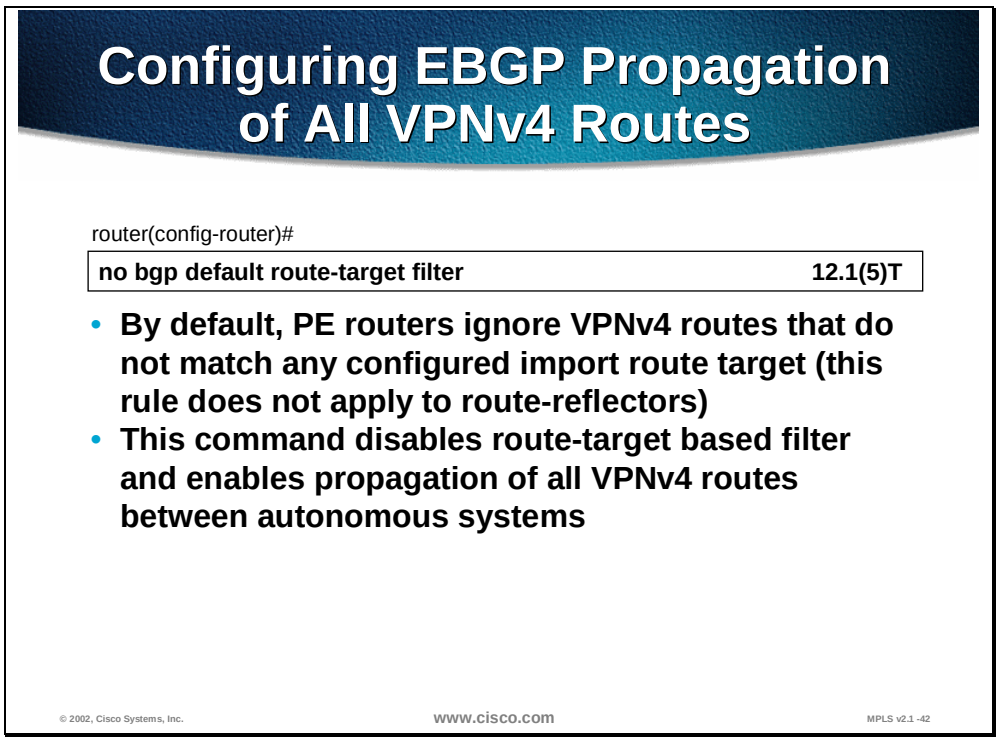
D) If the IPv4 routes overlap the VPN routes, the IPv4 route propagation must be disabled to avoid confusion.

Configuring Inter-AS MP-BGP Neighbors



Multi-protocol EBGP session is configured in exactly the same way as the multi-protocol IBGP session, the only difference being that the AS-number of the neighboring PE-router differs from the local AS-number.

Note The support for VPNv4 information exchange over an EBGP session has been added in IOS release 12.1(5)T.



By default, the PE routers discard VPNv4 updates not related to the VRFs configured on the PE routers, the only exceptions being BGP route reflectors. A PE router exchanging VPNv4 routes over an EBGP session would deploy the same filter (and drop some VPNv4 routes) unless it would be configured as a route reflector. The **no bgp default route-target-filter** command was introduced to disable the default VPNv4 filter and allow the PE router to propagate all VPNv4 routes between autonomous systems.

bgp default route-target filter

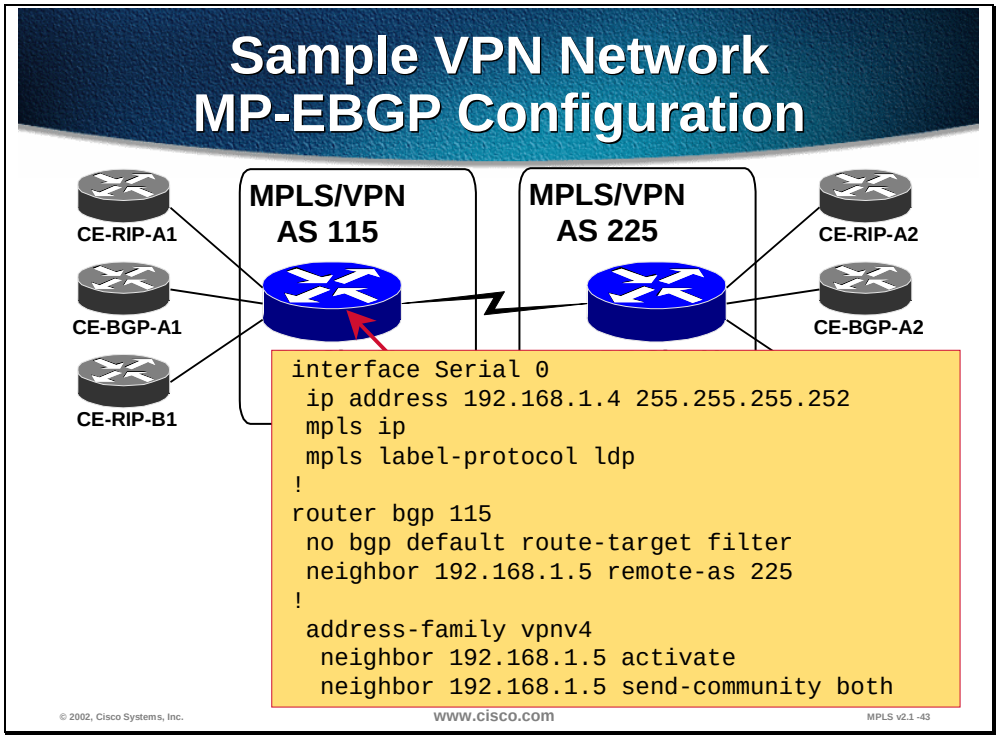
Use this BGP router configuration command to enable filtering of Multiprotocol BGP updates that are not imported into any VRF. Use the **no** form to disable this feature.

bgp defult route-target filter
no bgp defult route-target filter

Default

This feature is enabled by default.

Sample VPN Network MP-EBGP Configuration



The configuration example from page 31 continues with the configuration of multi-protocol IBGP sessions on the PE router. The following steps need to be performed:

Step 8 A loopback interface is defined that will serve as the BGP next-hop for VPNv4 routes and as the source address for IBGP session

Step 9 The remote PE router is configured as global BGP neighbor

Step 10 The source address for the TCP session is specified

Step 11 VPNv4 address family is selected

Step 12 The remote PE router is activated for VPNv4 route exchange

Step 13 Next-hop processing is disabled for VPNv4 route exchange in order to guarantee that the **loopback 0** interface will always be the BGP next-hop for VPNv4 routes propagated by this router to its MP-IBGP neighbors

Step 14 Propagation of standard and extended communities is configured

Practice

Q1) Which are the mandatory parameters that you have to configure on MP-BGP neighbor?

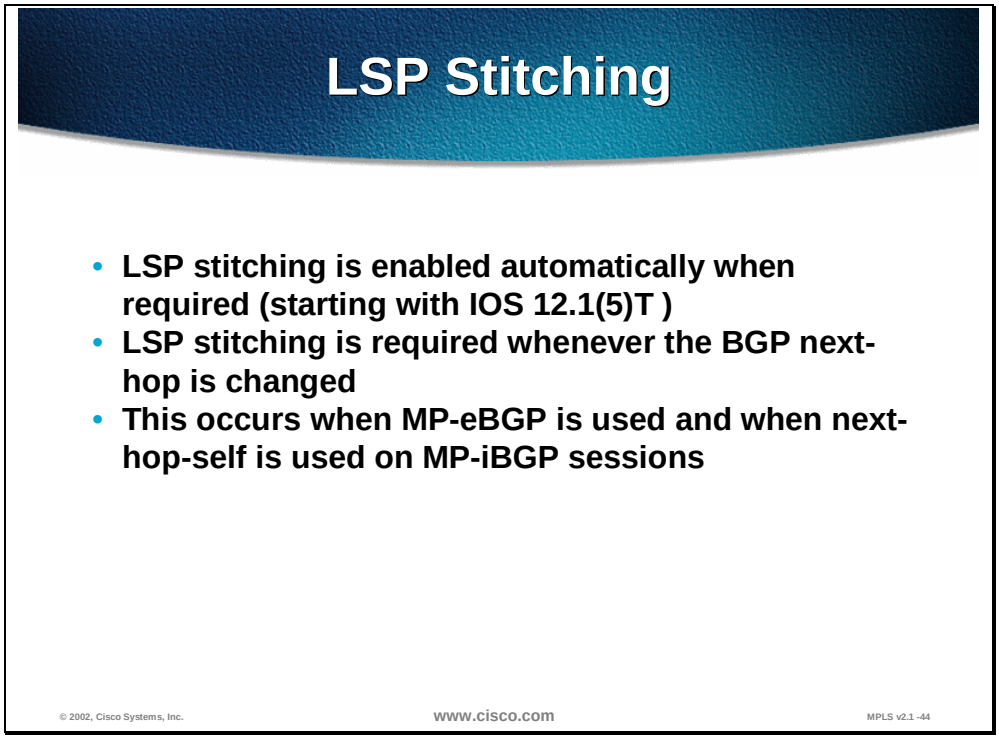
A) neighbor ip send-community extended

B) neighbor ip next-hop self

C) neighbor ip remove-private-as

D) neighbor ip activate

LSP Stitching

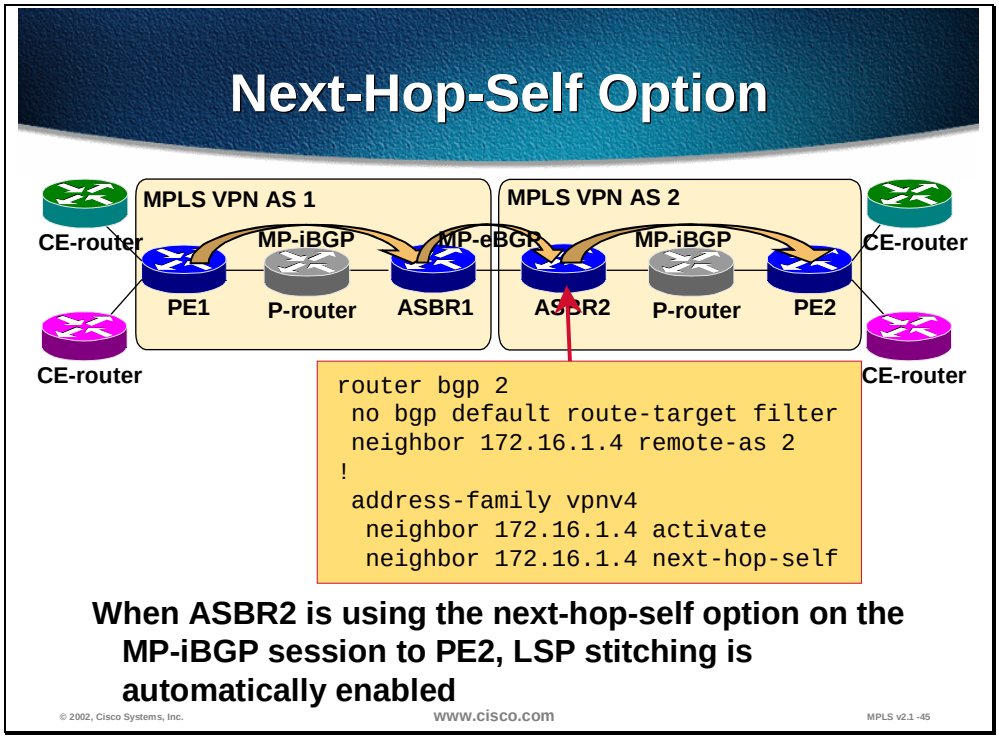


Support for LSP stitching was introduced in IOS 12.1(5)T

LSP Stitching makes it possible to change the BGP next-hop attribute when VPNv4 routes are propagated. Whenever the next-hop is changed, the router that performs the change assigns a dedicated label which is associated with old next-hop. The new label and the new next-hop are propagated together with the VPNv4 route.

Next-hop is changed on MP-eBGP sessions and when the next-hop-self option is used on MP-iBGP sessions. The LSP stitching is enabled by default and does not require any additional configuration.

Next-Hop-Self Option

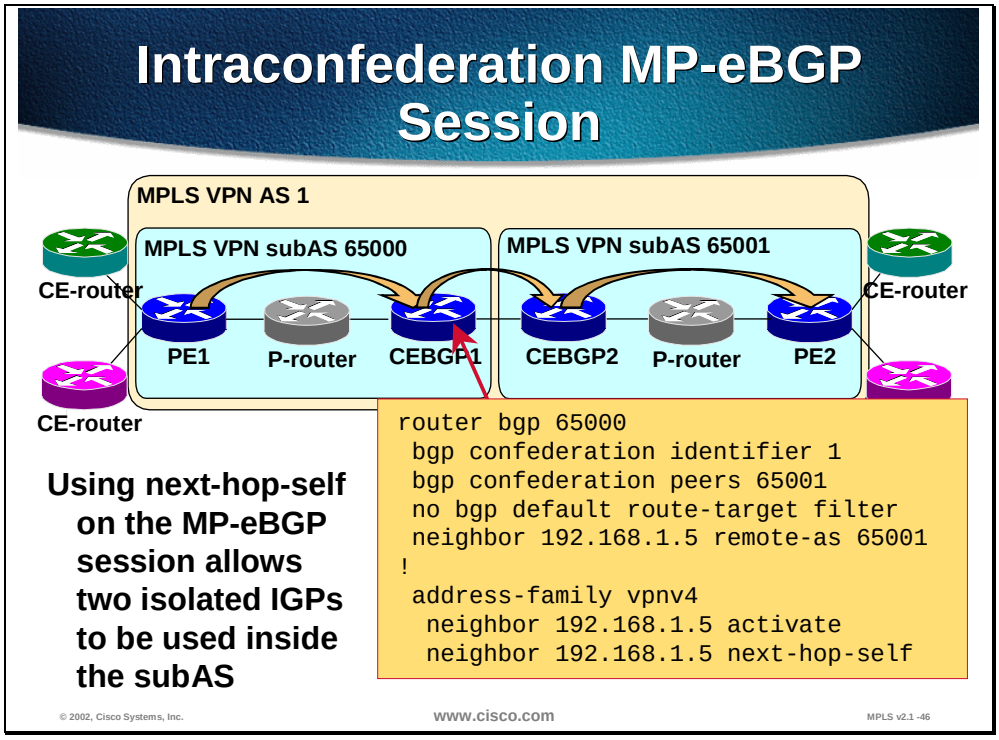


The figure illustrates a network where the next-hop-self option is used. ASBR2 receives VPNv4 route on the MP-eBGP session from ASBR1. The next-hop-self is used when ASBR2 propagates these routes on the MP-iBGP session to PE2.

Modifying the next-hop means that the VPN label must also be modified. A dedicated label is assigned by ASBR2 and propagated to PE2. This label will be used by ASBR2 to map incoming traffic from PE2 into the correct LSP toward ASBR1.

The feature is automatically enabled and does not require any additional configuration.

Intraconfederation MP-eBGP Sessions



Using next-hop-self on intraconfederation MP-eBGP session makes it possible to use isolated IGPs in the two subAS.

The BGP confederation feature is used to scale BGP by relaxing the requirement for full IBGP mesh. In this case, however, the confederation feature is also used to scale the IGP. A separate IGP process is in use in each subAS.

Because PE1 and PE2 are in two different subAS with two different IGP processes, PE2 does not have any IGP route how to reach PE1. As a consequence, the next-hop must be changed from the value indicating PE1 to the value indicating CEBGP1 as the BGP route is propagated across the intraconfederation subAS boundary.

Changing of next-hop on intraconfederation EBGP sessions is not enabled by default. It has to be configured. And since the parameter change affects the update sent rather than the BGP session as such, the configuration must be done in the address family configuration mode.

The next-hop-self parameter is configured for the neighbor 192.168.1.5 under the address family vpnv4.

Practice

Q1) Why would you want to disable propagation of IPv4 routing updates between MP-BGP neighbors?

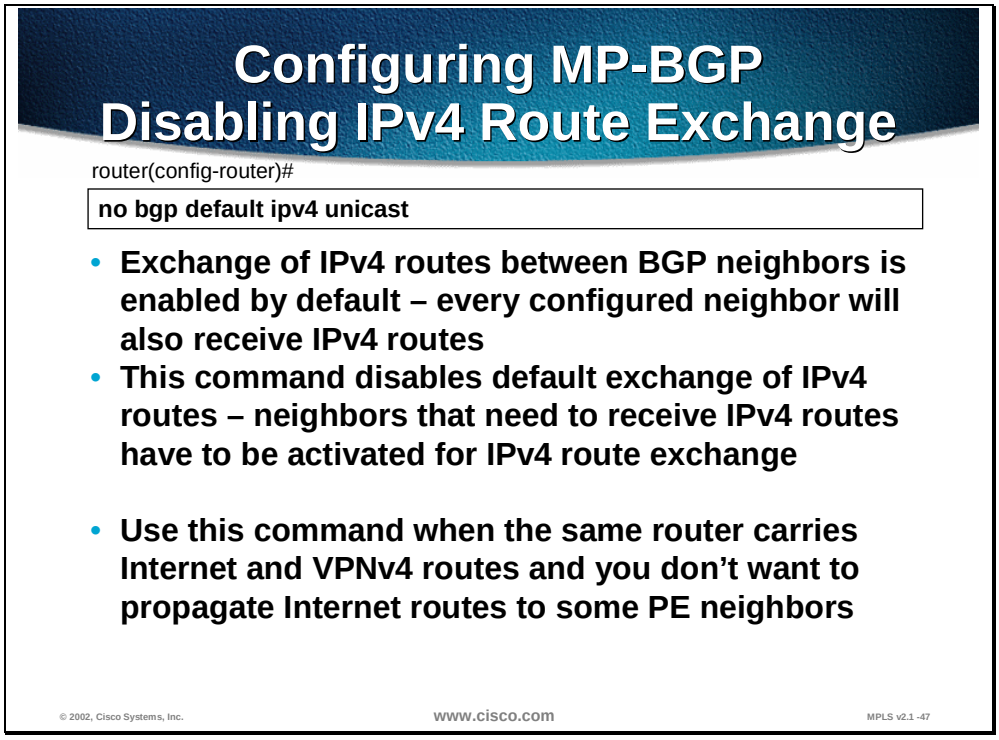
A) To avoid unnecessary memory, bandwidth and CPU consumption by sending all Internet routes to those PE routers that don't use them, IPv4 route propagation can be selectively disabled.

B) IPv4 routes are never used by PE routers so propagation of them should always be disabled.

C) IPv4 route propagation must never be disabled.

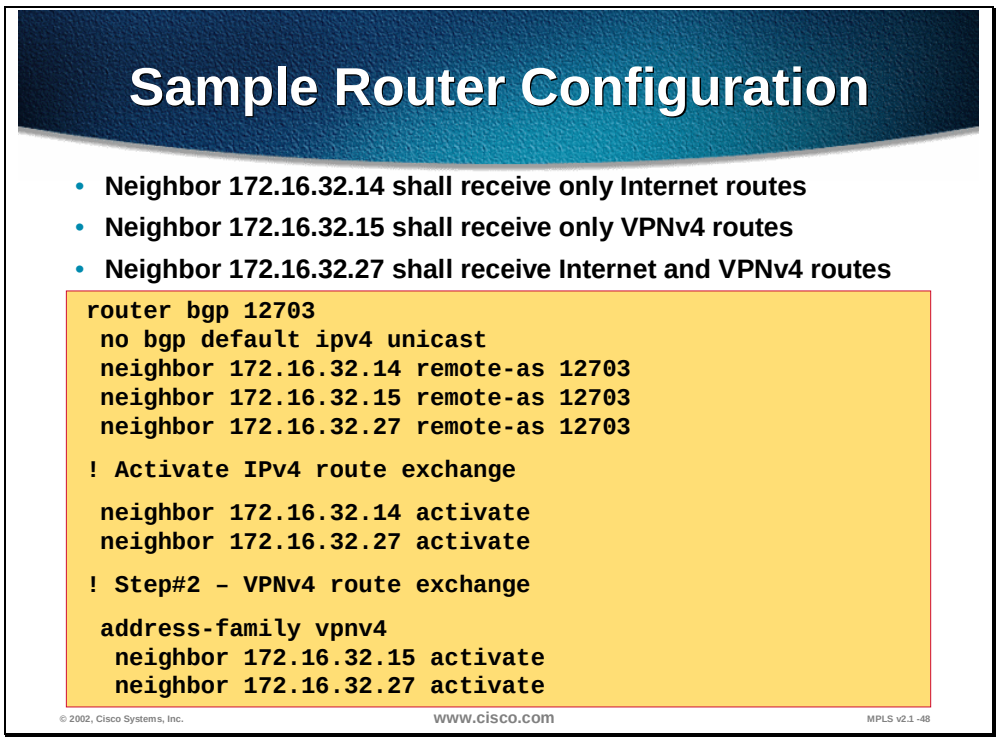
D) If the IPv4 routes overlap the VPN routes, the IPv4 route propagation must be disabled to avoid confusion.

Selective Activation of MP-BGP Session



The BGP configuration discussed so far is appropriate for scenarios where the PE routers provide Internet and VPN connectivity. If the PE routers provide only VPN connectivity, they don't need Internet routing and the IPv4 route exchange needs to be disabled. There are two ways of disabling IPv4 route exchange:

- If you only want to disable IPv4 route exchange for a few neighbors, the best option is to disable the IPv4 route exchange on a neighbor-by-neighbor basis by using **no neighbor activate** command
- If you want to disable IPv4 route exchange for most (or all) of the neighbors, you can use **no bgp default ipv4 unicast** command. After you enter this command, IPv4 route exchange has to be manually activated for each configured global BGP neighbor.



In this example, only a subset of BGP neighbors needs to receive IPv4 routes. The default propagation of IPv4 routes is thus disabled and IPv4 route exchange as well as VPNv4 route exchange is manually activated on a neighbor-by-neighbor basis.

Practice

Q1) How is the propagation of IPv4 routing updates between MP-BGP neighbors disabled?

A) Use no bgp default ipv4 unicast in the global BGP configuration mode.

B) Use neighbor ip shutdown in the global BGP configuration mode.

C) Use neighbor ip prefix-list XY in and ip prefix-list XY deny 0.0.0.0/0.

D) Use no bgp default route-target filter in the global BGP configuration mode.

Summary

After completing this section, you should be able to perform the following tasks:

- Configure BGP address families
- Configure MP-BGP neighbors
- Configure inter-AS MP-BGP neighbors
- Configure additional mandatory parameters on MP-BGP neighbors
- Configure propagation of standard and extended BGP communities
- Selectively enable IPv4 and MP-BGP sections between BGP neighbors

Next Steps

After completing this lesson, go to:

- Configuring Routing Protocols Between PE and CE Routers

Lesson Review

Instructions

Answer the following questions:

1. What is a BGP address family?
2. How many BGP address families do you have to configure on a PE router?
3. In which address family is the MP-IBGP neighbor configured?
4. What are the mandatory parameters that you have to configure on a MP-BGP neighbor?
5. What additional parameters have to be configured to support MP-EBGP neighbors?
6. How do you enable community propagation for VPNv4 MP-BGP sessions?
7. Why would you want to disable propagation of IPv4 routing updates between MP-BGP neighbors?
8. How is the propagation of IPv4 routing updates between MP-BGP neighbors disabled?

Configuring Routing Protocols between PE and CE Routers

Overview

This lesson describes configuration of various PE-to-CE routing protocols.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Configure VRF address families in routing protocols
- Configure per-VRF BGP parameters
- Configure static routes within a VRF
- Configure per-VRF OSPF process
- Propagate RIP, OSPF, and static routes across a MP-BGP backbone

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

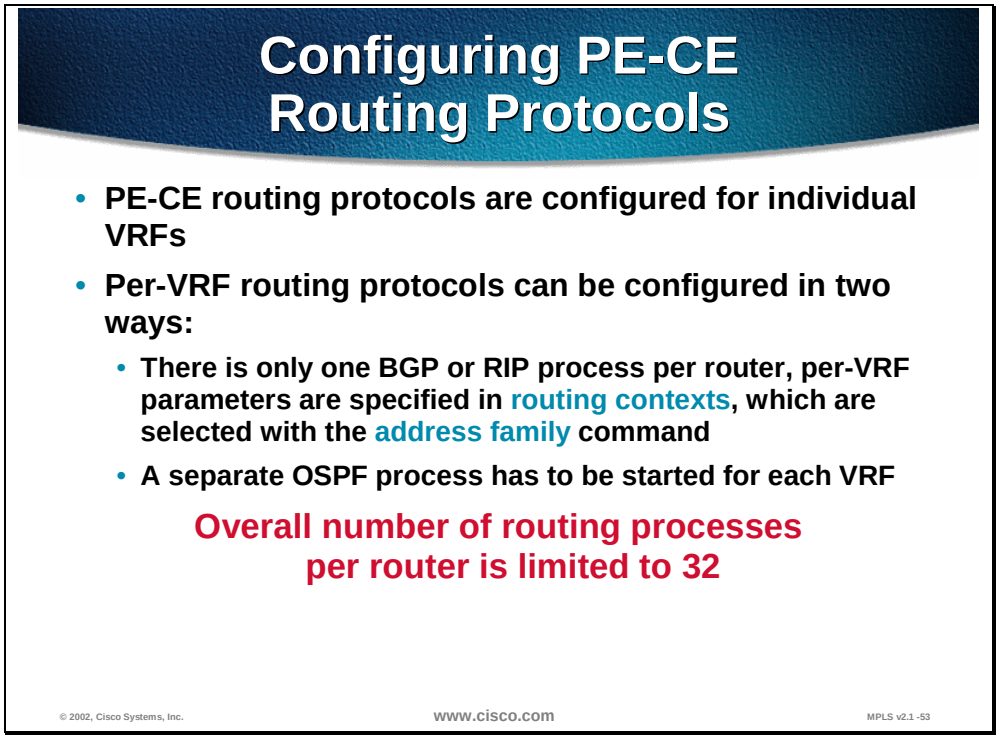
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- Configuring PE-CE Routing Protocols Limitations
- Configuring Address Families Inside Routing Processes
- Configuring per-VRF BGP Neighbors and Other BGP Parameters
- Configuring RIP Between PE and CE Router
- Propagating RIP Routes across MP-BGP Backbone
- Configuring per-VRF OSPF Process
- Configuring Static VRF Routes
- Summary
- Lesson Review

Configuring PE-CE Routing Protocols Limitations



After configuring VRFs and establishing MP-IBGP connectivity between PE routers, you have to configure routing protocols between the PE router and the attached CE routers. The PE-CE routing protocols need to be configured for individual VRFs – sites in the same VPN, but in different VRFs, cannot share the same PE-CE routing protocol.

Note	The per-VRF configuration of the PE-CE routing protocols is another good reason for grouping as many sites into a VRF as possible.

The per-VRF routing protocols can be configured in two ways:

- As individual **address families** belonging to the same routing process (similar to what you've already seen for BGP) or
- As separate routing processes. This option is used for more complex routing protocols that need to maintain separate topology database for each VRF, for example, OSPF

Note	Current IOS implementation limits the overall number of routing protocols in a router to 32. Two routing methods are predefined (static and connected) and two routing protocols are needed for proper MPLS/VPN backbone operation (BGP and backbone IGP). The number of PE-CE routing processes is therefore limited to 28.

Practice

Q1) How many VPN OSPF processes can run simultaneously in an MPLS VPN PE-router?

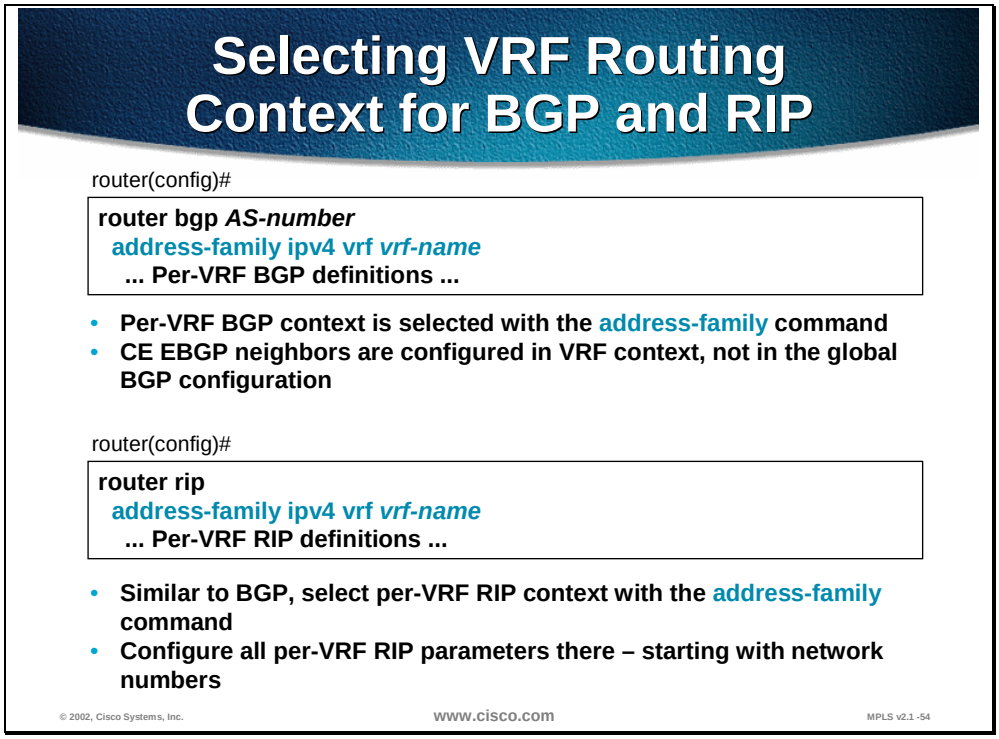
A) Only one process is used for all routing protocol contexts.

B) Up to the number of VRFs (it is only limitted by the amount of memory).

C) Up to 28.

D) Up to 32.

Configuring Address Families Inside Routing Processes



The VRF routing context is selected with the **address-family ipv4 vrf *name*** command in the RIP and BGP routing processes. All per-VRF routing protocol parameters (network numbers, passive interfaces, neighbors, filters etc.) are configured under this address family.

Note	Common parameters defined in the router configuration mode are inherited by all address families defined for this routing process and can be overridden for each individual address family.

router rip

To configure the Routing Information Protocol (RIP) routing process, use the **router rip** global configuration command. To turn off the RIP routing process, use the **no** form of this command.

router rip
no router rip

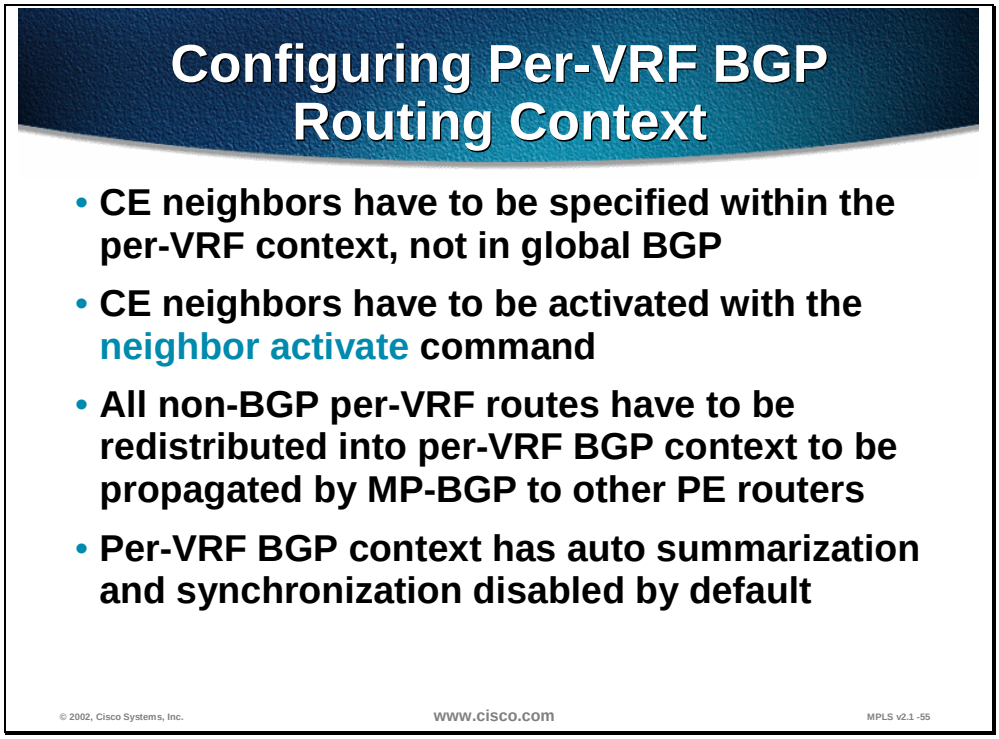
Syntax Description

This command has no arguments or keywords.

Default

No RIP routing process is defined.

Configuring per-VRF BGP Neighbors and Other BGP Parameters



When configuring BGP as the PE-CE routing protocol, start the per-VRF BGP configuration with the **address-family ipv4 vrf *name*** router configuration command. After entering the address family configuration mode, you define the BGP neighbors and activate them. You also have to configure redistribution from all other per-VRF routing protocols into BGP.

Note	You always have to configure BGP address-family for each VRF and configure route redistribution into BGP for each VRF even if you don't use BGP as the PE-CE routing protocol

Several BGP options have different default values when you configure per-VRF BGP routing context:

- BGP synchronization is **disabled** (default = enabled)
- Auto-summarization (automatic generation of classful networks out of subnets redistributed into BGP) is **disabled** (default = enabled), as the MPLS/VPN backbone has to propagate customer subnets unchanged to facilitate transparent end-to-end routing between customer sites
- Redistribution of internal BGP routes into IGP is **enabled** (default = disabled)

Practice

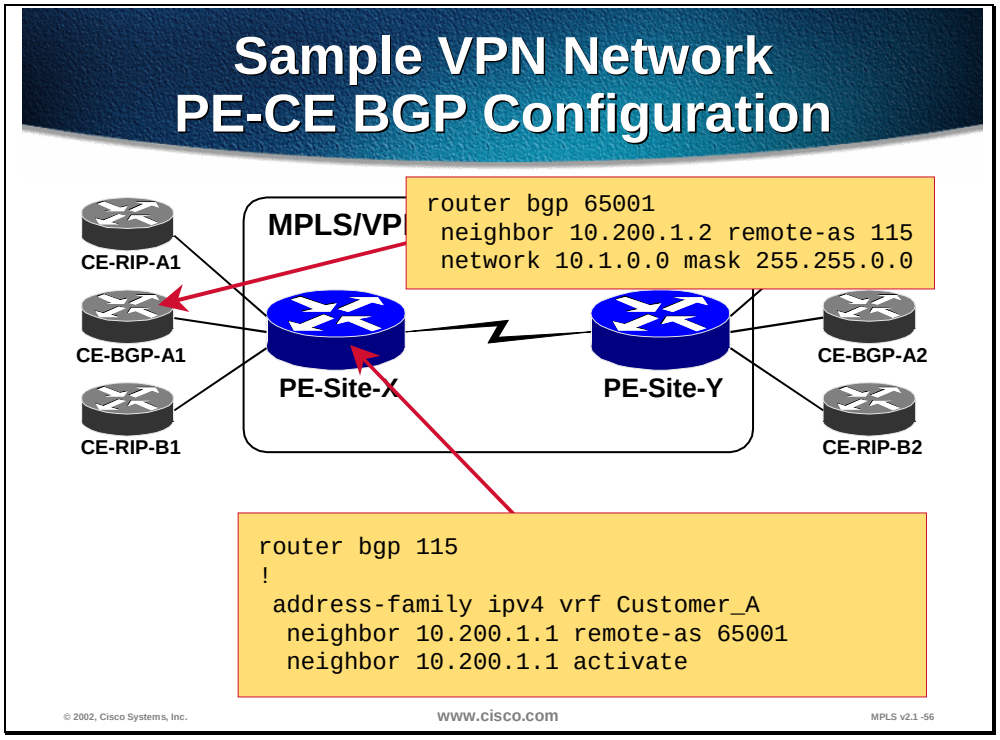
Q1) How do you configure routing context in RIP?

A) Use the ip vrf vrf command in the RIP configuration mode.

B) Use the router rip command in the VRF configuration mode.

C) Start the RIP process by using the router rip vrf vrf command.

D) Use the address-family ipv4 vrf command in the RIP configuration mode.



Continuing the example from page, BGP is started on the CE router, and the PE router is defined as a BGP neighbor. Similarly, the CE router is defined as a BGP neighbor and activated under address-family *ipv4 vrf Customer_A*.

Practice

Q1) Where do you configure CE EBGP neighbor?

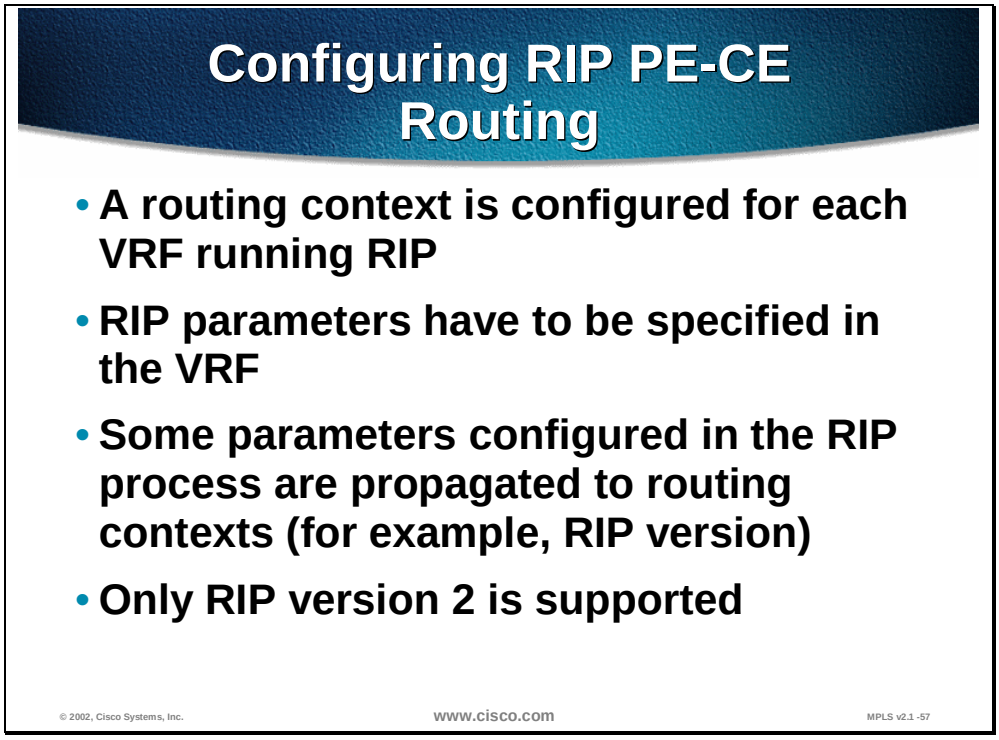
A) In the global (IPv4 unicast) address family and activate it in the customer VRF address family.

B) In the customer VRF IPv4 address family (customer VRF).

C) In the global (IPv4 unicast) address family and activate it in the VPNv4 address family.

D) In the customer VRF address family and activate it in the VPNv4 address family.

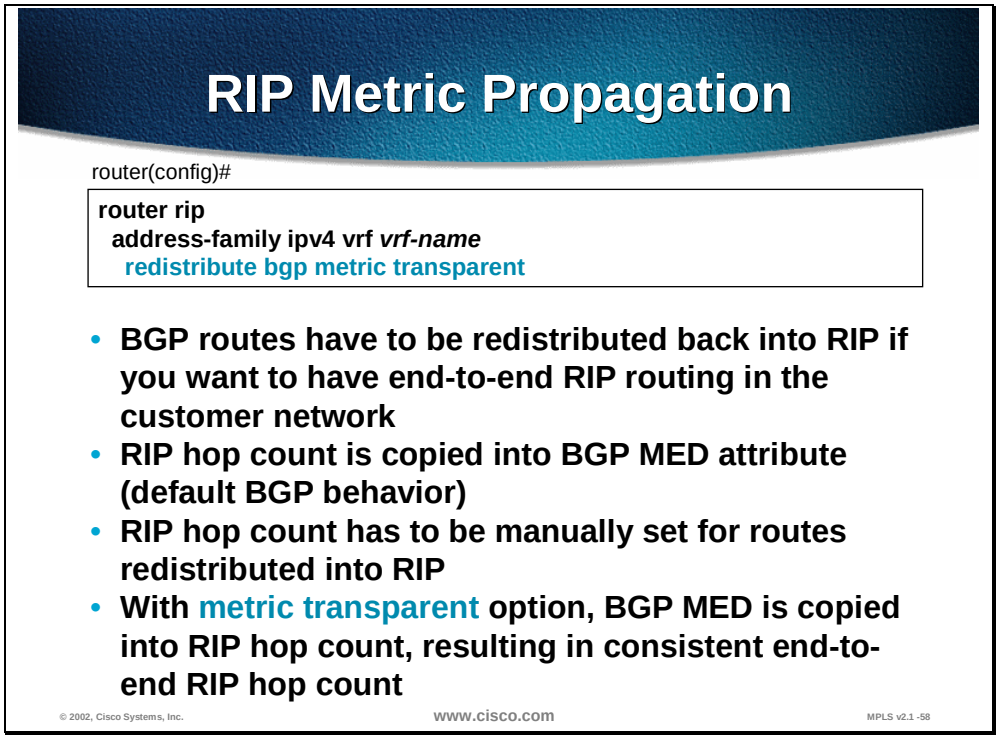
Configuring RIP Between PE and CE Router



Configuring RIP as the PE-CE routing protocol is even simpler than configuring BGP. You start the configuration of individual routing context with the **address-family ipv4 vrf *name*** router configuration command. All standard RIP parameters can be entered in the per-VRF routing context. Global RIP parameters entered in the scope of RIP router configuration are inherited by each routing context and can be overwritten if needed in each routing context.

Note Only RIPv2 is supported as the PE-CE routing protocol. It's a good configuration practice to configure RIP version as a global RIP parameter using the **version 2** router configuration command.

Propagating RIP Routes Across MP-BGP Backbone

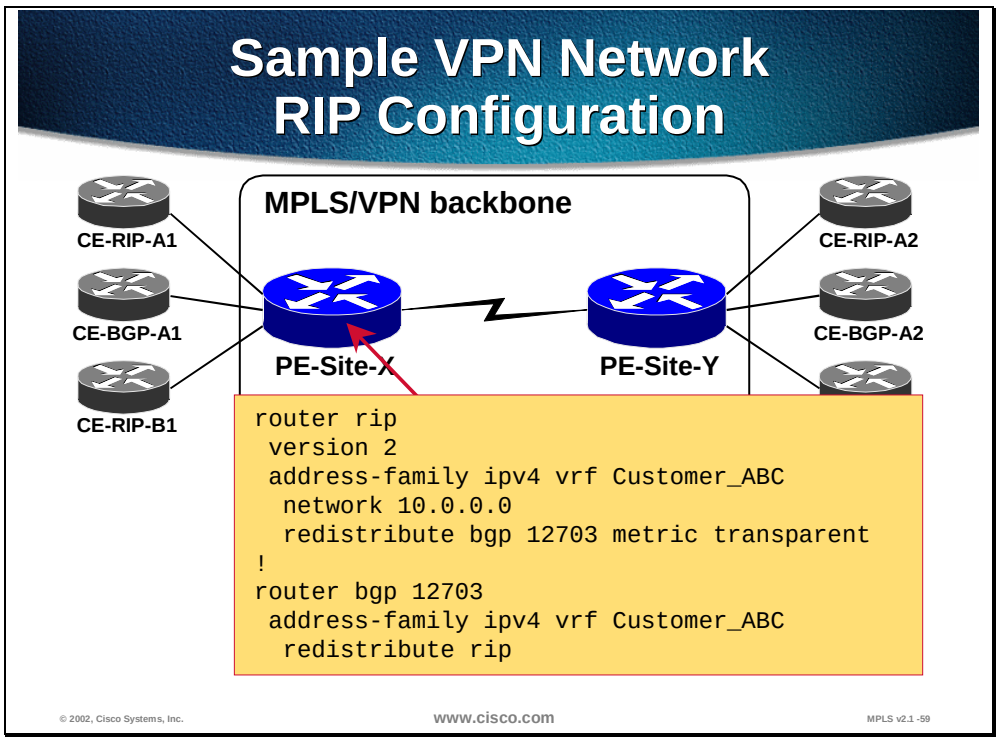


IGP metric is always copied into the MED attribute of the BGP route when an IGP route is redistributed into BGP. Within standard BGP implementation, the MED attribute is only used as a route selection criterion and is not copied back into the IGP metric – the IGP metric has to be specified in the **redistribute** command or by using the **default-metric** router configuration command.

The MPLS/VPN extension to the redistribute command – **metric transparent** option – allows MED to be inserted as the IGP metric of a route redistributed from BGP back into RIP. This extension gives you a transparent end-to-end (from customer's perspective) RIP routing:

- RIP hop count is inserted into BGP attribute MED when the RIP route is redistributed into BGP by the ingress PE router (enabled by default)
- The value of MED attribute (the original RIP hop count) is copied into RIP hop count, if so configured, when the BGP route is redistributed back into RIP. The whole MPLS/VPN backbone thus looks like a single hop to the CE routers.

Note You should **not** change the MED value within BGP if you use the **redistribute metric transparent** option.



RIP configuration in our sample network is exceedingly simple:

- The RIP routing process is configured. The RIP version is configured as the global RIP parameter
- The RIP routing context is configured for every VRF where you want to run RIP as the PE-CE routing protocol. The directly connected networks (configured on interfaces in the VRF) over which you want to run RIP are specified to be with standard RIP configuration
- Redistribution from BGP into RIP with metric propagation is configured
- BGP routing context is configured for every VRF. Redistribution of RIP routes into BGP has to be configured for every VRF for which you've configured the RIP routing context

Practice

Q1) How do you propagate RIP metric across an MPLS VPN backbone?

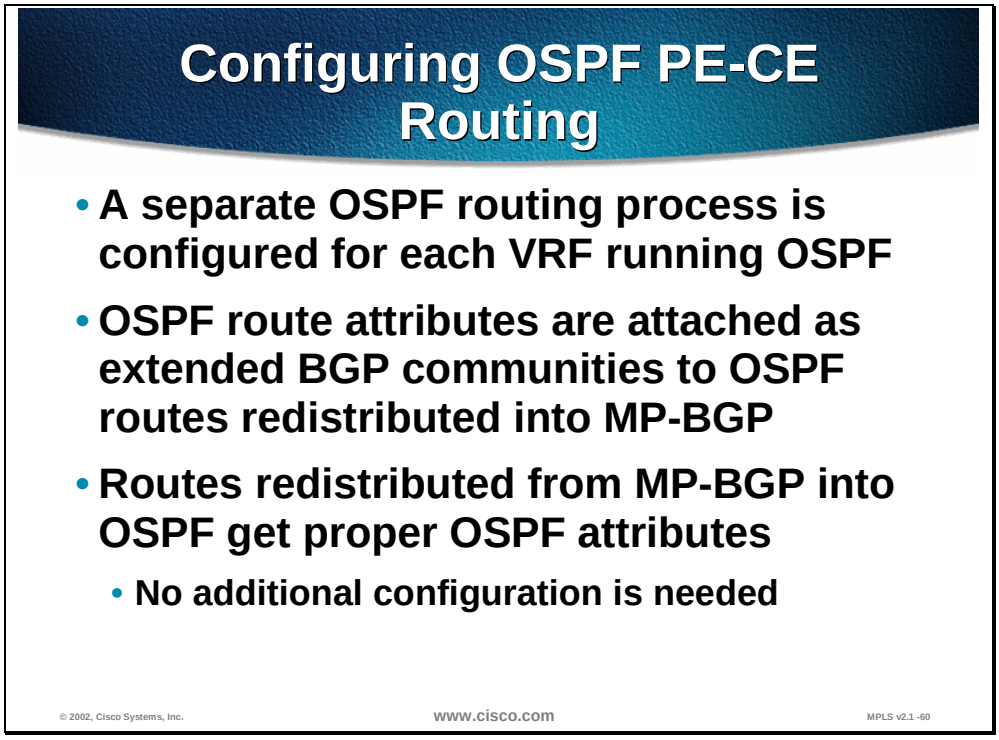
A) RIP hop count can be translated into BGP MED by using a route map on redistribution.

B) RIP hop count cannot be preserved across an MPLS backbone.

C) RIP hop count is carried in the RIP-hop-count extended community.

D) RIP hop count is automatically copied int BGP MED attribute.

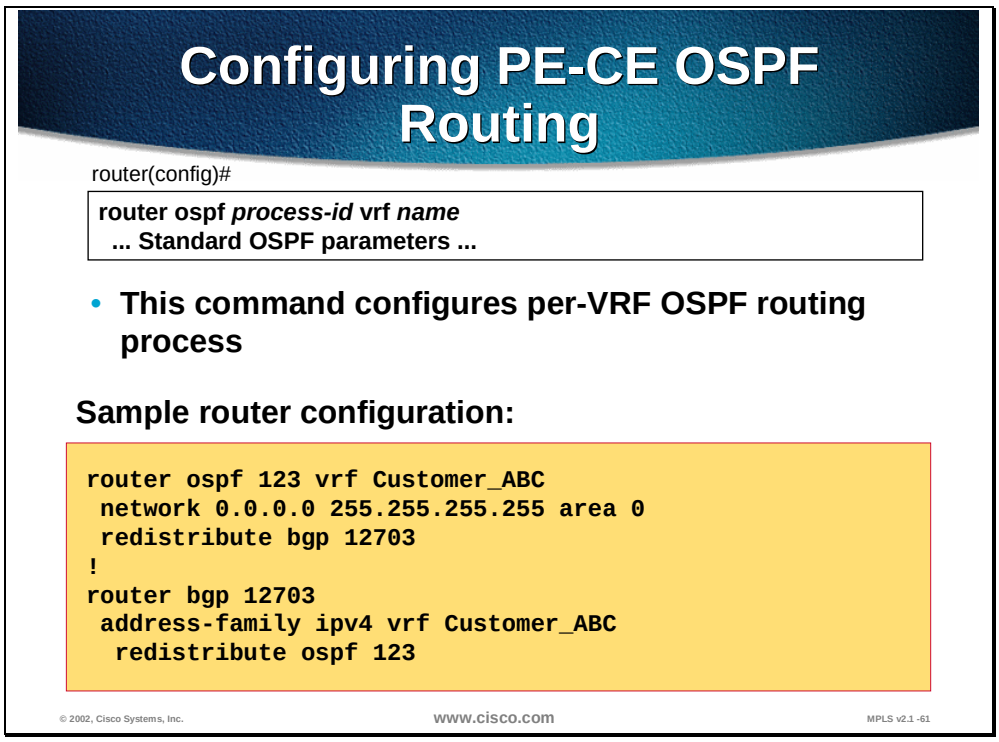
Configuring per-VRF OSPF Process



To configure OSPF as a PE-CE routing protocol, you need to start a separate OSPF process for each VRF in which you want to run OSPF. The pre-VRF OSPF process is configured in the same way as a standard OSPF process; you can use all OSPF features available in Cisco IOS.

Redistribution of OSPF routes into BGP has to be configured for RIP and the redistribution of BGP routes into OSPF can be configured if necessary. Alternatively, you can originate a default route into a per-VRF OSPF process by using the **default-information originate always** OSPF router configuration command.

Multi-protocol BGP propagates more than just OSPF cost across the MPLS/VPN backbone – please refer to the **Running OSPF in a VPN** lesson for more details. The propagation of additional OSPF attributes into MP-BGP is automatic and requires no extra configuration.



OSPF is the only PE-CE routing protocol, which is not fully VPN aware. A separate OSPF process is run for every VRF.

router ospf

To configure an OSPF routing process within a VRF, use the **router ospf** global configuration command. To terminate an OSPF routing process, use the **no** form of this command.

router ospf *process-id* **vrf** *vrf-name*
no router ospf *process-id* **vrf** *vrf-name*

Syntax Description

process-id	Internally used identification parameter for an OSPF routing process. It is locally assigned and can be any positive integer. A unique value is assigned for each OSPF routing process.
vrf-name	The name of the VRF where the OSPF process will reside.

Default

No OSPF routing process is defined.

Practice

Q1) How do you configure routing context in OSPF?

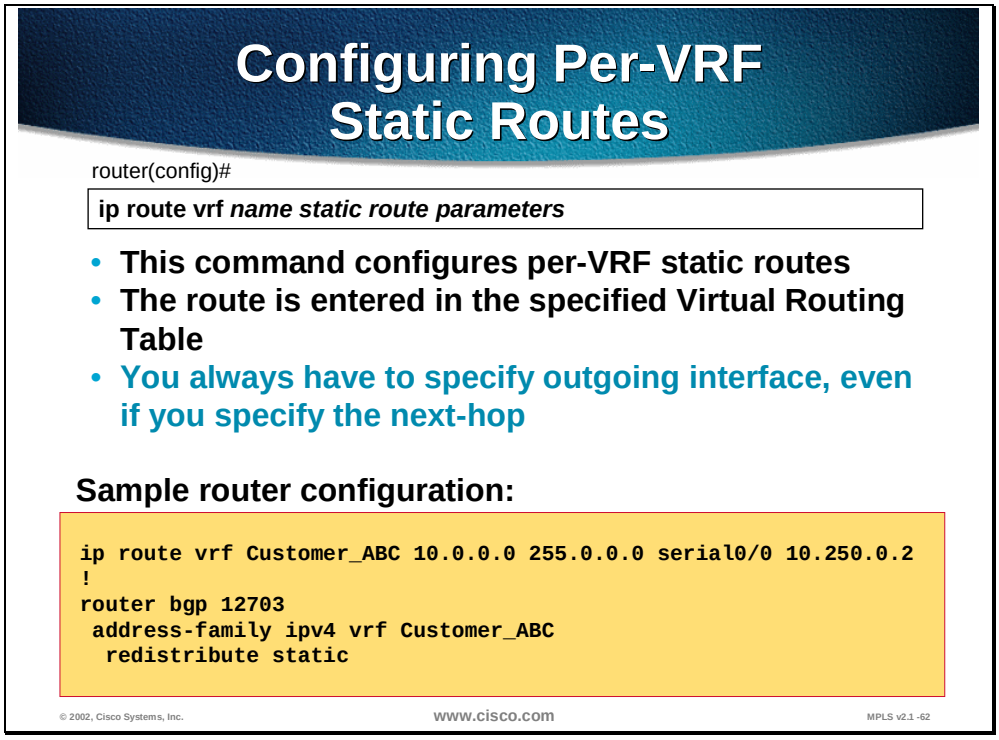
A) Use the address-family ipv4 vrf command in the OSPF configuration mode.

B) Use the router ospf pid command in the VRF configuration mode.

C) Use the ip vrf vrf command in the OSPF configuration mode.

D) Start the OSPF process by using the router ospf pid vrf vrf command.

Configuring Static VRF Routes



ip route vrf

To establish static routes for a VRF, use the **ip route vrf** command in global configuration mode. To disable static routes, use the **no** form of this command.

ip route vrf *vrf-name prefix mask* [*next-hop-address*] [*interface* {*interface-number*}] [global] [*distance*] [permanent] [tag *tag*]
no ip route vrf *vrf-name prefix mask* [*next-hop-address*] [*interface* {*interface-number*}] [global] [*distance*] [permanent] [tag *tag*]

Syntax Description

vrf-name	Name of the VPN routing/forwarding instance (VRF) for the static route.
prefix	IP route prefix for the destination in dotted-decimal format.
mask	Prefix mask for the destination in dotted-decimal format.
next-hop-address	(Optional) IP address of the next hop (the forwarding router that can be used to reach that network).
interface	Type of network interface to use.
interface-number	Number identifying the network interface to use.
global	(Optional) Specifies that the given next hop address is in the non-VRF routing table.
distance	(Optional) An administrative distance for this route.
permanent	(Optional) Specifies that this route will not be removed, even if the interface shuts down.

tag *tag*	(Optional) Label (tag) value that can be used for controlling redistribution of routes through route maps.

Practice

Q1) How do you propagate static VRF routes between PE routers?

A) All VRF routes are automatically inserted into MP-BGP.

B) By redistributing them into the IGP used between PE routers.

C) By redistributing static routes into BGP..

Summary

After completing this section, you should be able to perform the following tasks:

- Configure VRF address families in routing protocols
- Configure per-VRF BGP parameters
- Configure static routes within a VRF
- Configure per-VRF OSPF process
- Propagate RIP, OSPF, and static routes across a MP-BGP backbone

Next Steps

After completing this lesson, go to:

- Monitoring MPLS/VPN Operation

Lesson Review

Instructions

Answer the following questions:

1. How do you configure the routing context in RIP?
2. How do you configure the routing context in OSPF?
3. How many VPN OSPF processes can run simultaneously in an MPLS/VPN PE-router?
4. Where do you configure a CE EBGP neighbor?
5. How do you propagate static VRF routes between PE routers?
6. How do you propagate RIP metric across the MPLS/VPN backbone?

Monitoring MPLS/VPN Operation

Overview

This lesson describes various show commands available to monitor MPLS VPN operation.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Monitor individual VRFs and routing protocols running in them
- Monitor MP-BGP sessions between the PE routers
- Monitor inter-AS MP-BGP sessions between the PE routers
- Monitor MP-BGP table
- Monitor CEF and LFIB structures associated with MPLS/VPN

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

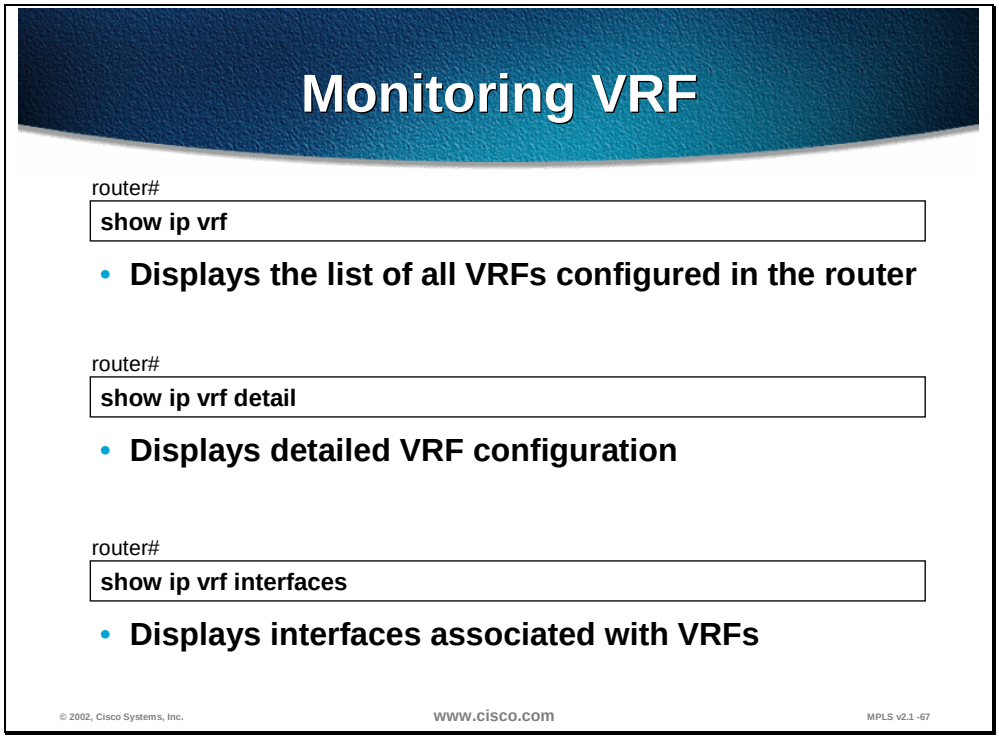
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- The Show Commands that Operate on VRFs and Sample Printouts
- The show ip bgp Commands
- The show ip bgp vpnv4 Commands
- The show ip cef Commands
- The show tag forwarding Commands
- Other vrf aware Commands
- Comprehensive Case Study: Route Propagation
- Comprehensive Case Study: Packet Forwarding
- Summary
- Lesson Review

The show Commands that Operate on VRFs and Sample Printouts



show ip vrf

To display the set of defined VRFs (VPN routing/forwarding instances) and associated interfaces, use the **show ip vrf** command in EXEC mode.

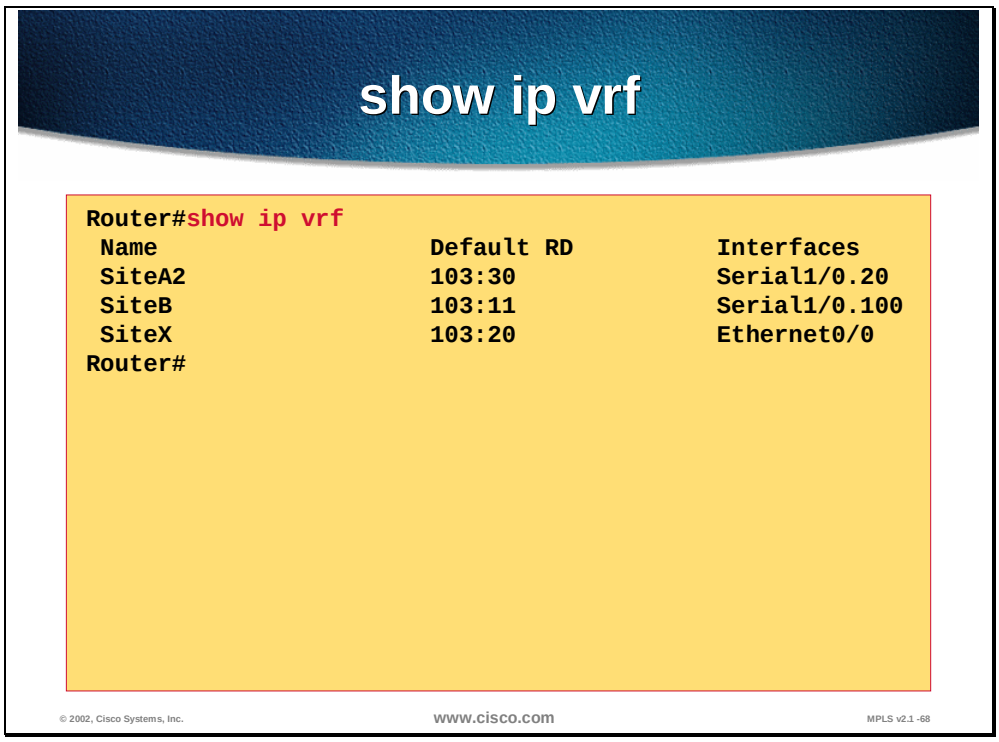
show ip vrf [{**brief** | **detail** | **interfaces**}] [*vrf-name*] [*output-modifiers*]

Syntax Description

brief	(Optional) Displays concise information on the VRF(s) and associated interfaces.
detail	(Optional) Displays detailed information on the VRF(s) and associated interfaces.
interfaces	(Optional) Displays detailed information about all interfaces bound to a particular VRF, or any VRF.
vrf-name	(Optional) Name assigned to a VRF.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.

Defaults

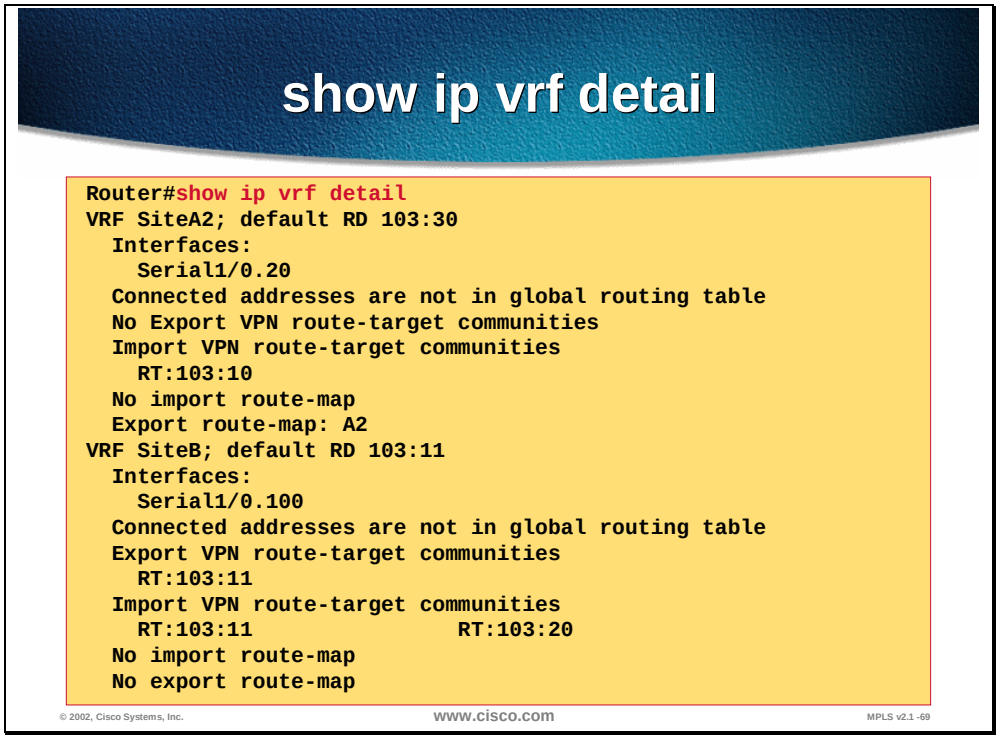
When no optional parameters are specified, the command shows concise information about all configured VRFs.



The **show ip vrf** command displays concise information on the VRF(s) and associated interfaces. The following table describes the fields displayed by this command.

Table: show ip vrf field descriptions

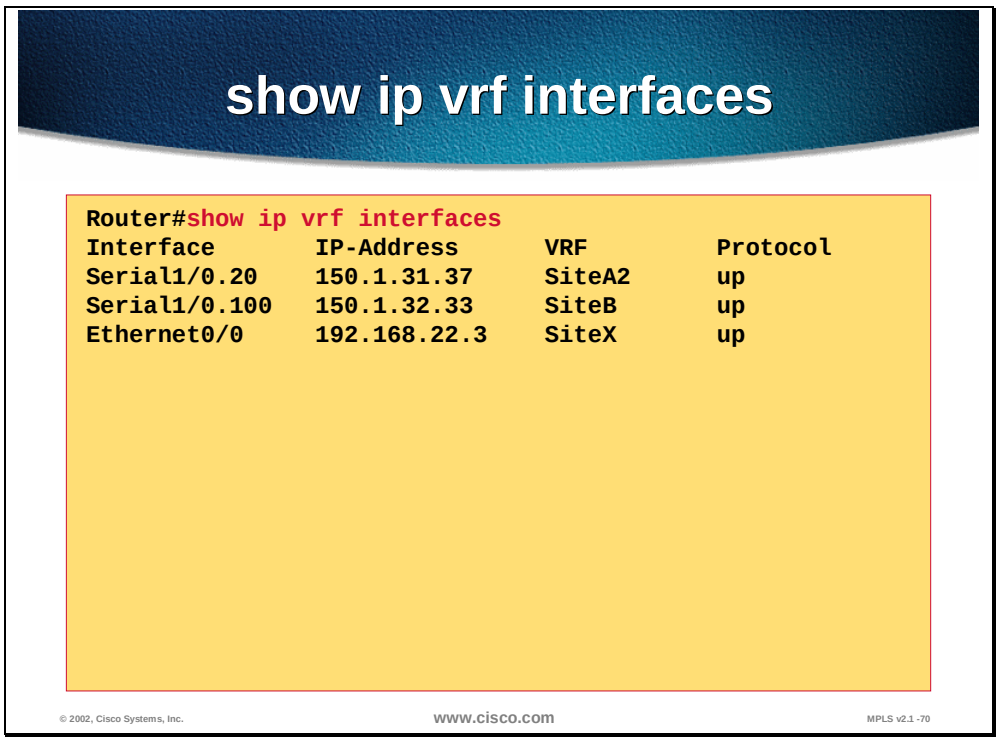
Field	Description
Name	Specifies the VRF name.
Default RD	Specifies the default route distinguisher.
Interfaces	Specifies the network interfaces.



To display detailed information on the VRFs and associated interfaces, use the **show ip vrf detail** command. The following table describes the additional fields shown by this command.

Table: show ip vrf detail Field Descriptions

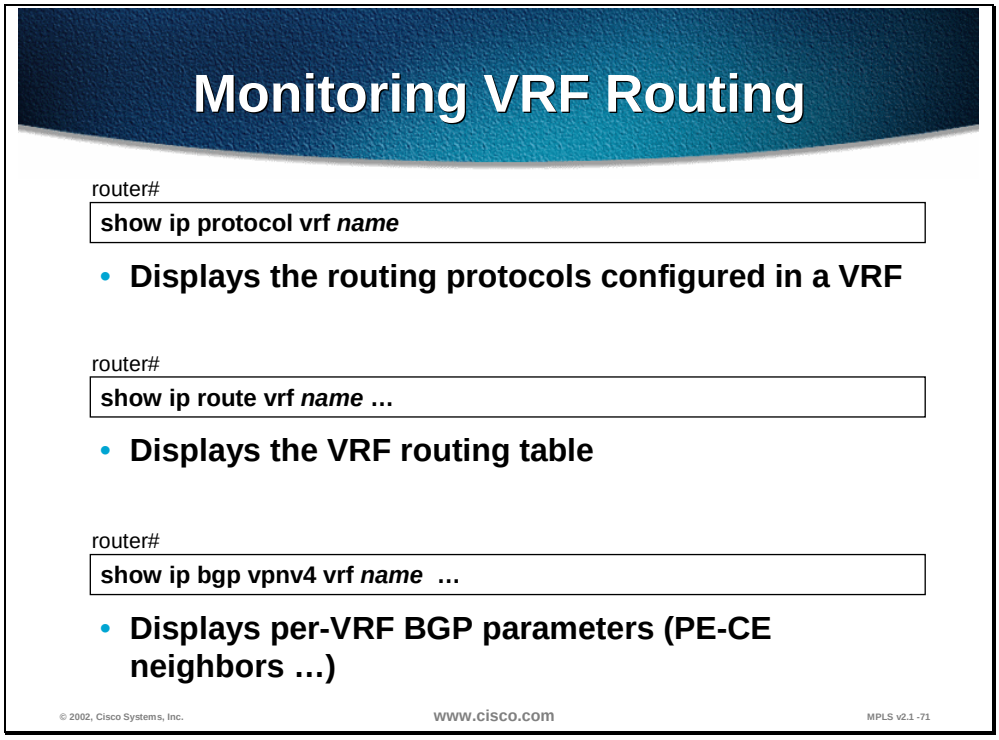
Field	Description
Interfaces	Specifies the network interfaces.
Export	Specifies VPN route-target export communities.
Import	Specifies VPN route-target import communities.



To display the interfaces bound to a particular VRF (or interfaces bound to any VRF), use the **show ip vrf interfaces** command, which displays the fields described in the following table.

Table: show ip vrf interfaces Field Descriptions

Field	Description
Interface	Specifies the network interfaces for a VRF.
IP-Address	Specifies the IP address of a VRF interface.
VRF	Specifies the VRF name.
Protocol	Displays the state of the protocol (up/down) for each VRF interface.



There are three commands that can be used to monitor VRF routing:

- **show ip protocol vrf** displays the summary information about routing protocols running in a VRF
- **show ip route vrf** displays the VRF routing table
- **show ip bgp vpnv4 vrf** displays the VRF BGP table

show ip protocols vrf

To display the routing protocol information associated with a VRF, use the **show ip protocols vrf** command in EXEC mode.

show ip protocols vrf *vrf-name*

Syntax Description

vrf-name	Name assigned to a VRF.

show ip route vrf

To display the IP routing table associated with a VRF (VPN routing/forwarding instance), use the **show ip route vrf** command in EXEC mode.

show ip route vrf *vrf-name* [connected] [*protocol* [*as-number*] [*tag*] [*output-modifiers*]] [list *number* [*output-modifiers*]] [profile] [static [*output-modifiers*]] [summary [*output-modifiers*]] [supernets-only [*output-modifiers*]] [traffic-engineering [*output-modifiers*]]

Syntax Description

vrf-name	Name assigned to the VRF.
connected	(Optional) Displays all connected routes in a VRF.
protocol	(Optional) To specify a routing protocol, use one of the following keywords: bgp, egp, eigrp, hello, igrp, isis, ospf, or rip.
as-number	(Optional) Autonomous system number.
tag	(Optional) IOS routing area label.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.
list *number*	(Optional) Specifies the IP access list to display.
profile	(Optional) Displays the IP routing table profile.
static	(Optional) Displays static routes.
summary	(Optional) Displays a summary of routes.
supernets-only	(Optional) Displays supernet entries only.
traffic-engineering	(Optional) Displays only traffic-engineered routes.

show ip bgp vpnv4

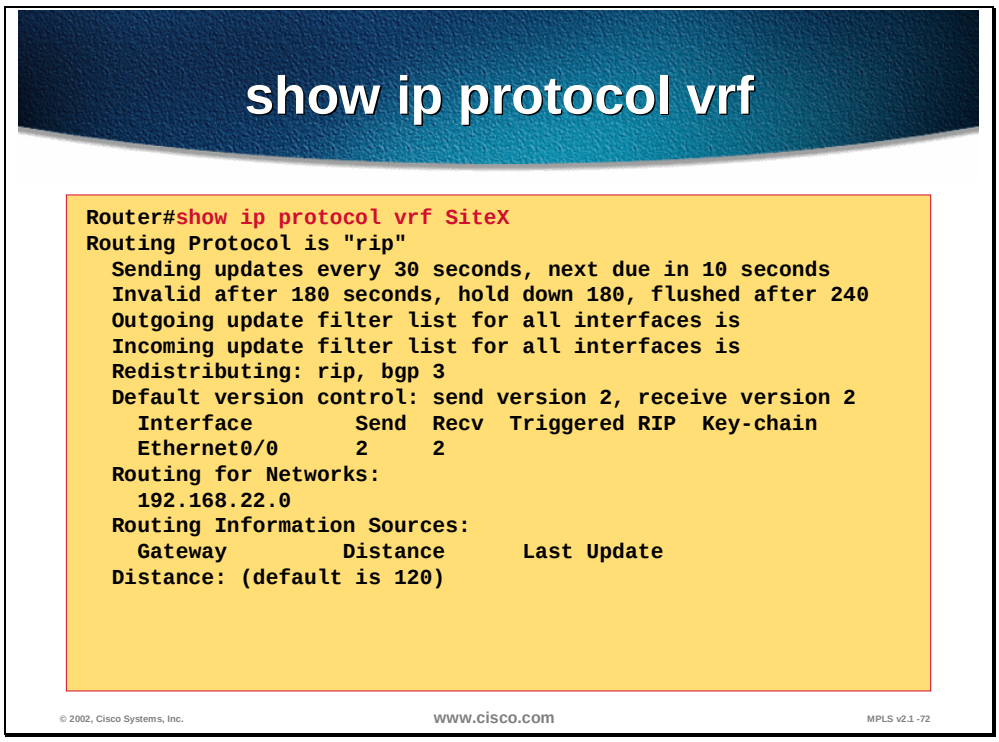
To display VPN address information from the BGP table, use the **show ip bgp vpnv4** command in EXEC mode.

show ip bgp vpnv4 {all | rd *route-distinguisher* | vrf *vrf-name*} [*ip-prefix/length* [**longer-prefixes**] [*output-modifiers*]] [*network-address* [*mask*] [**longer-prefixes**] [*output-modifiers*]] [cidr-only] [community] [community-list] [dampened-paths] [filter-list] [flap-statistics] [inconsistent-as][neighbors] [paths [*line*]] [peer-group] [quote-regexp] [regexp] [summary] [tags]

Syntax Description

all	Displays the complete VPNv4 database.
rd *route-distinguisher*	Displays NLRIs that have a matching route distinguisher.
vrf *vrf-name*	Displays NLRIs associated with the named VRF.
ip-prefix/length	(Optional) IP prefix address (in dotted decimal format) and length of mask (0 to 32).
longer-prefixes	(Optional) Displays the entry, if any, that exactly matches the specified prefix parameter, as well as all entries that match the prefix in a "longest-match" sense. That is,

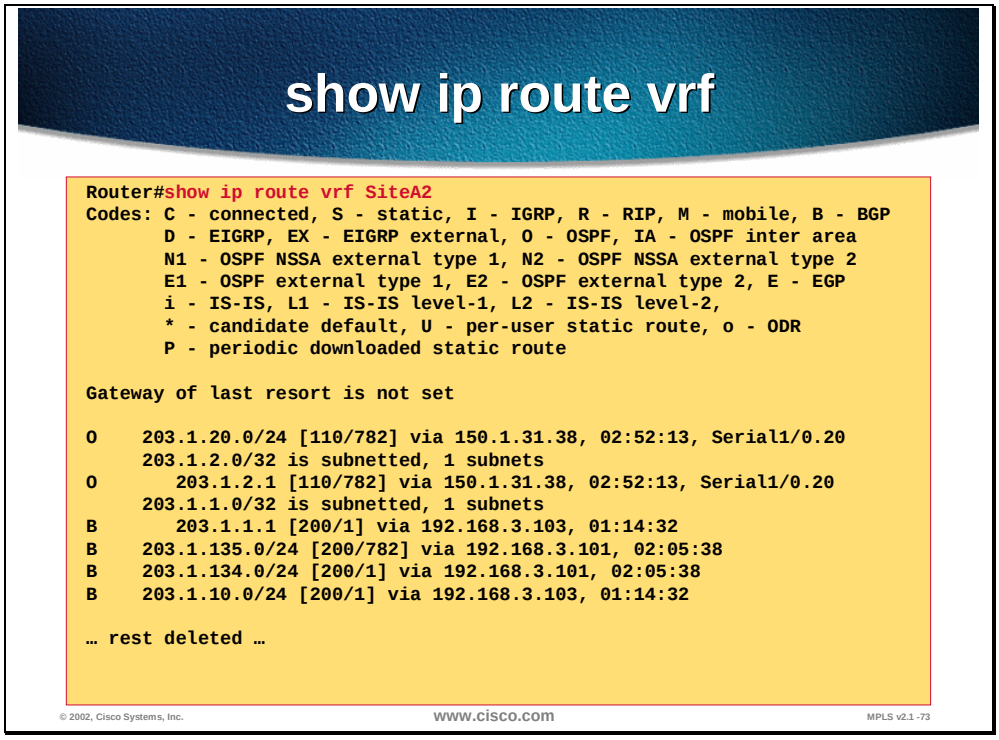
	prefixes for which the specified prefix is an initial substring.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.
network-address	(Optional) IP address of a network in the BGP routing table.
mask	(Optional) Mask of the network address, in dotted decimal format.
cidr-only	(Optional) Displays only routes that have non-natural net masks.
community	(Optional) Displays routes matching this community.
community-list	(Optional) Displays routes matching this community list.
dampened-paths	(Optional) Displays paths suppressed on account of dampening (BGP route from peer is up and down).
filter-list	(Optional) Displays routes conforming to the filter list.
flap-statistics	(Optional) Displays flap statistics of routes.
inconsistent-as	(Optional) Displays only routes that have inconsistent autonomous systems of origin.
neighbors	(Optional) Displays details about TCP and BGP neighbor connections.
paths	(Optional) Displays path information.
line	(Optional) A regular expression to match the BGP AS paths.
peer-group	(Optional) Displays information about peer groups.
quote-regexp	(Optional) Displays routes matching the AS path "regular expression."
regexp	(Optional) Displays routes matching the AS path “regular expression.”
summary	(Optional) Displays BGP neighbor status.
tags	(Optional) Displays incoming and outgoing BGP labels for each NLRI.



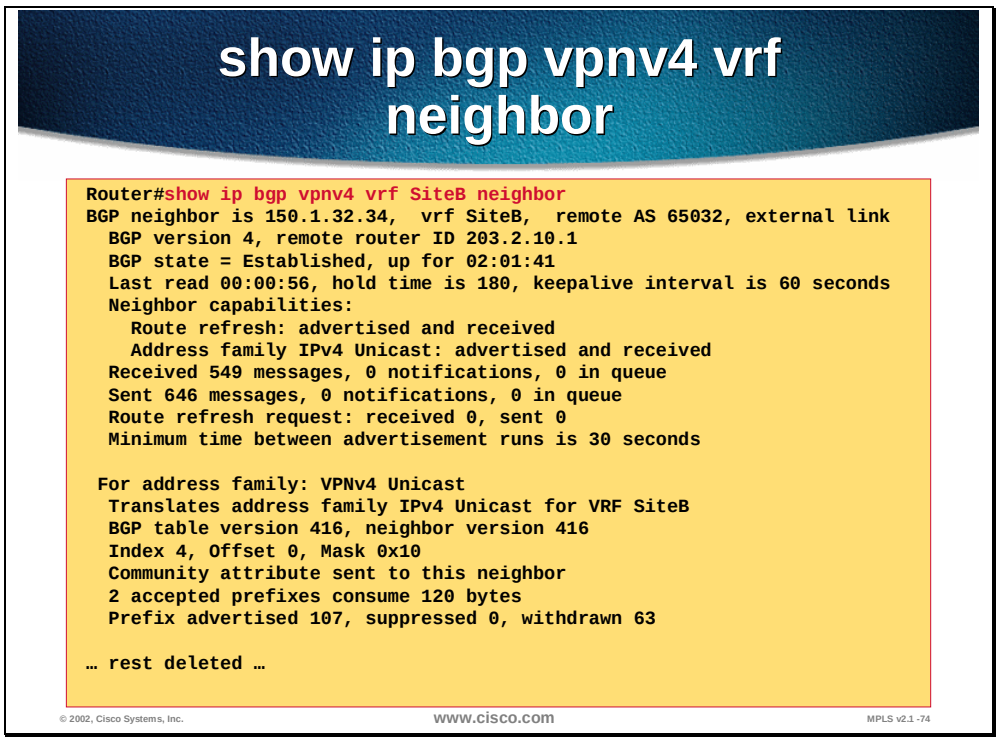
The **show ip protocol vrf** command displays summary information about all routing protocol instances active in the specified VRF. The fields displayed by this command are shown in the following table.

Table: show ip protocols vrf Field Descriptions

Field	Description
Gateway	Displays the IP address of the router identifier for all routers in the network.
Distance	Displays the metric used to access the destination route.
Last update	Displays the last time the routing table was updated from the source.



The **show ip route vrf** command displays the contents of the VRF IP routing table in the same format as used by the **show ip route** command.



show ip bgp vpnv4 neighbors

To display BGP neighbors configured in a VRF, use the **show ip bgp vpnv4 vrf neighbors** privileged EXEC command.

show ip bgp vpnv4 {all | vrf *vrf-name***} neighbors**

Syntax Description

vpnv4	Specifies VPN IPv4 information.
all	Displays all VPN BGP neighbors
vrf *vrf-name*	Displays neighbors associated with the named VRF.
neighbors	Displays details on TCP and BGP neighbor connections.

Defaults

This command has no default values.

Usage Guidelines

Use this command to display detailed information about BGP neighbors associated with MPLS VPN.

Practice

Q1) How would you verify the contents of a VRF routing table?

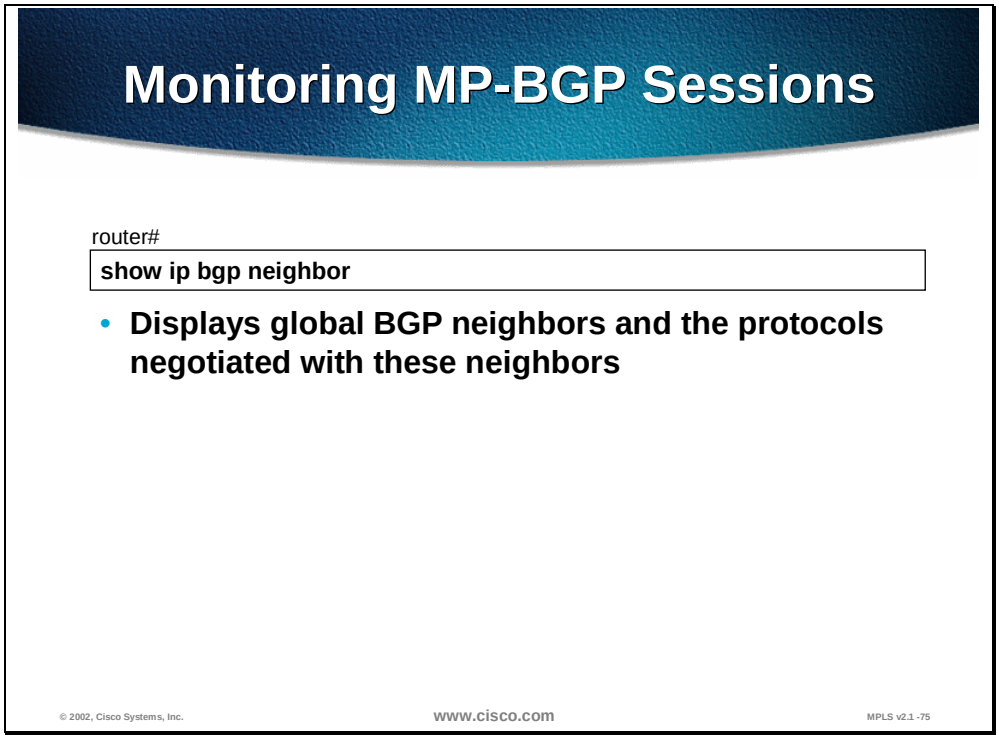
A) show vrf vrf-name routing

B) show ip route vrf vrf-name

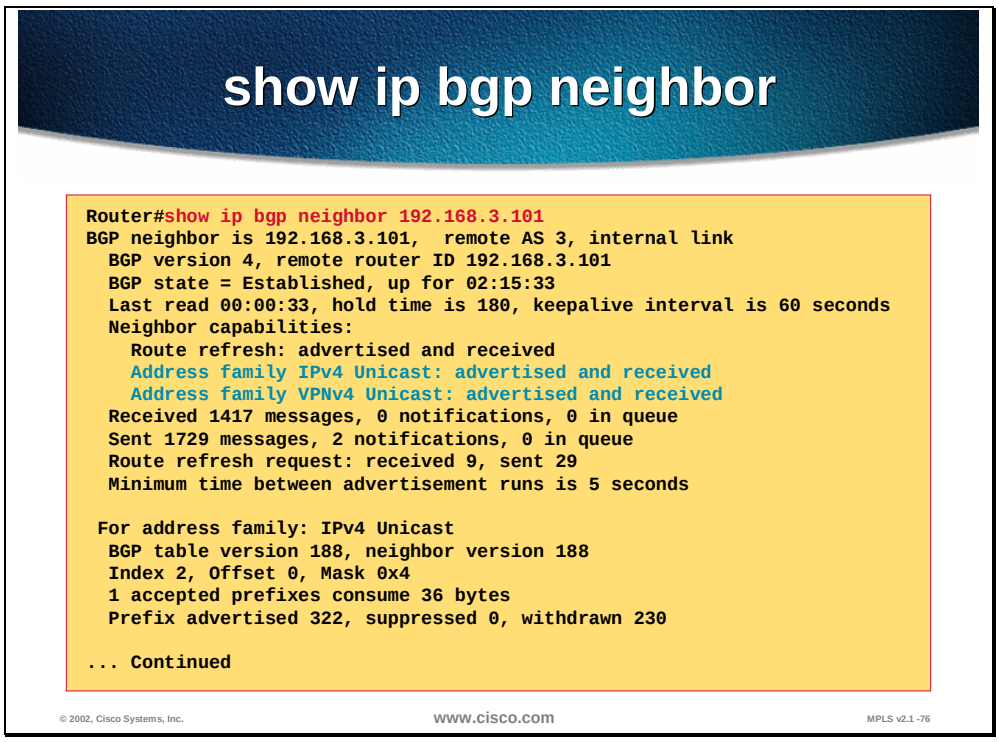
C) show vrf vrf-name ip route

D) The conent of the VRF routing tables cannot be verfied

The show ip bgp Commands



The **show ip bgp neighbor** command, described in details in the **Basic BGP Technology and Configuration** lesson is also used to monitor BGP sessions with other PE routers as well as the address families negotiated with these neighbors.



show ip bgp neighbors

To display information about the TCP and Border Gateway Protocol (BGP) connections to neighbors, use the **show ip bgp neighbors** EXEC command.

show ip bgp neighbors [*address*] [**received-routes** | **routes** | **advertised-routes** | {**paths** *regular-expression*} | **dampened-routes**]

Syntax Description

address	(Optional) Address of the neighbor whose routes you have learned from. If you omit this argument, all neighbors are displayed.
received-routes	(Optional) Displays all received routes (both accepted and rejected) from the specified neighbor.
routes	(Optional) Displays all routes that are received and accepted. This is a subset of the output from the received-routes keyword.
advertised-routes	(Optional) Displays all the routes the router has advertised to the neighbor.
paths *regular-expression*	(Optional) Regular expression that is used to match the paths received.
dampened-routes	(Optional) Displays the dampened routes to the neighbor at the IP address specified.

Examples

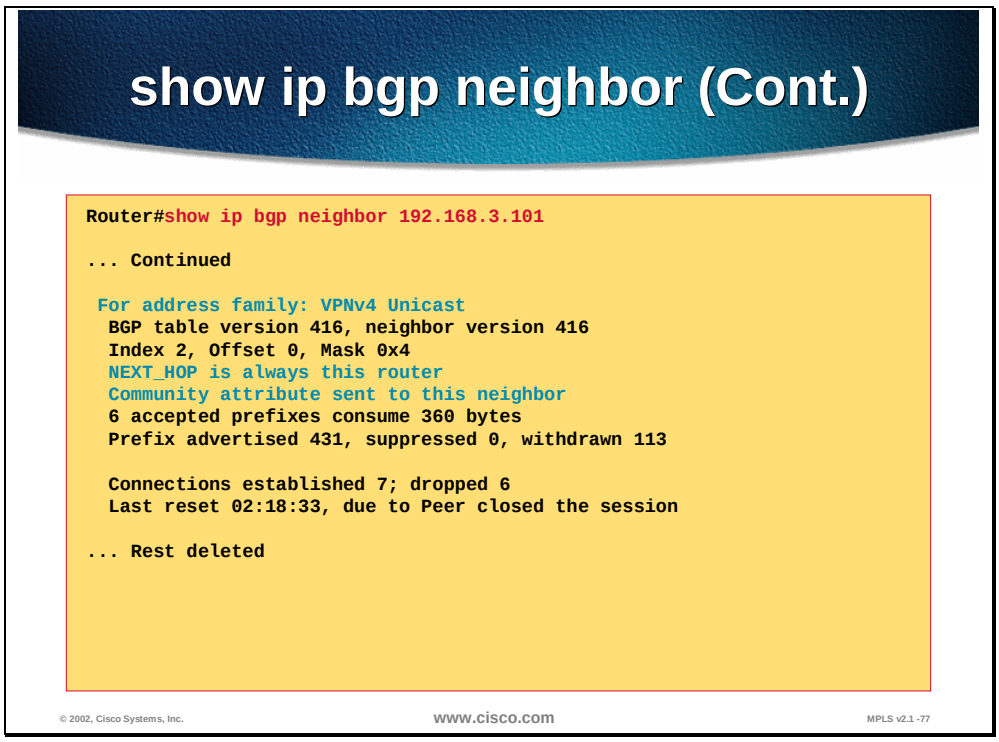
The following is sample output from the **show ip bgp neighbors** command:

```
Router# show ip bgp neighbors 171.69.232.178

BGP neighbor is 171.69.232.178,  remote AS 10, external link
 Index 1, Offset 0, Mask 0x2
  Inbound soft reconfiguration allowed
  BGP version 4, remote router ID 171.69.232.178
  BGP state = Established, table version = 27, up for
00:06:12
  Last read 00:00:12, hold time is 180, keepalive interval
is 60 seconds
  Minimum time between advertisement runs is 30 seconds
  Received 19 messages, 0 notifications, 0 in queue
  Sent 17 messages, 0 notifications, 0 in queue
  Inbound path policy configured
  Route map for incoming advertisements is testing
  Connections established 2; dropped 1
Connection state is ESTAB, I/O status: 1, unread input
bytes: 0
Local host: 171.69.232.181, Local port: 11002
Foreign host: 171.69.232.178, Foreign port: 179

Enqueued packets for retransmit: 0, input: 0, saved: 0

Event Timers (current time is 0x530C294):
Timer          Starts    Wakeups            Next
Retrans            12          0             0x0
TimeWait            0          0             0x0
AckHold            12         10             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0

iss:  133981889  snduna:  133982166  sndnxt:  133982166
sndwnd:  16108
irs: 3317025518  rcvnxt: 3317025810  rcvwnd:      16093
delrcvwnd:    291

SRTT: 441 ms, RTTO: 2784 ms, RTV: 951 ms, KRTT: 0 ms
minRTT: 0 ms, maxRTT: 300 ms, ACK hold: 300 ms
Flags: higher precedence, nagle

Datagrams (max data segment is 1460 bytes):
Rcvd: 15 (out of order: 0), with data: 12, total data bytes:
291
Sent: 23 (retransmit: 0), with data: 11, total data bytes:
276
```

The following table describes the fields shown in the display.

Field	Description
BGP neighbor	IP address of the BGP neighbor and its autonomous system number. If the neighbor is in the same autonomous system as the router, then the link between them is internal; otherwise, it is considered external.
BGP version	BGP version being used to communicate with the remote router; the neighbor's router ID (an IP address) is also specified.
BGP state	Internal state of this BGP connection.
table version	Indicates that the neighbor has been updated with this version of the primary BGP routing table.
up for	Amount of time that the underlying TCP connection has been in existence.
Last read	Time that BGP last read a message from this neighbor.
hold time	Maximum amount of time that can elapse between messages from the peer.
keepalive interval	Time period between sending keepalive packets, which help ensure that the TCP connection is up.
Received	Number of total BGP messages received from this peer, including keepalives.
notifications	Number of error messages received from the peer.
Sent	Total number of BGP messages that have been sent to this peer, including keepalives.
notifications	Number of error messages the router has sent to this peer.
Connections established	Number of times the router has established a TCP connection and the two peers have agreed speak BGP with each other.
dropped	Number of times that a good connection has failed or been taken down.
Connection state	State of BGP peer.
unread input bytes	Number of bytes of packets still to be processed.
Local host, Local port	Peering address of local router, plus port.
Foreign host, Foreign port	Neighbor's peering address.
Event Timers	Table displays the number of starts and wakeups for each timer.
iss	Initial send sequence number.
snduna	Last send sequence number the local host sent but has not received an acknowledgment for.
sndnxt	Sequence number the local host will send next.
sndwnd	TCP window size of the remote host.
irs	Initial receive sequence number.
rcvnxt	Last receive sequence number the local host has acknowledged.
rcvwnd	Local host's TCP window size.
delrecvwnd	Delayed receive window---data the local host has read from the connection, but has not yet subtracted from the receive window the host has advertised to the remote host. The value in this field gradually increases until it is larger than a full-sized packet, at which point it is applied to the rcvwnd field.
SRTT	A calculated smoothed round-trip timeout.
RTTO	Round-trip timeout.
RTV	Variance of the round-trip time.
KRTT	New round-trip timeout (using the Karn algorithm). This field separately tracks the round-trip time of packets that have been retransmitted.

Field	Description
minRTT	Smallest recorded round-trip timeout (hard wire value used for calculation).
maxRTT	Largest recorded round-trip timeout.
ACK hold	Time the local host will delay an acknowledgment in order to piggyback data on it.
Flags	IP precedence of the BGP packets.
Datagrams: Rcvd	Number of update packets received from neighbor.
with data	Number of update packets received with data.
total data bytes	Total bytes of data.
Sent	Number of update packets sent.
with data	Number of update packets with data sent.
total data bytes	Total number of data bytes.



The **show ip bgp neighbor** command displays per address-family information for neighbors that exchange MP-BGP updates with this router. The most interesting details of the printout produced by this command are highlighted in blue color in the example above.

Practice

Q1) How would you verify an VPNv4 information exchange with a MP-BGP neighbor?

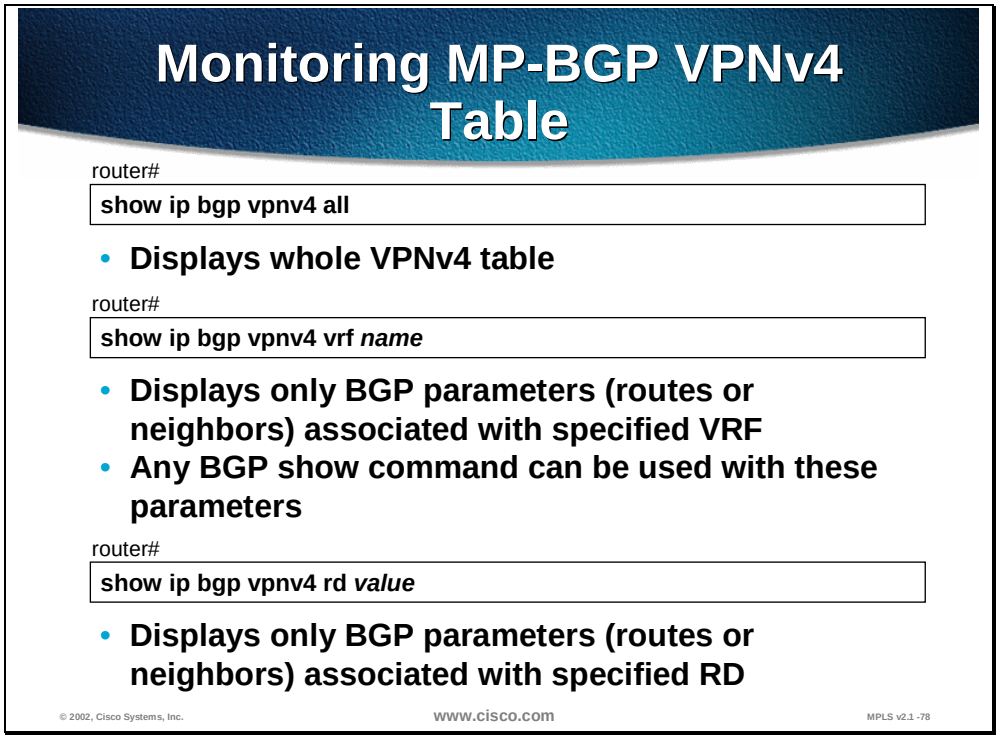
A) show bgp vpnv4

B) show bgp exchange

C) show ip bgp neighbor

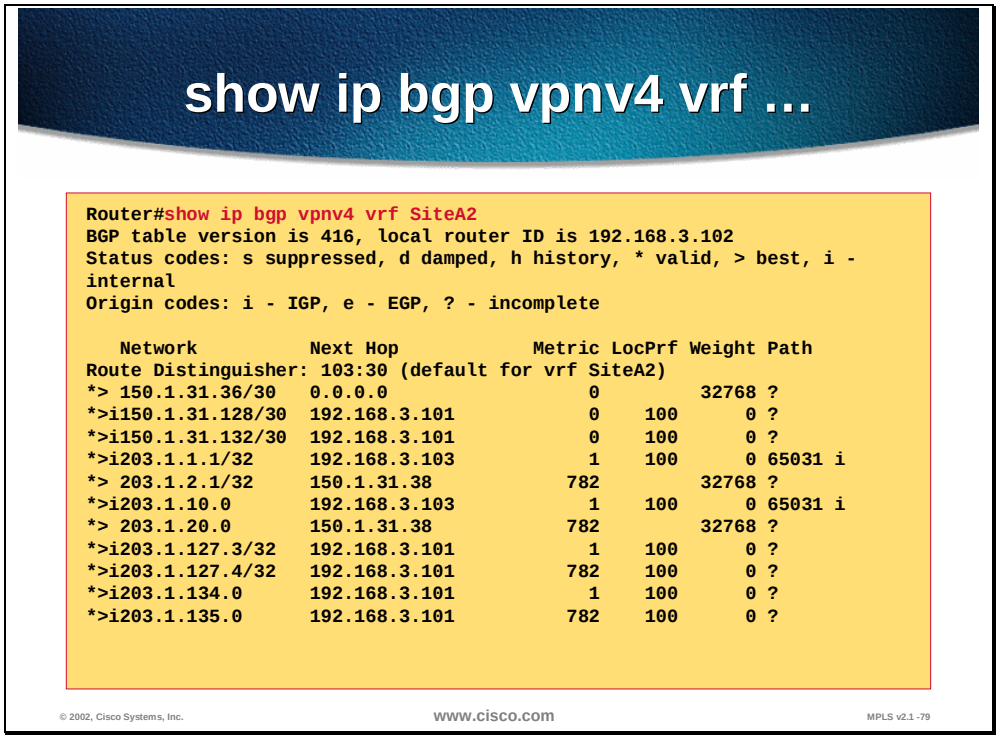
D) show bgp neighbor vpnv4

The show ip bgp vpnv4 Commands



The **show ip bgp** command is used to display IPv4 BGP information as well as VPNv4 BGP information. To display VPNv4 BGP information, use the **vpnv4** keyword followed by one of these keywords:

- **all** to display the whole contents of VPNv4 BGP table
- **vrf *name*** to display VPNv4 information associated with the specified VRF
- **rd *value*** to display VPNv4 information associated with the specified route distinguisher.



show ip bgp vpnv4 vrf *name*

To display VPNv4 information from the BGP database associated with a VRF, use the **show ip bgp vpnv4 vrf *name*** privileged EXEC command.

show ip bgp vpnv4 vrf *vrf-name* [*ip-prefix/length* **[longer-prefixes]** [*output-modifiers*]] [*network-address* [*mask*] **[longer-prefixes]** [*output-modifiers*]] **[cidr-only] [community][community-list] [dampened-paths] [filter-list] [flap-statistics] [inconsistent-as] [neighbors] [paths** [*line*]**] [peer-group] [quote-regexp] [regexp] [summary] [tags]**

Syntax Description

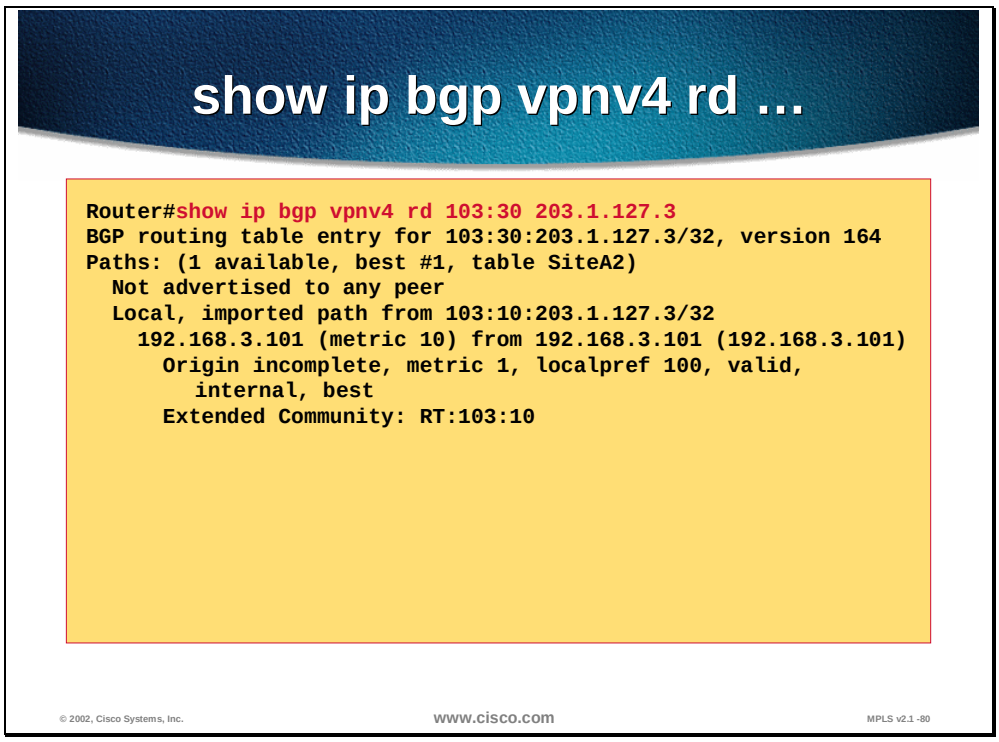
vrf *vrf-name*	Displays NLRIs associated with the named VRF.

Defaults

This command has no default values.

Usage Guidelines

Use this command to display VPNv4 information associated with a VRF from the BGP database. A similar command – **show ip bgp vpnv4 all** – displays all available VPNv4 information. The command **show ip bgp vpnv4 summary** displays BGP neighbor status.



show ip bgp vpnv4 rd *value*

To display all VPNv4 routes that contain specified route distinguisher, use the **show ip bgp vpnv4 rd** privileged EXEC command.

show ip bgp vpnv4 rd *route-distinguisher* [*ip-prefix/length* **[longer-prefixes]** [*output-modifiers*]] [*network-address* [*mask*] **[longer-prefixes]** [*output-modifiers*]] **[cidr-only] [community][community-list] [dampened-paths] [filter-list] [flap-statistics] [inconsistent-as] [paths** [*line*]**] [quote-regexp] [regexp] [summary] [tags]**

Syntax Description

rd *route-distinguisher* Displays NLRIs that have a matching route distinguisher.

Defaults

This command has no default values.

Usage Guidelines

Use this command to display VPNv4 information associated with a VRF from the BGP database. A similar command – **show ip bgp vpnv4 all** – displays all available VPNv4 information. The command **show ip bgp vpnv4 summary** displays BGP neighbor status.

Practice

Q1) How would you display all routes with a specified route distinguisher?

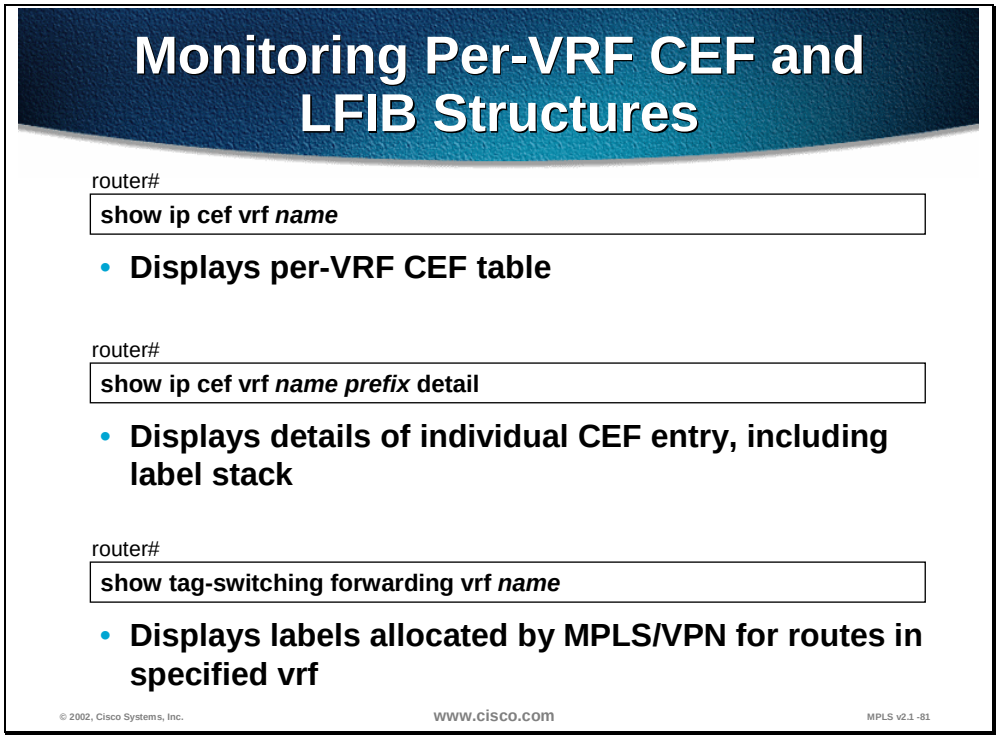
A) show ip bgp route-distinguisher route-distinguiser

B) show ip bgp rd route-distinguiser

C) show ip route vpnv4 rd route-distinguiser

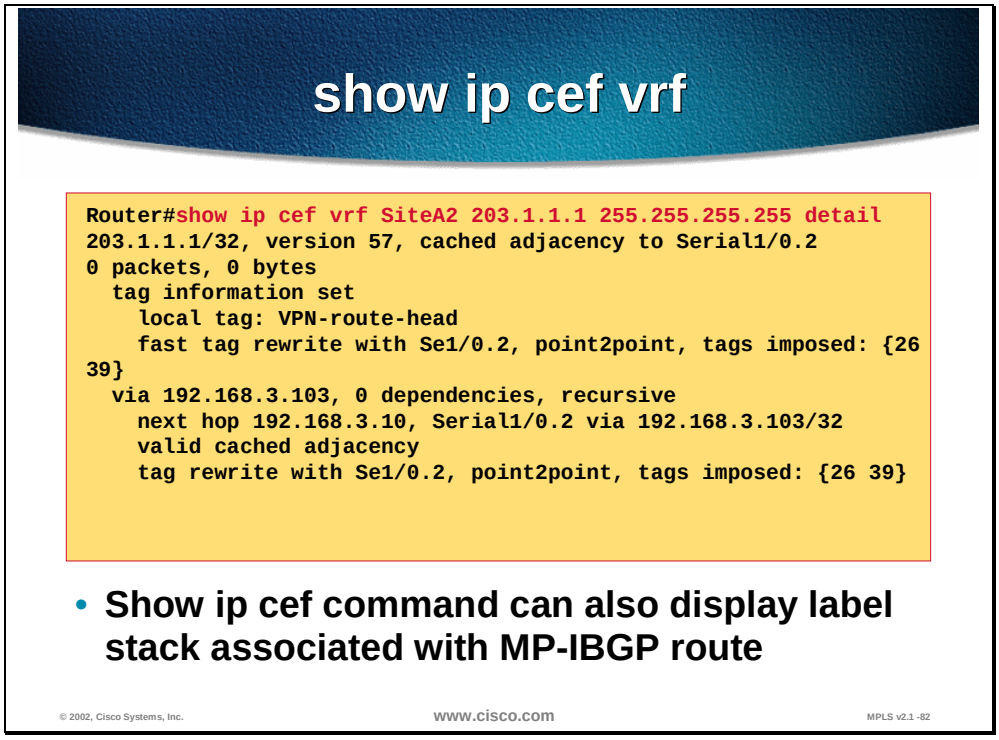
D) show ip bgp vpnv4 rd route-distinguiser

The show ip cef Commands



There are three commands that can be used to display per-VRF FIB and LFIB structures:

- **show ip cef vrf** command displays the VRF Forwarding Information Base
- **show ip cef vrf detail** command displays detailed information about a single entry in the VRF FIB
- **show tag-switching forwarding vrf** command displays all labels allocated to VPN routes in the specified VRF.



show ip cef vrf

To display the CEF forwarding table associated with a VRF, use the **show ip cef vrf** privileged EXEC command.

show ip cef vrf *vrf-name* **[***ip-prefix* **[***mask* **[longer-prefixes]] [detail] [***output-modifiers***]] [***interface interface-number***] [adjacency [***interface interface-number***] [detail] [discard] [drop] [glean] [null] [punt] [***output-modifiers***]] [detail [***output-modifiers***]] [non-recursive [detail] [***output-modifiers***]] [summary [***output-modifiers***]] [traffic [prefix-length] [***output-modifiers***]] [unresolved [detail] [***output-modifiers***]]**

Syntax Description

vrf-name	Name assigned to the VRF.
ip-prefix	(Optional) IP prefix of entries to show, in dotted decimal format (A.B.C.D).
mask	(Optional) Mask of the IP prefix in dotted decimal format.
longer-prefixes	(Optional) Displays table entries for all of the more specific routes.
detail	(Optional) Displays detailed information for each CEF table entry.
output-modifiers	(Optional)
interface	(Optional) Type of network interface to use: **ATM, Ethernet**, **Loopback**, **POS** (packet over SONET) or **Null**.
interface-number	Number identifying the network interface to use.
adjacency	(Optional) Displays all prefixes resolving through adjacency.

discard	Discards adjacency.
drop	Drops adjacency.
glean	Gleans adjacency.
null	Null adjacency.
punt	Punts adjacency.
non-recursive	(Optional) Displays only non-recursive routes.
summary	(Optional) Displays a CEF table summary.
traffic	(Optional) Displays traffic statistics.
prefix-length	(Optional) Displays traffic statistics by prefix size.
unresolved	(Optional) Displays only unresolved routes.

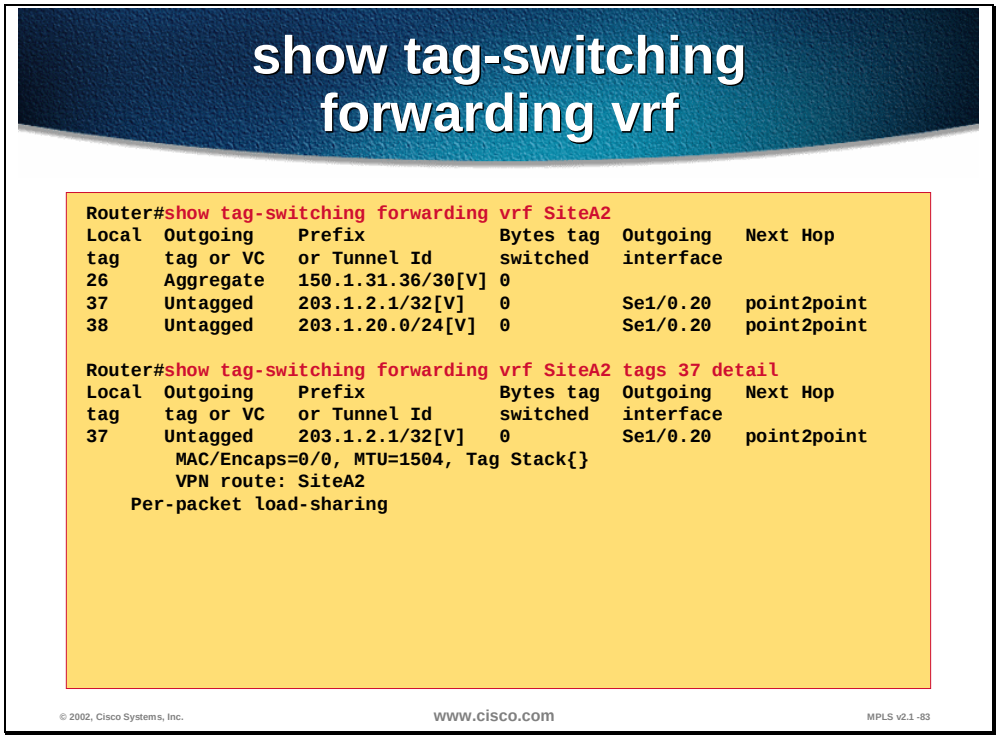
Defaults

This command has no default values.

Usage Guidelines

Used with the *vrf-name* argument, the **show ip cef vrf** command shows a shortened display of the CEF table.
Used with the **detail** argument, the **show ip cef vrf** command shows detailed information for all CEF table entries.

The show tag forwarding Commands



show tag-switching forwarding vrf

To display label forwarding information for advertised VRF routes, use the **show tag-switching forwarding vrf** command in EXEC mode. To disable the display of label forwarding information, use the **no** form of this command.

show tag-switching forwarding vrf *vrf-name* [*ip-prefix/length* [*mask*]] [detail] [*output-modifiers*]

Syntax Description

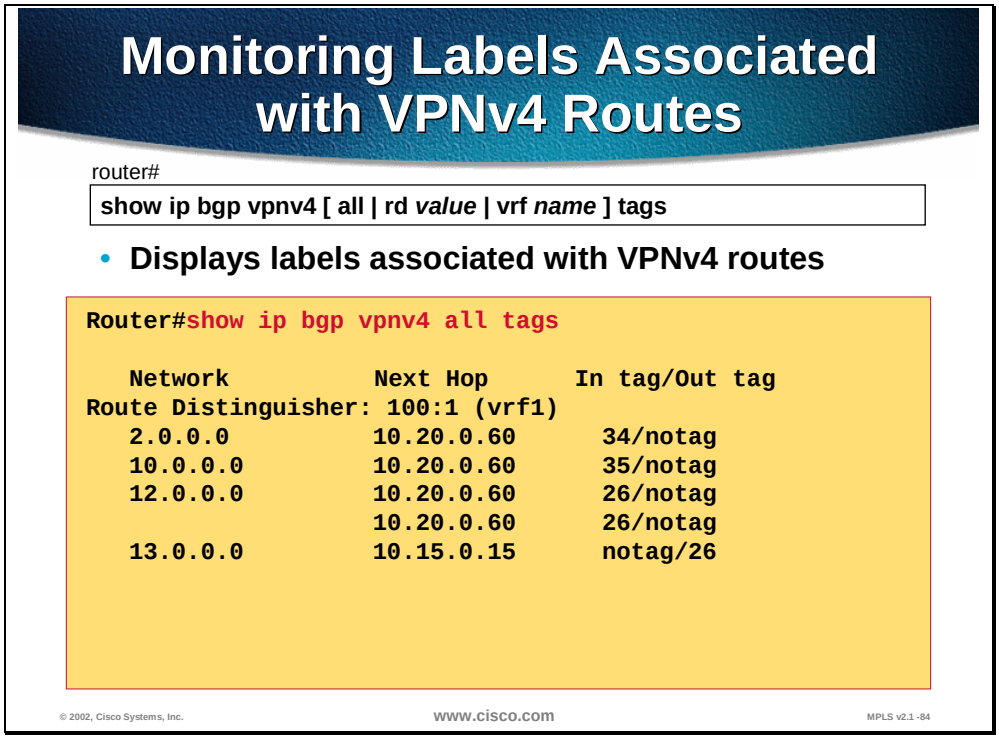
vrf-name	Displays NLRIs associated with the named VRF.
ip-prefix/length	(Optional) IP prefix address (in dotted decimal format) and length of mask (0 to 32).
mask	(Optional) Destination network mask in dotted decimal format.
detail	(Optional) Displays detailed information on the VRF routes.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.

Defaults

No default behavior or values.

Usage Guidelines

Use this command to display label forwarding entries associated with a particular VRF or IP prefix.

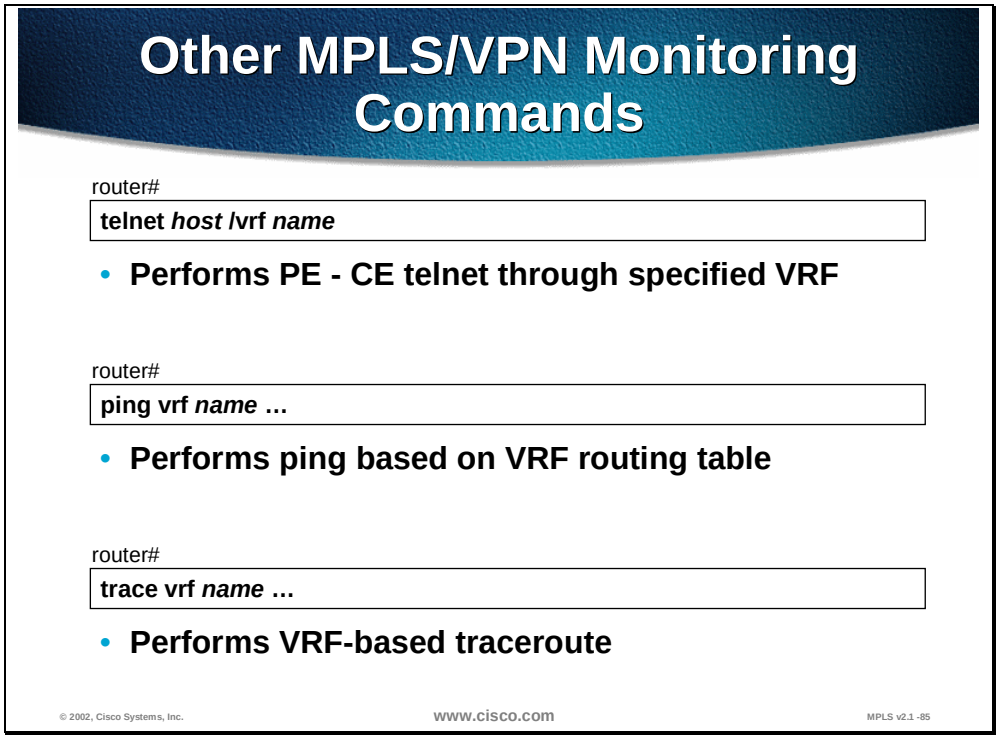


The **show ip bgp vnpv4 tags** command can be used to display tags assigned to local or remote VRF routes by the local or remote PE router. The command displays tags associated with all VPNv4 routes (when using **all** keyword) or tags associated with a specified route distinguisher or VRF.

The following fields are displayed in the printout:

Field	Description
Network	Displays the network address from the BGP table.
Next Hop	Specifies the BGP next hop address.
In Tag	Displays the label (if any) assigned by this router.
Out Tag	Displays the label assigned by the BGP next hop router.

Other vrf Aware Commands



Three additional IOS monitoring commands are VRF-aware:

- **telnet** command can be used to connect to a CE router from a PE router using the **/vrf** option
- **ping vrf** command can be used to ping a destination host reachable through a VRF
- **trace vrf** command can be used to trace a path toward a destination reachable through a VRF.

Practice

Q1) When doing show tag forwarding-table vrf name detail, why do you only see labels for routes learned from CE routers?

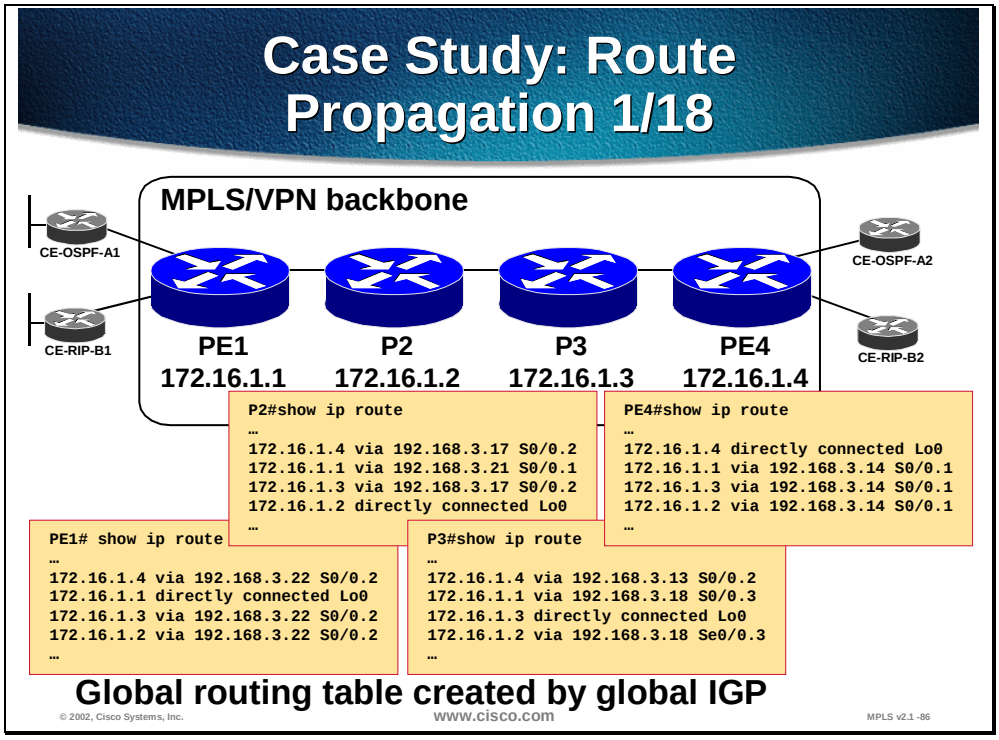
A) A local label is assigned to local customer routes to allow label-switching of packets towards the CE router. But routes received by MP-BGP from other PE routers will always be IP routed resulting in a tag-imposition.

B) There will be labels for all routes. Not just for routes received from CE routers.

C) Label assignment is controlled by the customer.

D) CE routers always use EBGP which results in label assignment, while other PE routers always use MP-BGP.

Comprehensive Case Study: Route Propagation



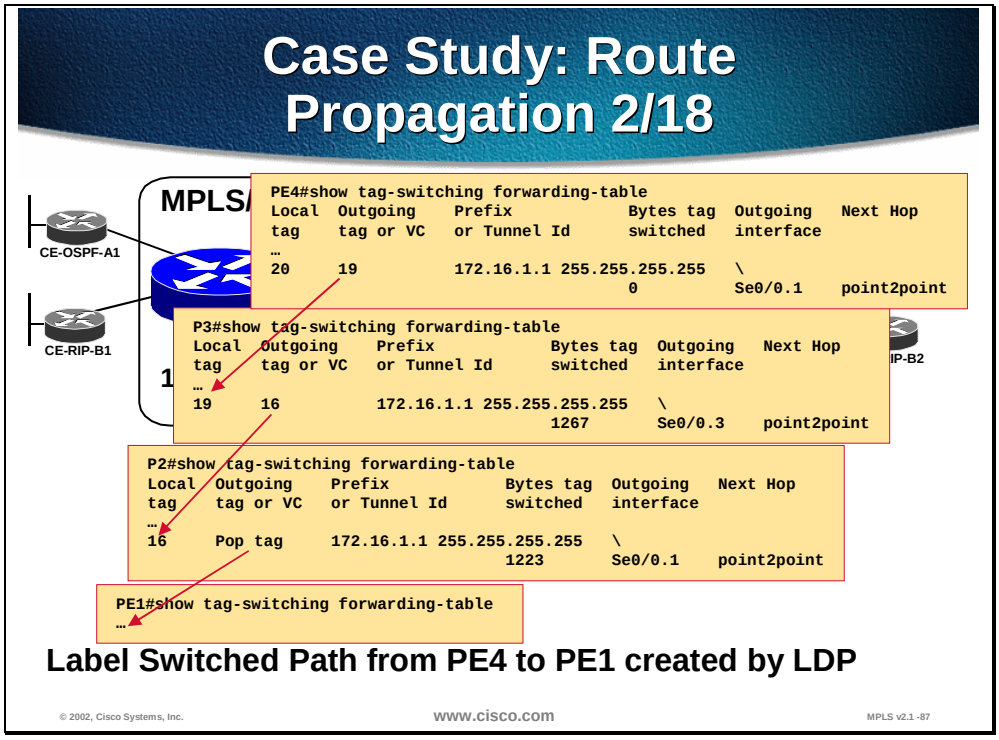
The case study, which is described on the following pages, is a comprehensive explanation of the features and mechanisms involved in MPLS/VPN. In addition to that, several monitoring commands (show commands) are used to display the content in various tables. These can be used in troubleshooting situations to verify a function or find which of the functions that is failing.

This first figure illustrates how the IGP inside the P network (MPLS/VPN backbone) find all the P and PE routers. The IGP should, in the ideal case, not carry any other routes than loopback addresses of P and PE routers and the links in between those routers.

The IGP information is essential for the PE routers to exchange packets. Reachability between the PE routers loopback interfaces is a requirement for the MPLS/VPN backbone to operate.

The command **show ip route** (without any references to any vrf) is used to display this information.

In this particular example, the focus will be on how PE4 can reach the loopback interface in PE1, 172.16.1.1.



The LDP (or TDP) protocol is used to establish label switched paths between all PE routers. In this figure the LSP from PE4 to the loopback interface in PE1 is illustrated.

The P and PE routers must exchange the LDP (or TDP) protocol. Each IGP route will be assigned a label. This label is propagated to each neighbor. The labels used for forwarding are stored in the LFIB. This is displayed by the **show tag-switching forwarding-table**

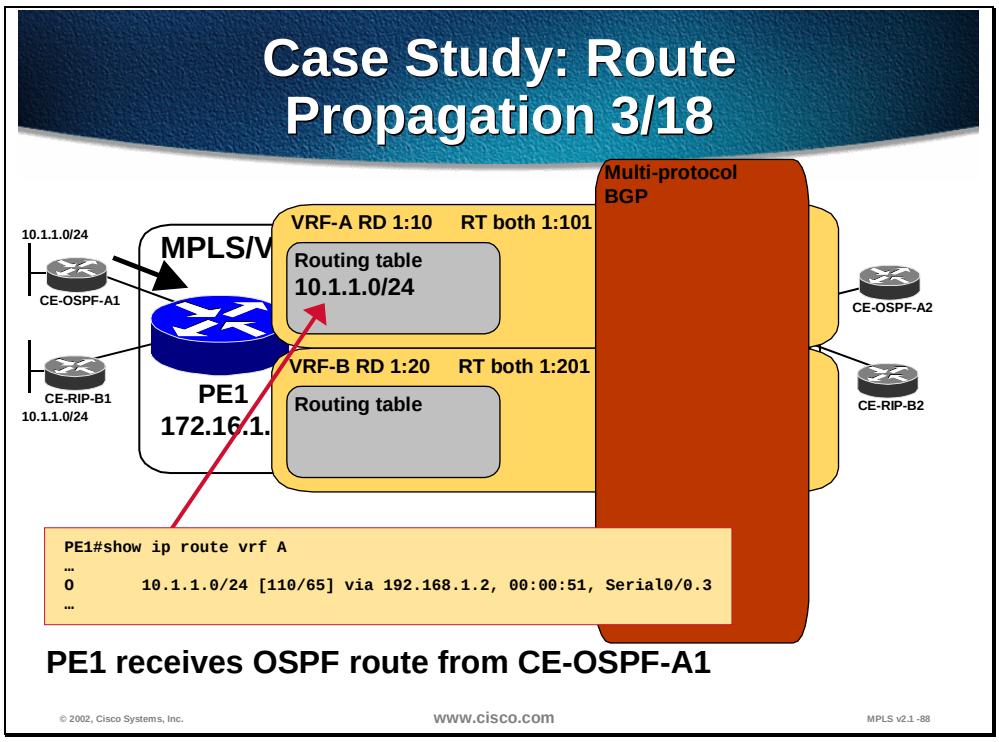
The LSP from PE4 via P3 and P2 to PE1 can be manually traced by starting in PE4 and check which label PE4 will use and which route PE4 will use as next hop. PE4 will use label value 19 and forward the packets to P3.

The focus is moved to P3, the next hop router. The same command is given there, but now the search is for what P3 will do with the incoming label value 19. It turns out that P3 will forward those packets to P2 but with the label swapped to label value 16.

So the focus moves again, now to P2. P2 will take incoming packets with label value 16 and forward to PE1. But as P2 is doing this packet forward, the top most label is removed, penultimate hop popping.

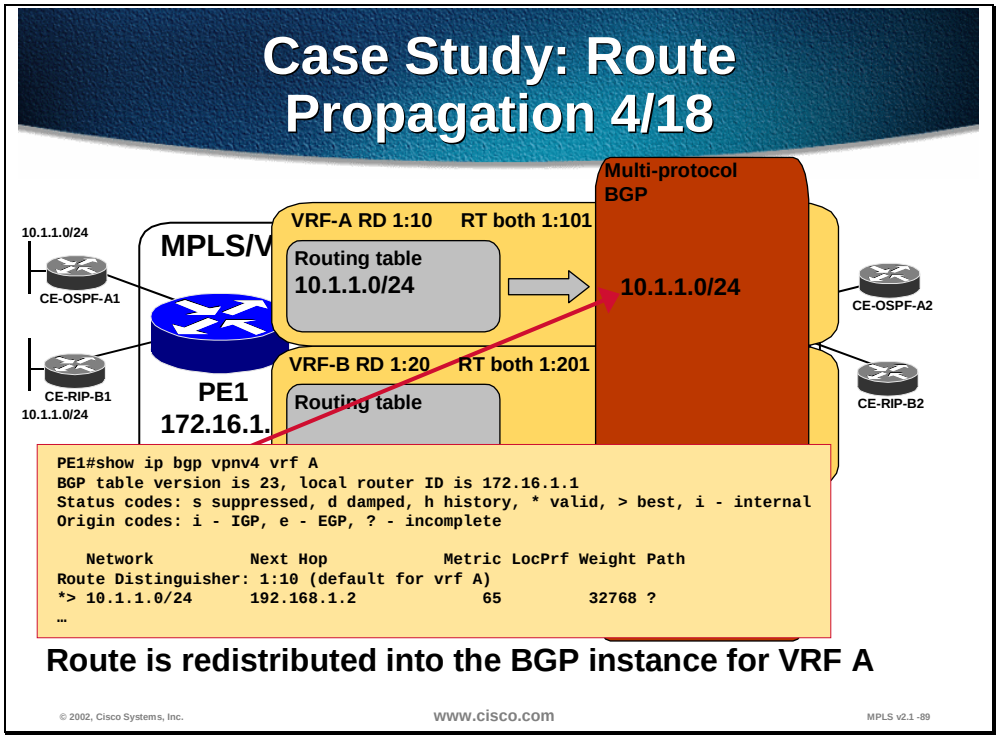
This means that when we reach PE1, the LSP is complete.

It is clear, when come this far in the analysis of the network, that there is an LSP from PE4 to PE1 which is a prerequisite for MPLS/VPN operation.

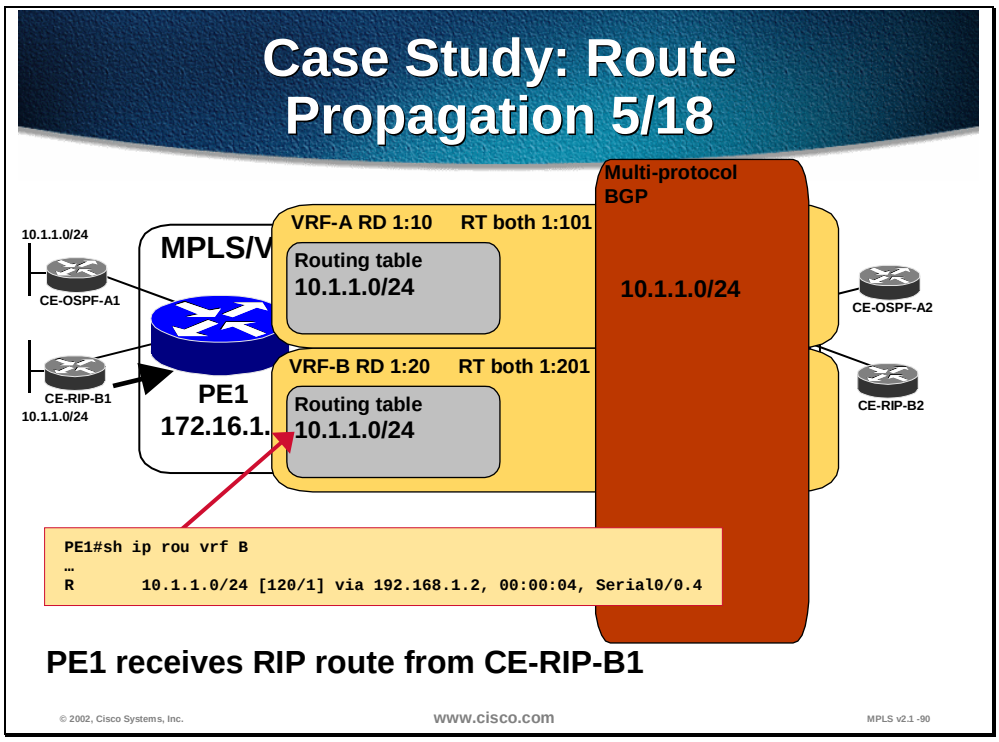


This figure illustrates PE1. PE1 has two vrfs. One for customer A and one for customer B. PE1 is using OSPF as the PE to CE routing protocol with the customer A CE router CE-OSPF-A1. It is running RIPv2 with the customer B CE router CE-RIP-B1. PE1 is also running MP-iBGP with PE4.

The route 10.1.1.0/24 has been received by OSPF from router CE-OSPF-A1. It will be stored in the vrf A routing table. This can be verfied by the **show ip route vrf A** command.

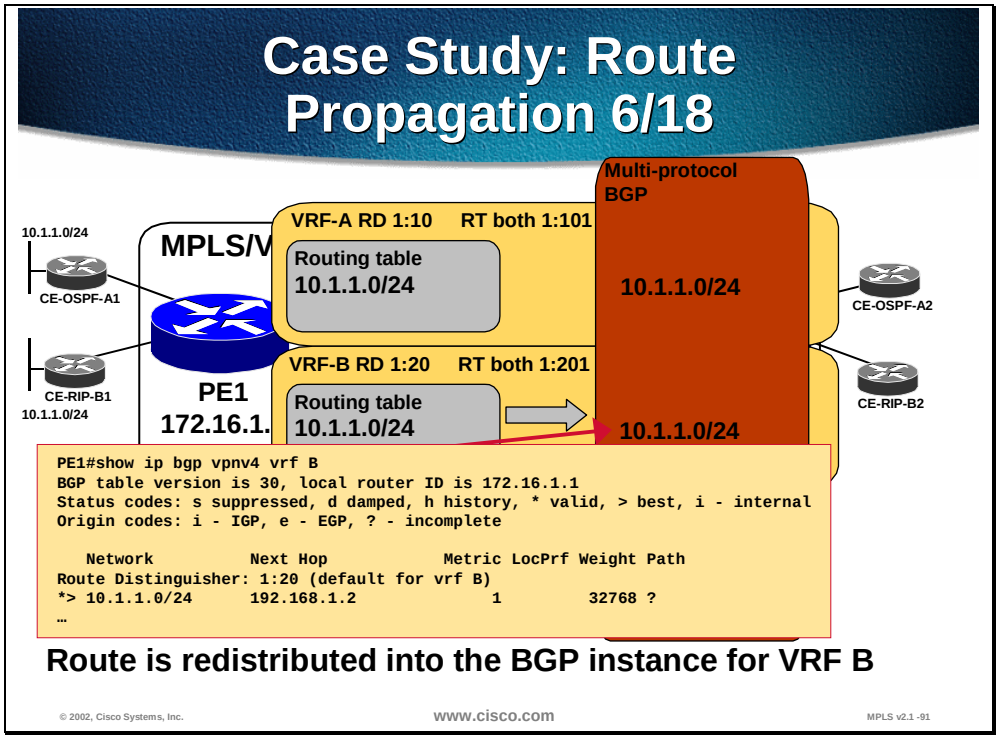


For proper MPLS/VPN operation redistribution is required from the PE-CE routing protocol into MP-BGP (unless BGP is the PE-CE routing protocol). The route has been redistributed and this is verified by using the **show ip bgp vpnv4 vrf A** command.

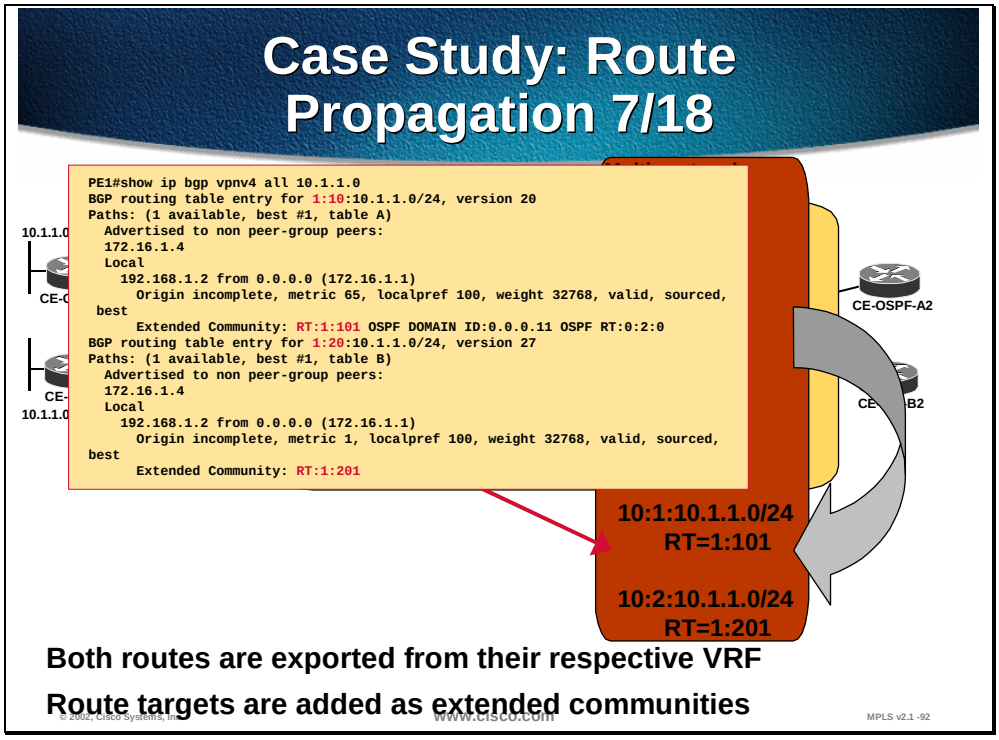


The corresponding verification steps that were made for customer A is here done for customer B. The PE-CE routing protocol is RIP, so the **show ip route vrf B** command shows the network 10.1.1.0/24 as a RIP route.

It is important to notice that the route 10.1.1.0/24 in vrf A and the route 10.1.1.0/24 in vrf B are two different routes. The two vrfs are working in parallel completely isolated from each other.



Route redistribution from PE-CE routing protocol (RIP) to BGP is required also for vrf B. The redistribution is verified by the command **show ip bgp vpnv4 vrf B**.



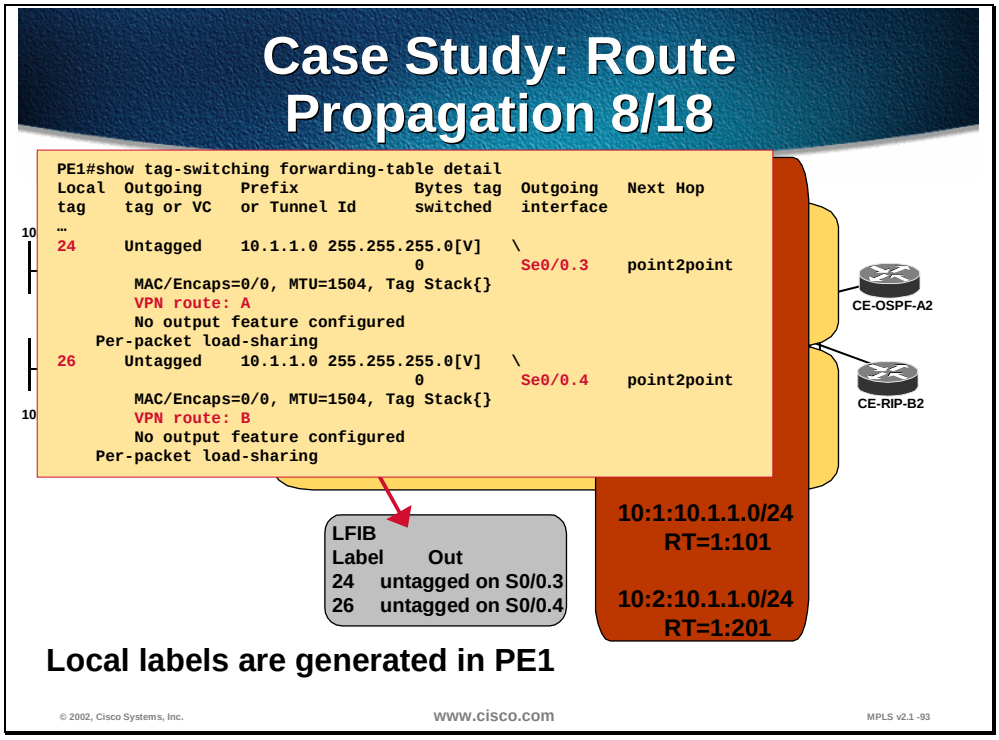
This figure illustrates the logical copying of BGP routes from the vrf context into the VPNv4 context. This copying does not really take place. Cisco IOS optimizes this so that only one copy of each route is actually required.

The copying from the vrf context to vpnv4 illustrates the export function where the IPv4 route is converted into a VPNv4 route and the export route-targets are attached.

The **show ip bgp vpnv4 all 10.1.1.0** command shows all routes in the MP-BGP table that has an IPv4 part which is 10.1.1.0. As can be seen from the output, two such routes exist.

One of the routes is the VPNv4 route 1:10:10.1.1.0/24 which is created from the IPv4 route 10.1.1.0/24 in vrf A by prefixing the route-distinguisher 1:10. The vrf A is using 1:101 as an export route-target. That is why the BGP extended community attribute RT 1:101 is attached to the VPNv4 route. Additional extended communities for OSPF are also attached.

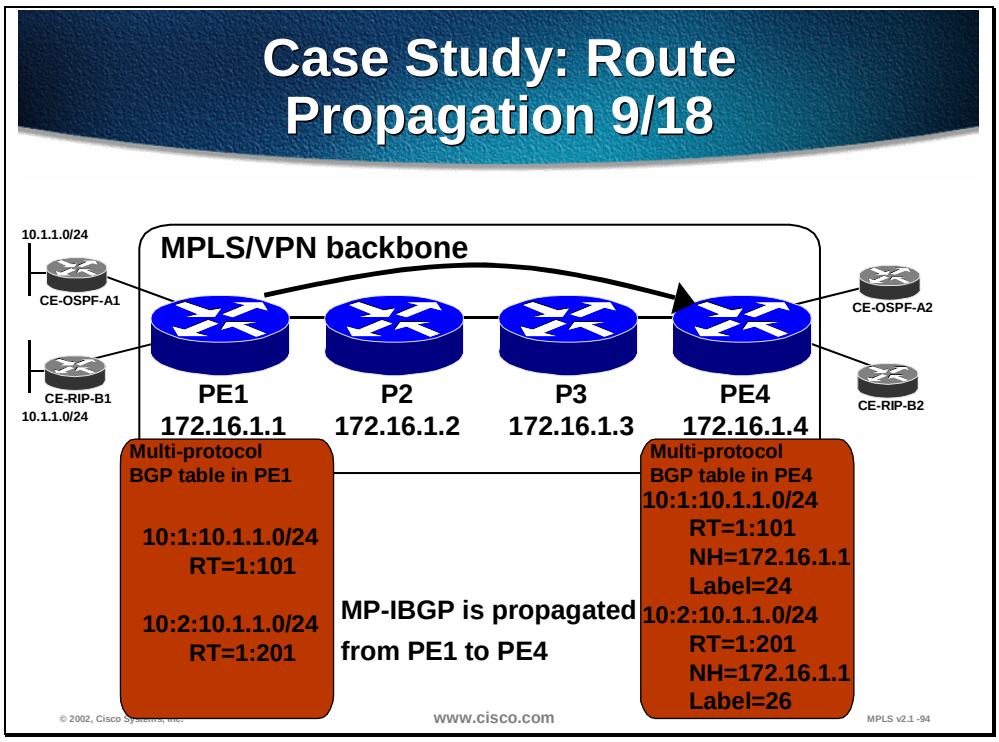
The other route is the VPNv4 route 1:20:10.1.1.0/24 which is created from the IPv4 route 10.1.1.0/24 in vrf B by prefixing the route-distinguisher 1:20. Vrf B is using 1:201 as export route-target.



Router PE1 must allocate local labels for the VPN routes. It is essential for packet forwarding.

One label must be allocated for the 10.1.1.0/24 route in vrf A and another label for the 10.1.1.0/24 route in vrf B. In this example the **show tag-switching forwarding-table detail** shows that the 10.1.1.0/24 route in vrf A has been allocated label value 24 and the 10.1.1.0/24 route in vrf B has been allocated label value 26.

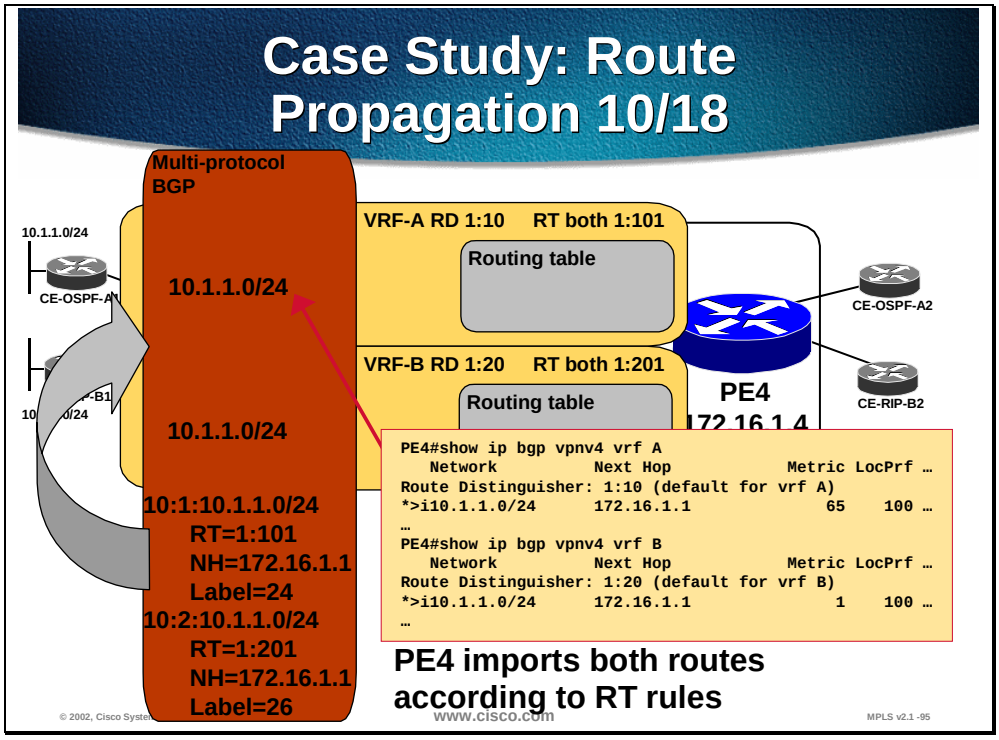
PE1 is now prepared to receive MPLS packets with label value 24 or 26. Packets with label 24 will be untagged (all labels removed) and forwarded out on Serial0/0.3. Packets with label 26 will be untagged and forwarded out on Serial0/0.4.



PE1 and PE4 are having an MP-iBGP session established. PE1 will follow the normal BGP route propagation rules and advertise the two VPNv4 routes it has itself originated. 10:1:10.1.1.0/24 and 10:2:10.1.1.0/24 are both propagated to PE4.

PE1 will indicate its own loopback, 172.16.1.1, as next-hop. This loopback interface belongs to the global routing in PE1 (no vrf forwarding). This means that PE4 can use its global routing and label switched paths to reach the next-hop.

The allocated VPN label values 24 for 10:1:10.1.1.0/24 and 26 for 10:2:10.1.1.0/24 are attached to their respective VPNv4 routes by PE1. PE4 will therefore have all the essential information for packet forwarding: the next-hop indicates how to cross the P network and the VPN label that PE1 expects.



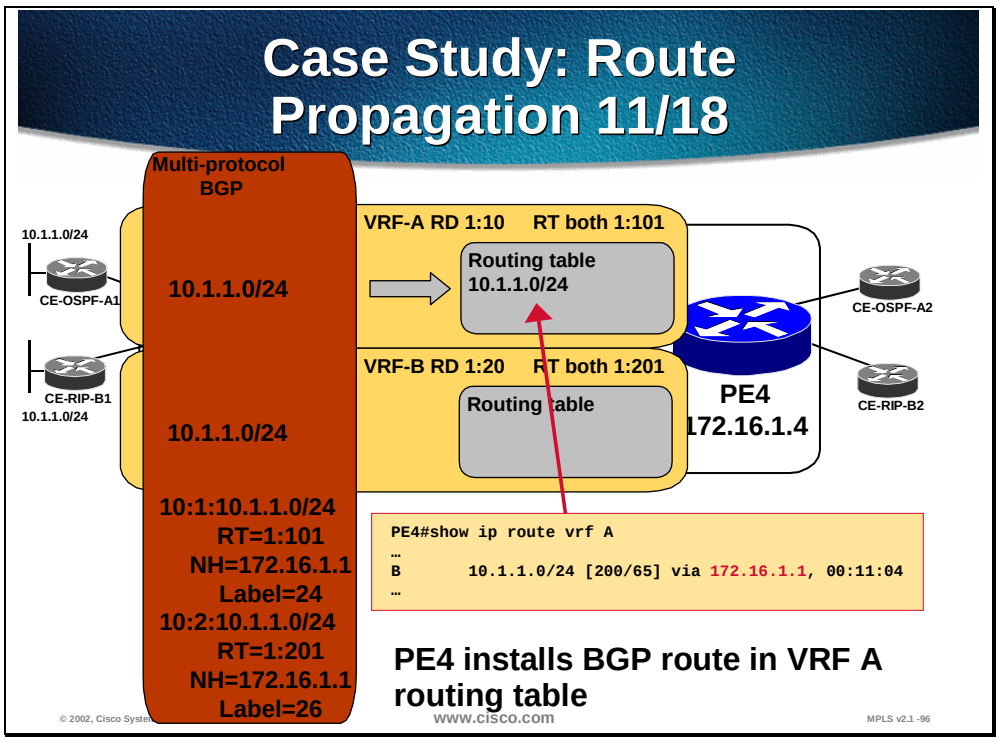
The figure illustrates PE4 and the internal data structures in that router.

The received VPNv4 MP-BGP routes must be imported into the correct vrf before they can be used. The import mechanism is controlled by the route-target attributes. Vrf A will import all routes with RT 1:101 while vrf B will import all routes with RT 1:201.

The import is illustrated as a copying of routes. However, the Cisco IOS implementation has optimized this and in reality, only one copy of each route is stored in the table.

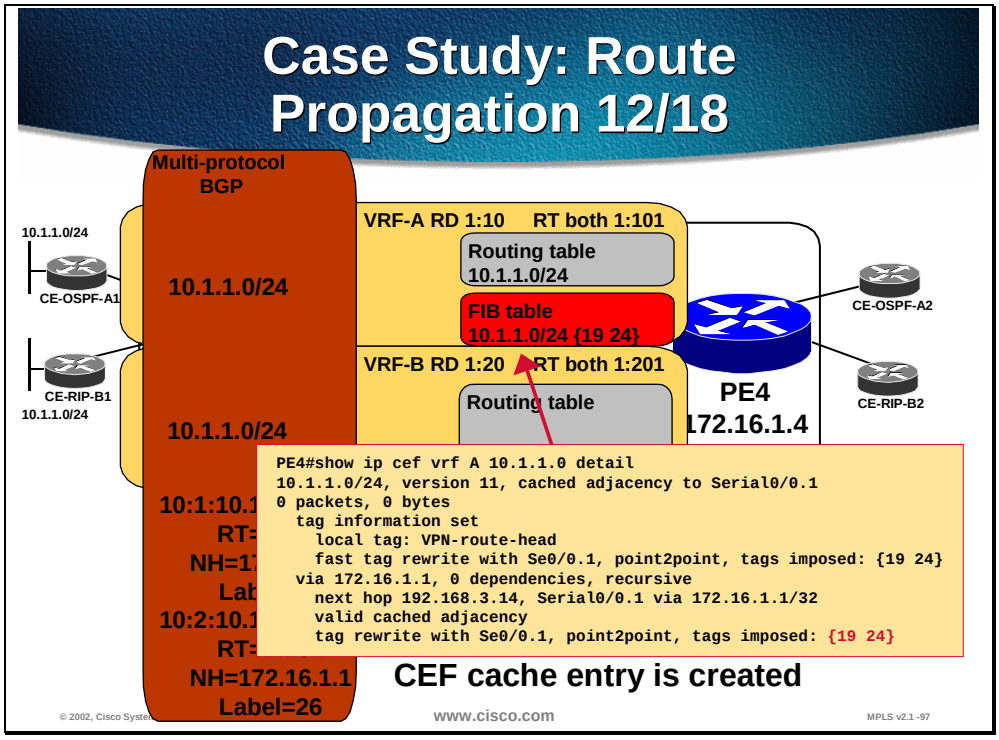
The route 10:1:10.1.1.0/24 is imported into vrf A. As this is done, it is converted back to an IPv4 route. The route 10:2:10.1.1.0/24 is imported into vrf B and converted to an IPv4 route.

The import function into vrf A is verified by the command **show ip bgp vpnv4 vrf A**. And the import into vrf B is verfied by **show ip bgp vpnv4 vrf B**.



The virtual routing and forwarding instance (the vrf) is just like a real router. It has routing protocols and data structures and procedures on how they interact. The BGP protocol in vrf A has selected the route 10.1.1.0/24 with next-hop 172.16.1.1 as best. Vrf A will use the rules on administrative distance when installing the route in the routing table. In this case, there was no other routing protocol in that vrf that had found 10.1.1.0/24. That is why the internal BGP route is installed with an administrative distance 200.

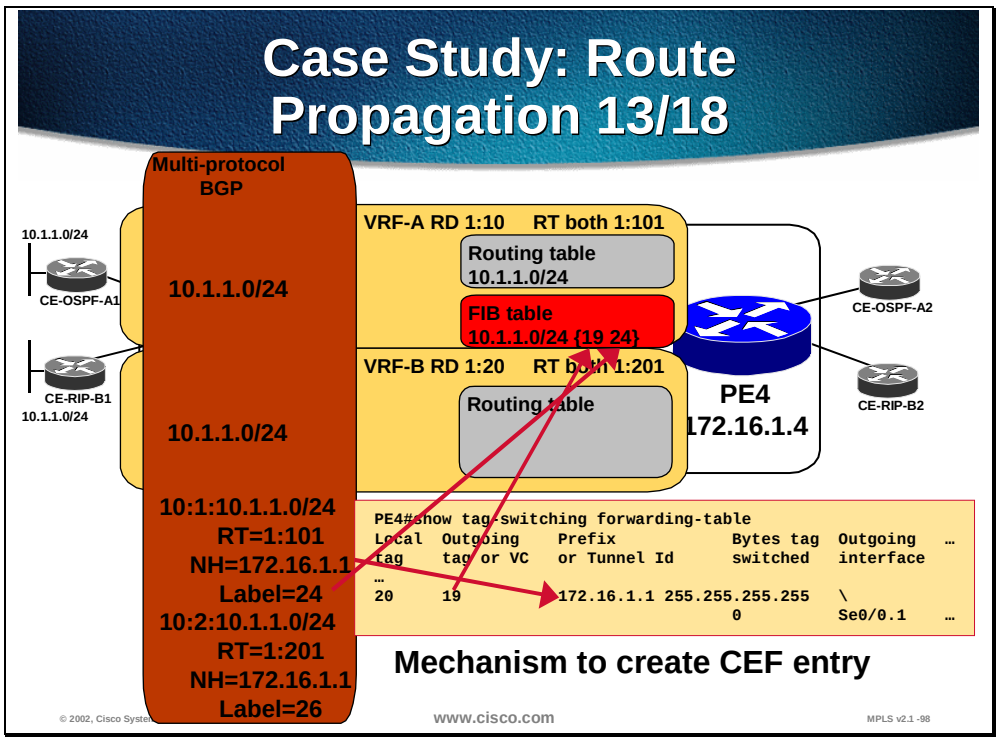
The **show ip route vrf A** command displays the content of the routing table in vrf A.



CEF switching is a requirement for MPLS/VPN operation. The installation of the route 10.1.1.0/24 into the vrf A routing table was just a preliminary step. The CEF cache, FIB table, must also be built before any packet can be forwarded.

Just like in a real router that is running CEF switching, the virtual router (vrf) has a FIB table. The entries in the FIB table are created as the entries in the routing table are created. In this case, the figure shows how an entry in the FIB table in vrf A is created for the destination network 10.1.1.0/24. The command **show ip cef vrf A 10.1.1.0 detail** shows the CEF cache entry including the label stack which is associated with it.

The FIB table will be used when an IP packet arrives on any of the interfaces which belongs to vrf A. IP packets with destination in subnet 10.1.1.0/24 will be forwarded out on interface Serial0/0.1 (towards router P3) as MPLS packets with an imposed label stack of label 19 and label 24.

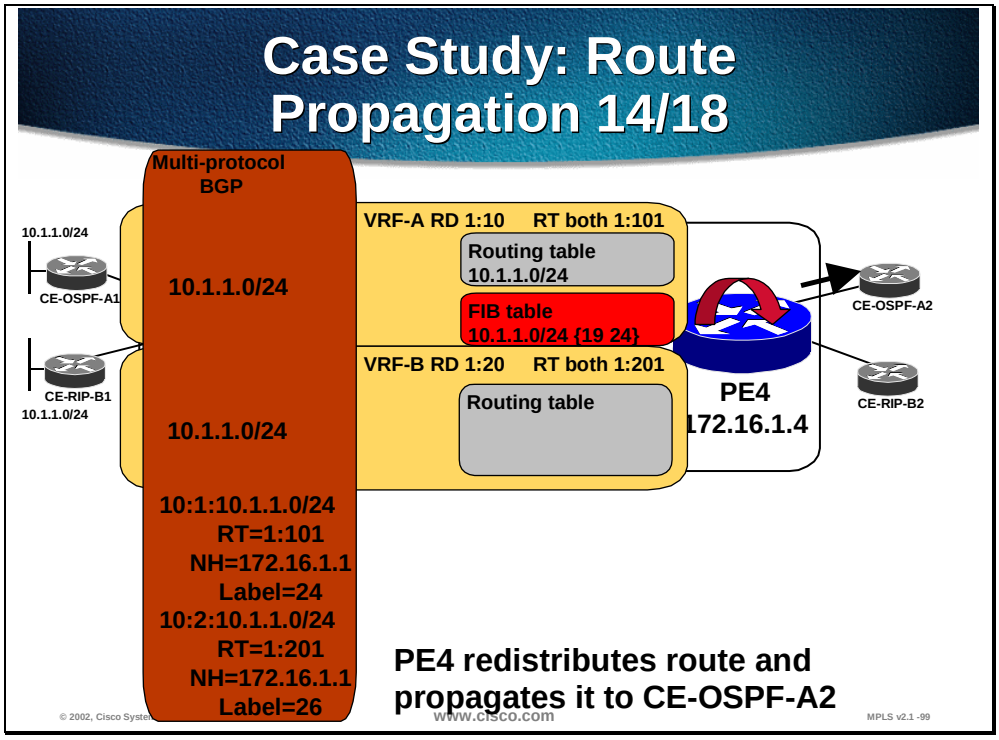


The label stack in the CEF cache entry is created using some BGP attributes.

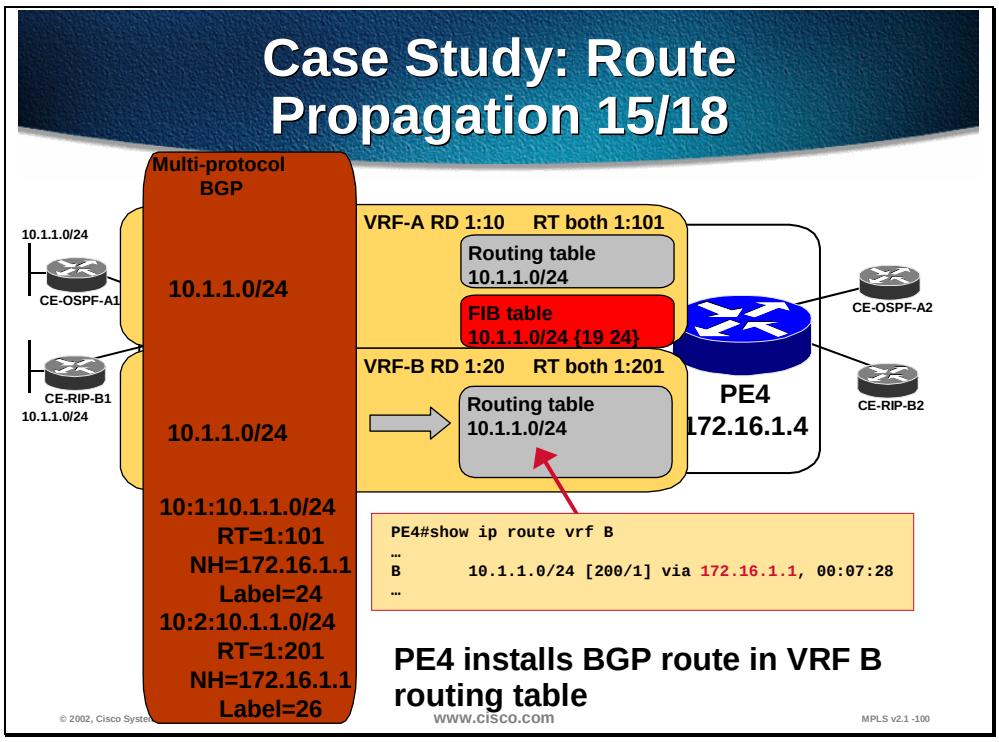
The FIB entry for 10.1.1.0/24 in vrf A is created from a BGP route in the routing table. This BGP route has a next-hop 172.16.1.1. The next-hop is always a destination in the global routing table. The router is therefore looking up in the global routing table how to reach 172.16.1.1. It turns out that there is a label switched path established for that destination. That means that PE4 should use label 19 and forward packets out on Serial0/0.1 (towards P3) in order to have the packet delivered to 172.16.1.1 (PE1). The top-most label in the label stack must therefore be 19.

The second label in the stack is derived from the VPN label information in the MP-BGP entry for the route. In this case the VPN label was 24.

Thus a label stack of 19 and 24 has been created. This label stack is used in the FIB table for the entry 10.1.1.0/24 in vrf A.

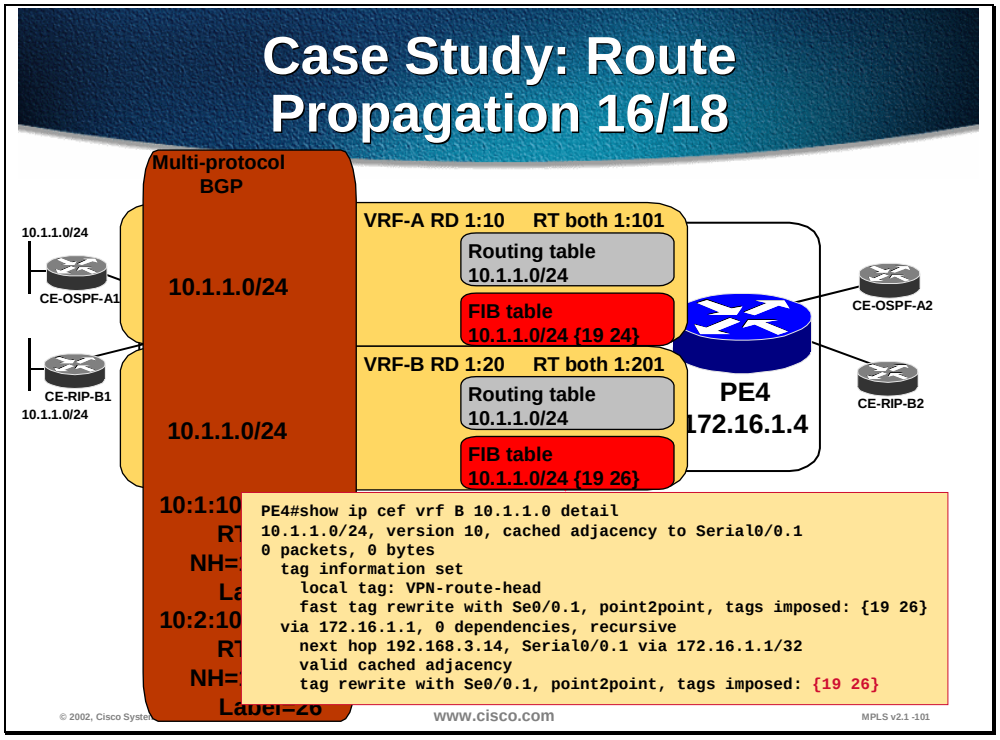


PE4 and CE-OSPF-A2 are running OSPF as the PE-CE routing protocol. This means that the BGP route in the vrf A routing table must be redistributed into the OSPF process that is running in the vrf. When this is done, the IPv4 OSPF route can be propagated to CE-OSPF-A2.



The vrf B is working in parallel with vrf A without any interactions. The BGP protocol in vrf B has selected the route 10.1.1.0/24 with next-hop 172.16.1.1 as best. Vrf B will use the same rules as vrf A is using when installing the route in the routing table. So the internal BGP route 10.1.1.0/24 is installed with an administrative distance 200.

The **show ip route vrf B** command displays the content of the routing table in vrf B.

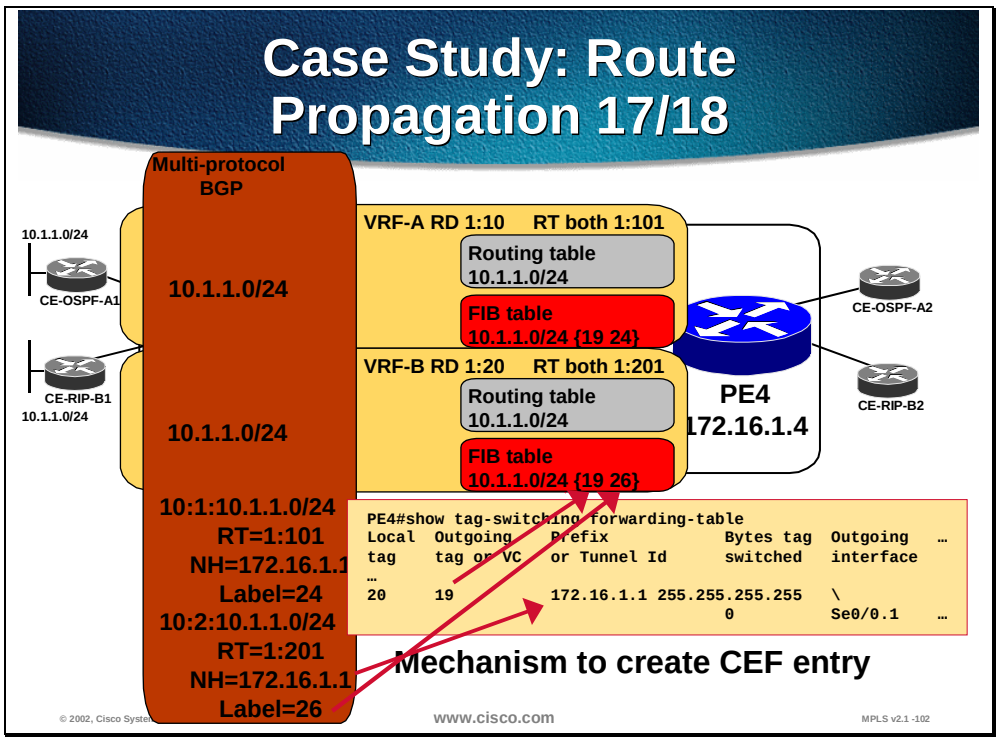


CEF switching must of course also be enabled on all interfaces in vrf B. The installation of the route 10.1.1.0/24 into the vrf B routing table is immediately followed by the creation of a FIB table entry.

In this case, the figure shows how an entry in the FIB table in vrf B is created for the destination network 10.1.1.0/24. The command **show ip cef vrf B 10.1.1.0 detail** shows the CEF cache entry including the label stack which is associated with it.

The FIB table will be used when an IP packet arrives on any of the interfaces which belongs to vrf B. IP packets with a destination in subnet 10.1.1.0/24 will be forwarded out on interface Serial0/0.1 (towards router P3) as MPLS packets with an imposed label stack of label 19 and label 26.

The important thing here is that there are two different FIB tables in the figure. One for vrf A and one for vrf B. The vrf A FIB table will be used when IP packets arrive on an interface that belongs to vrf A while the vrf B FIB table will be used when IP packets arrive on an interface that belongs to vrf B. As can be seen, both the FIB tables have entries for network 10.1.1.0/24. The incoming interface is controlling which of these entries that will be used for packet forwarding.

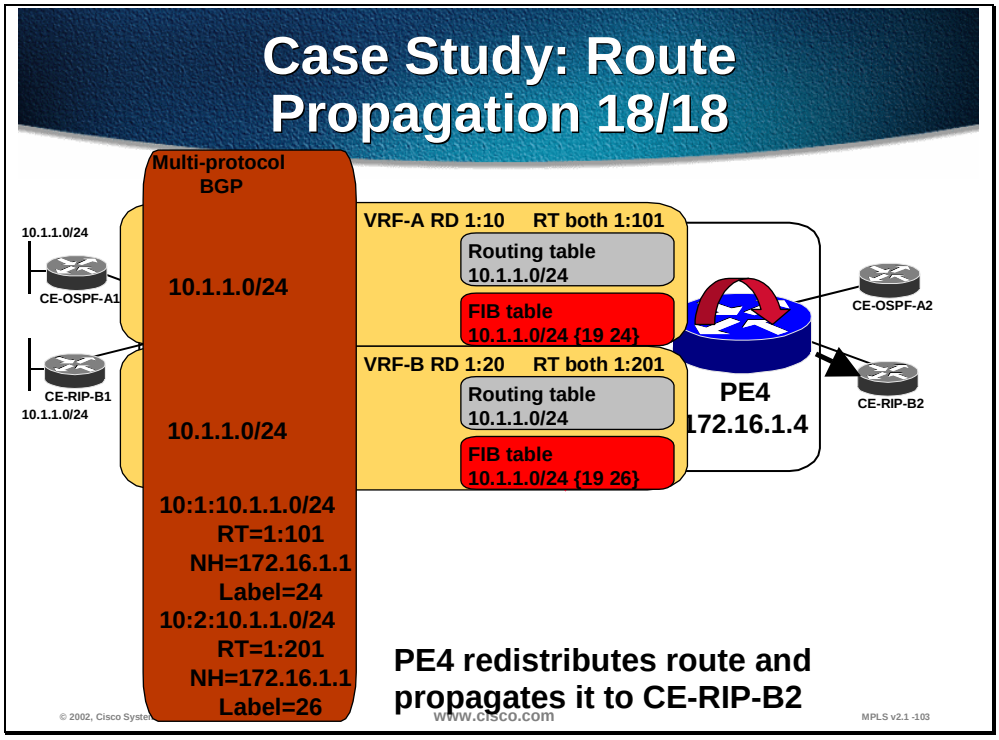


The label stack in the CEF cache entry in vrf B is created using the corresponding BGP attributes as was illustrated when the CEF cache entry in vrf A was created.

The vrf B BGP route 10.1.1.0/24 also has the next-hop 172.16.1.1. The global routing how to reach the next hop is the same in this case as in the vrf A case. That means that PE4 should use label 19 and forward packets out on Serial0/0.1 (towards P3) in order to have the packet delivered to 172.16.1.1 (PE1). Also in this case, the top-most label in the label stack must therefore be 19.

The second label in the stack is derived from the VPN label information in the MP-BGP entry for the route. In this case the VPN label was 26.

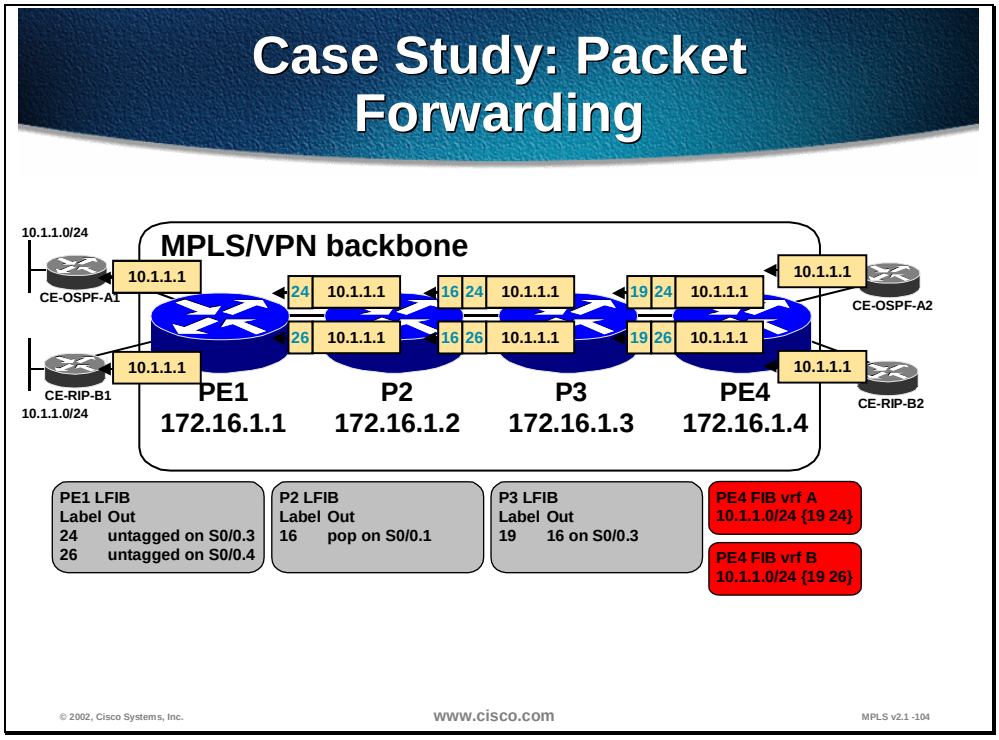
Thus a label stack of 19 and 26 has been created. This label stack is used in the FIB table for the entry 10.1.1.0/24 in vrf B.



PE4 and CE-RIP-B2 are running RIPv2 as the PE-CE routing protocol. This means that the BGP route in the vrf B routing table must be redistributed into the RIP process that is running in the vrf. When this is done, the IPv4 RIP route can be propagated to CE-RIP-B2.

This completes the case study on route propagation.

Comprehensive Case Study: Packet Forwarding



This case study illustrates how packets are forwarded from CE-OSPF-A2 via the P network to CE-OSPF-A1 and how packets are forwarded from CE-RIP-B2 via the P network to CE-RIP-B1. Before this packet forwarding can occur, route propagation must be completed.

CE-OSPF-A2 is using OSPF as the routing protocol. It has received information from PE4 that the destination network 10.1.1.0/24 is reachable via PE4. CE-OSPF-A2 therefore forwards a packet to destination 10.1.1.1 to PE4.

PE4 receives the packet on an interface that belongs to vrf A. PE4 performs CEF switching of the IP packet using the FIB table for vrf A. The FIB entry for network 10.1.1.0/24 in vrf A tells PE4 to forward the packet as an MPLS packet to P3 with the label stack {19 24} imposed.

P3 receives an MPLS packet and therefore does its lookup in the LFIB. It looks only on the top-most label and label switches the packet to P2. The top most label was replaced and the value is now 16.

P2 does an LFIB lookup and finds that MPLS packets with label 16 should have their top most label popped off and whatever is left should be propagated to PE1. This means that the packet is transmitted out to PE1 with only a single label, label value 24.

PE1 receives an MPLS packet with label 24. The LFIB in PE1 says that the packet should have all its labels removed and the IP packet should be forwarded out to CE-OSPF-A1.

CE-RIP-B2 is using its RIP information to forward packets with destination 10.1.1.1 to PE4. PE4 now receives the packet on a vrf B interface so when the packet is propagated to P3, the label stack will be {19 26}. P3 and P2 will not see any difference between the two packets in this case study. They will perform the same operations. Only PE1 will see the difference because the VPN label value 26 is exposed. That is why PE1 forwards that packet to CE-RIP-B1.

Summary

After completing this section, you should be able to perform the following tasks:

- Monitor individual VRFs and routing protocols running in them
- Monitor MP-BGP sessions between the PE routers
- Monitor inter-AS MP-BGP sessions between the PE routers
- Monitor MP-BGP table
- Monitor CEF and LFIB structures associated with MPLS/VPN

Next Steps

After completing this lesson, go to:

- Troubleshooting MPLS/VPN

Lesson Review

Instructions

Answer the following questions:

1. How would you verify the contents of a VRF routing table?
2. How would you display an individual entry in a VRF CEF table?
3. How would you display routing protocols running in a VRF?
4. Why is the BGP protocol always running in every VRF?
5. How would you inspect the label stack associated with a remote MPLS/VPN route?
6. How would you verify VPNv4 information exchange with a MP-BGP neighbor?
7. How would you display all routes with a specified route distinguisher?
8. How would you display all labels associated with a VRF?
9. Why do you only see labels for routes learned from CE routers?
10. Would you ever see labels for routes received through MP-BGP in your LFIB?

Troubleshooting MPLS/VPN

Overview

This lesson describes common MPLS VPN troubleshooting tasks.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Verify proper PE-to-PE connectivity
- Verify proper redistribution of VPN routes and creation of MPLS labels
- Verify VPN route propagation and data forwarding

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

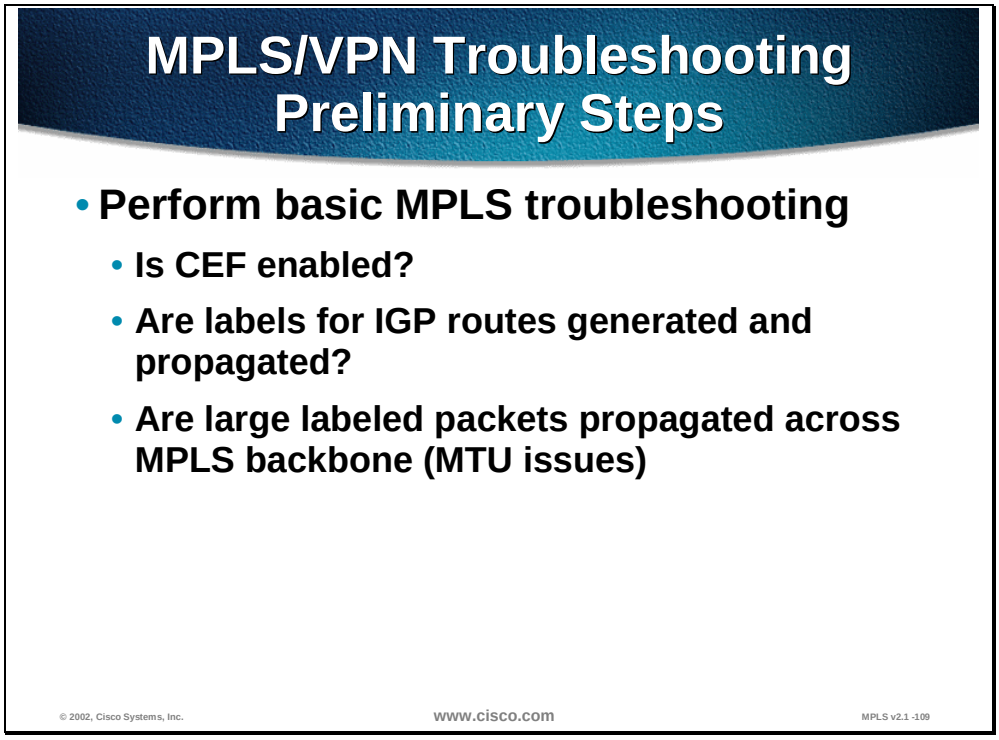
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- Basic MPLS Troubleshooting
- Troubleshooting Route Propagation
- Troubleshooting VPN Data Forwarding
- Summary
- Lesson Review

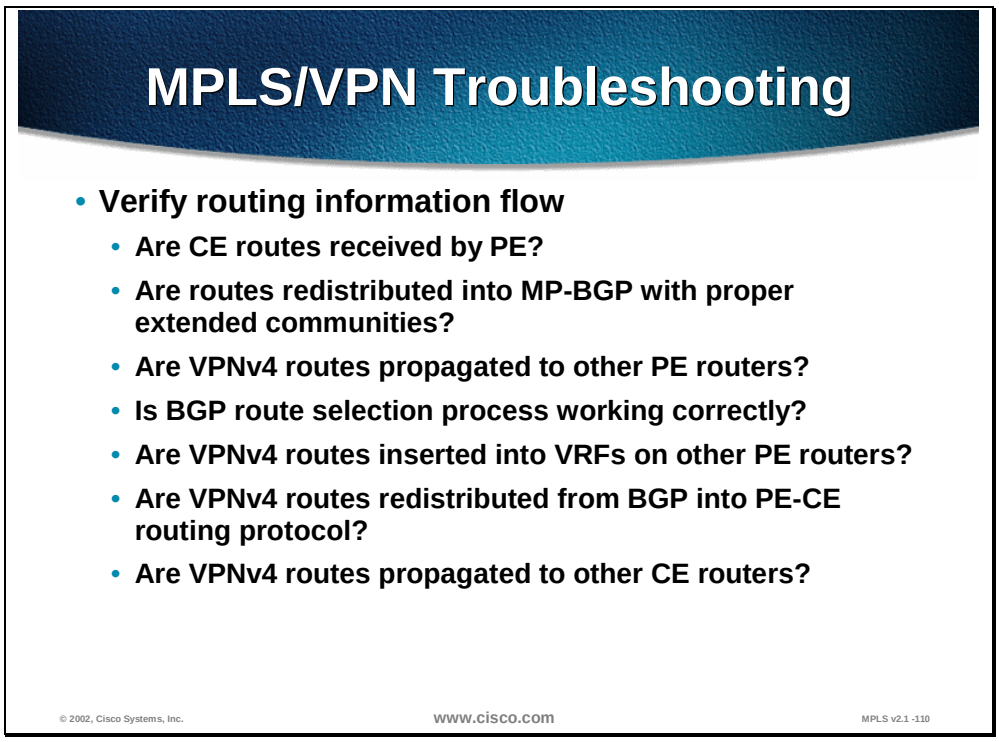
Basic MPLS Troubleshooting



Before you start in-depth MPLS/VPN troubleshooting, you should ask the following standard MPLS troubleshooting questions:

- Is CEF enabled on all routers in the transit path between the PE routers?
- Are labels for BGP next-hops generated and propagated?
- Are there any MTU issues in the transit path (for example, LAN switches not supporting jumbo Ethernet frame)?

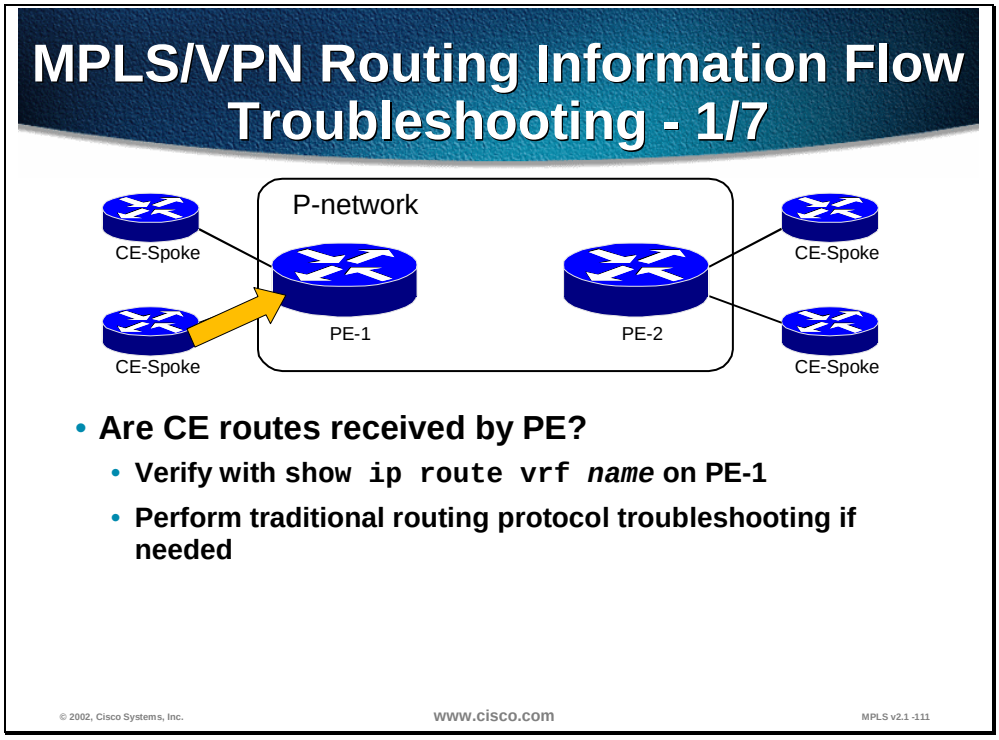
Please refer to the “Configuring Frame-mode MPLS on Cisco IOS Platforms” and “Configuring Cell-mode MPLS on Cisco IOS Platforms” for detailed description of these troubleshooting steps.



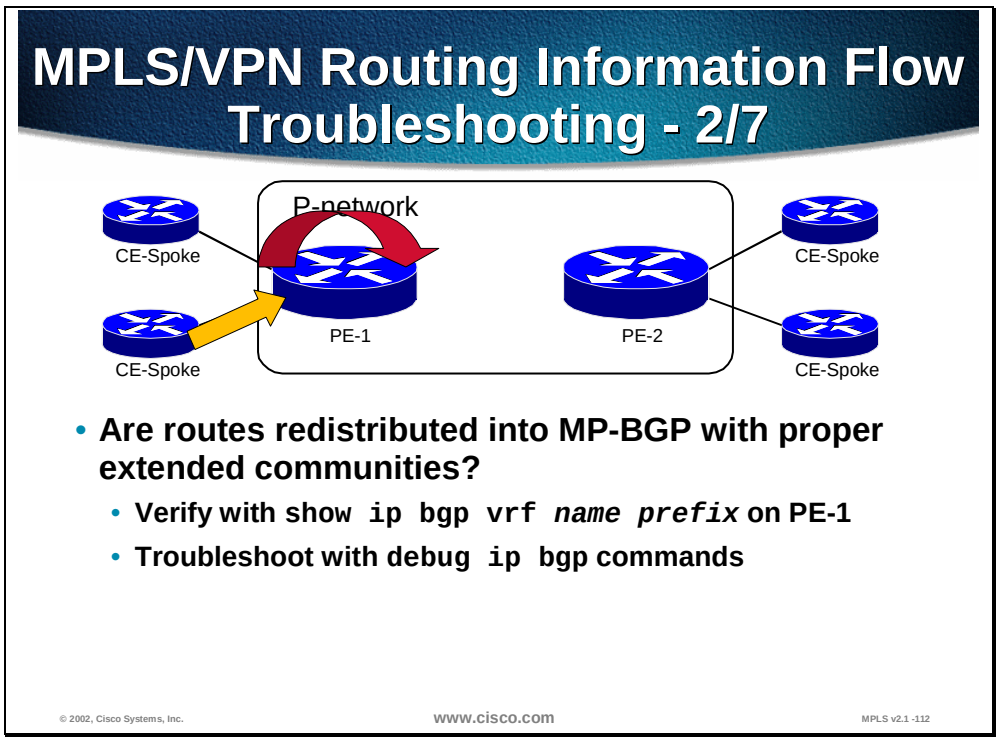
MPLS/VPN troubleshooting consists of two major steps:

- Verify the routing information flow using the checks outlined in the slide
- Verify the packet forwarding (discussed later in this section)

Troubleshooting Route Propagation



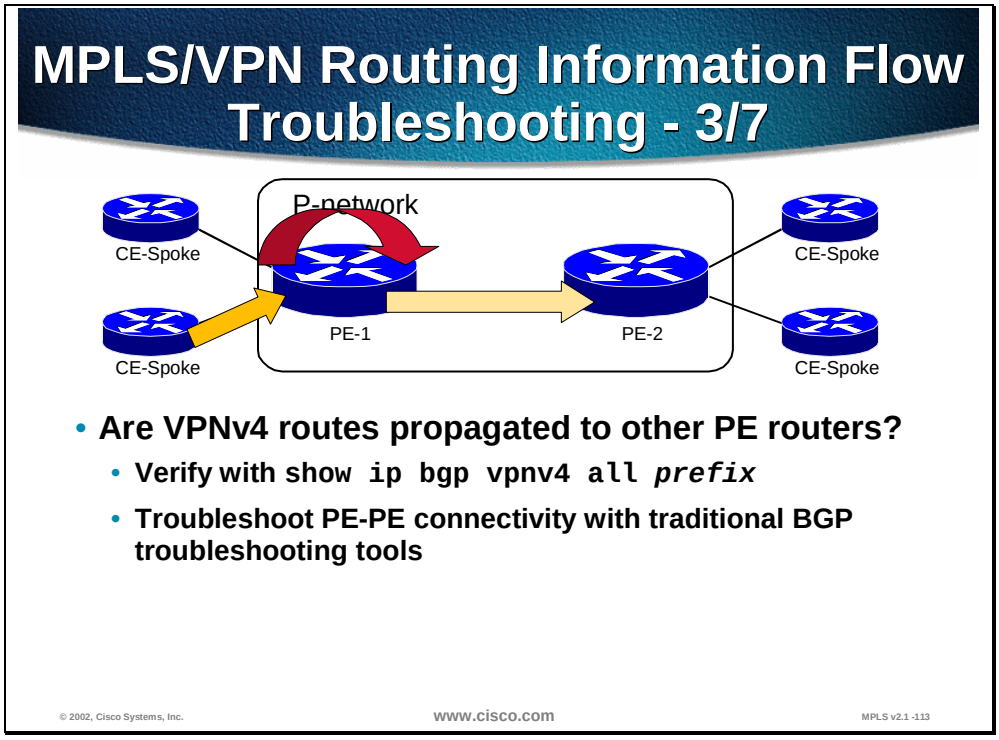
Routing information flow troubleshooting has to verify end-to-end routing information propagation between CE routers. The first step to check is the CE to PE router routing information exchange. Use the **show ip route vrf *name*** command to verify that the PE router receives customer routers from the CE router. Use traditional routing protocol troubleshooting if needed (the troubleshooting of standard enterprise routing protocols is described in the **Cisco Internetworking Troubleshooting** course and BGP-specific troubleshooting is described in the individual implementation lessons of the **BGP curriculum**).



The CE routes received by the PE router need to be redistributed into MP-BGP; otherwise, they will not get propagated to other PE routers. Common configuration mistakes in this step include:

- Not configuring redistribution between the PE-CE routing protocol and per-VRF routing context of the BGP
- Using route-map on redistribution that filters CE routes

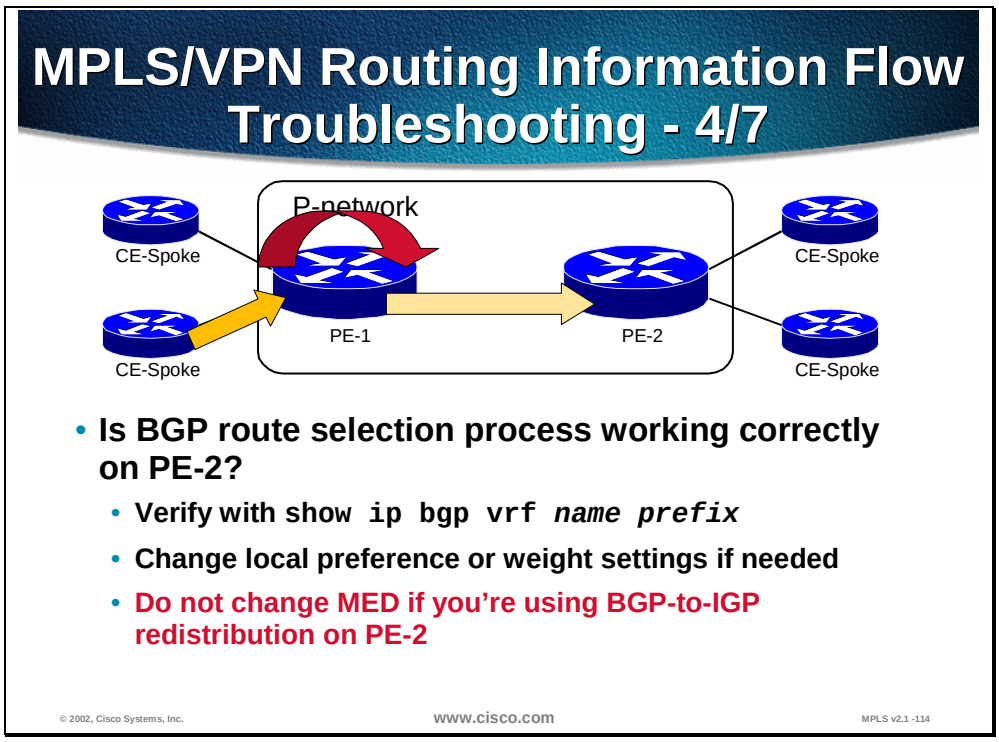
Proper redistribution of CE routes into per-VRF instance of BGP can be verified with the **show ip bgp vrf *name*** command. The route distinguisher prepended to the IPv4 prefix and the route targets attached to the CE route can be verified with the **show ip bgp vrf *name prefix*** command.



The CE routes redistributed into MP-BGP need to be propagated to other PE routers. Verify the proper route propagation with the **show ip bgp vpnv4** command on the remote PE router.

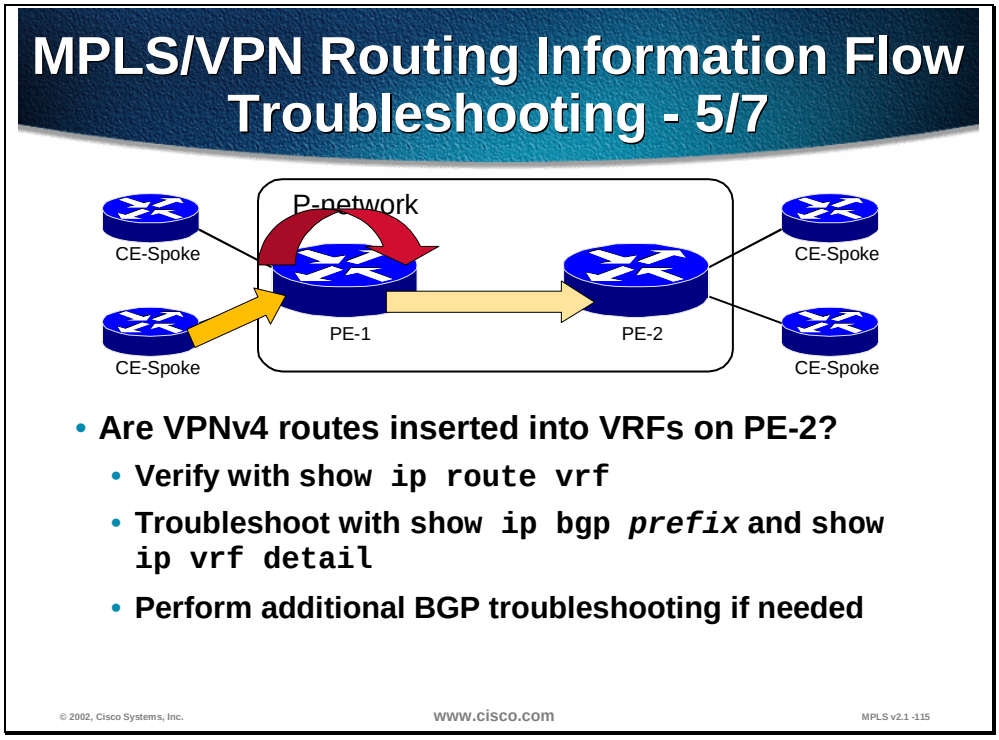
Note	Routes sent by the originating PE router might not be received by remote PE router because of automatic route-target-based filters installed on the remote PE router. Please refer to the chapter **Large Scale MPLS VPN Deployment** in the **MPLS VPN Solutions** lesson for more details on automatic route filters.

Automatic route filters are based on route targets; verify that the route targets attached to the CE route in the originating PE router match at least one of the route targets configured as import route targets in the VRF on the receiving PE router.



In complex environments with multi-homed customer sites, the BGP route selection process might affect the proper MPLS/VPN operation. Use standard BGP route selection tools (weights or local preference) to influence BGP route selection. MED should not be changed inside the MPLS/VPN backbone if you plan to use two-way route redistribution between the PE-CE routing protocol and BGP.

Please refer to the **BGP Filtering and Route Selection** lesson for more information on BGP weights and to **Advanced BGP Configuration** lesson for more information on BGP local preference and MED.



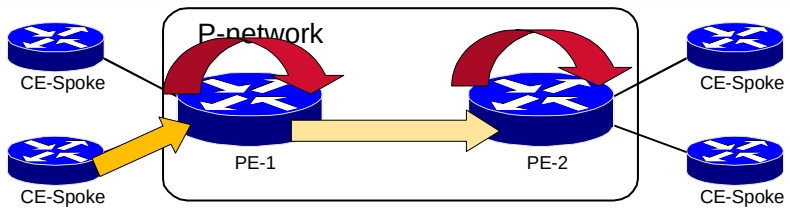
The VPNv4 routes received by the PE router have to be inserted into the proper VRF, which can be verified with **show ip route vrf** command. Common configuration mistakes in this step include:

- Wrong import route targets configured in the VRF
- The route-map configured as import route-map is rejecting the VPNv4 routes (please refer to further sections in this lesson for more information on import route-map).

The validity of the import route targets can be verified with the **show ip bgp vpnv4 all *prefix*** command, which displays the route targets attached to a VPNv4 route and with the **show ip vrf detail** command that lists the import route targets for a VRF. At least one route target attached to the VPNv4 route needs to match at least one route-target in the VRF.

Note	Be patient when troubleshooting this step – the import of VPNv4 routes into VRFs is not immediate and can take more than a minute in worst circumstances. Please refer to the **MPLS VPN Solutions** lesson for more information on improving route import speed.

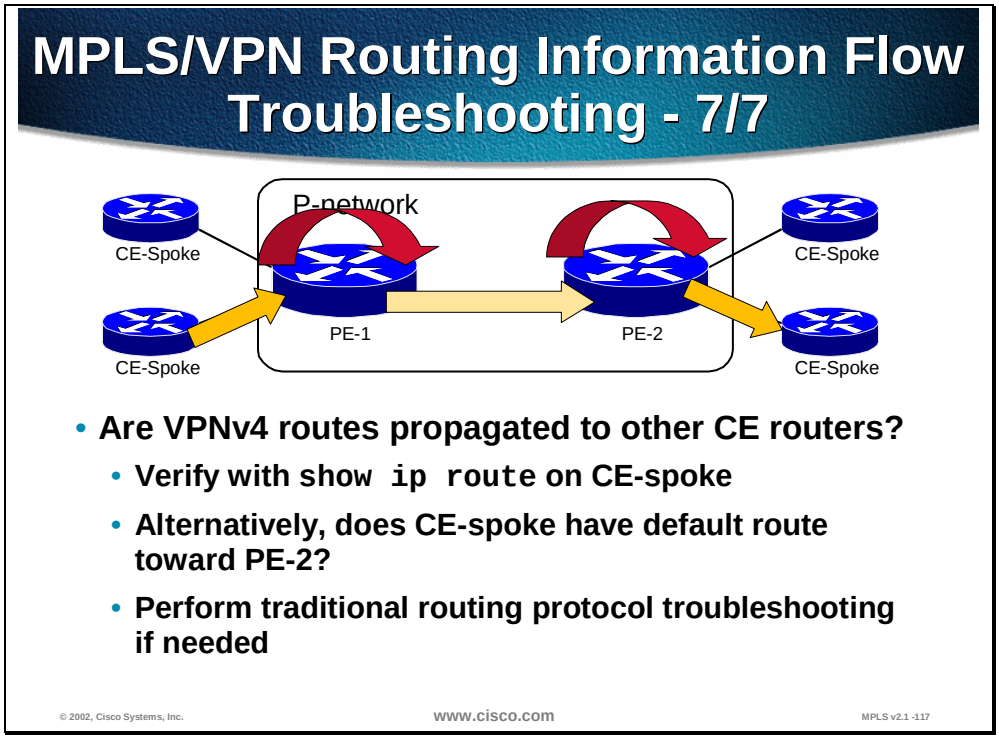
MPLS/VPN Routing Information Flow Troubleshooting - 6/7



- **Are VPNv4 routes redistributed from BGP into PE-CE routing protocol?**
  - **Verify redistribution configuration - is IGP metric specified?**
  - **Perform traditional routing protocol troubleshooting**

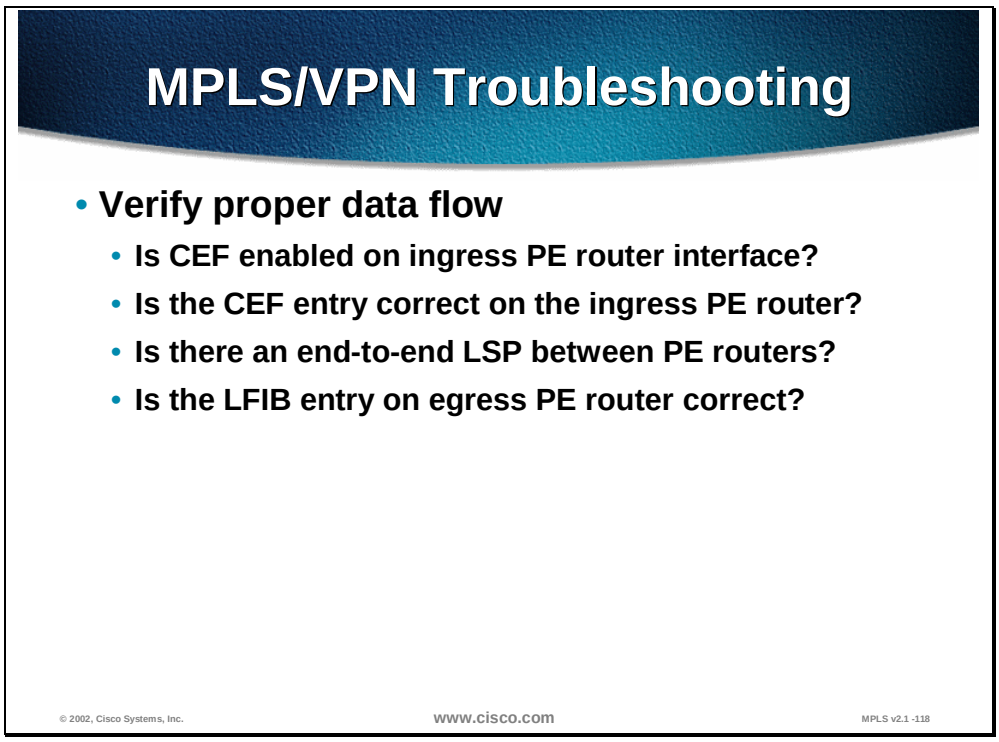
Finally, the BGP routes received via MP-BGP and inserted into the VRF need to be redistributed into the PE-CE routing protocol. A number of common redistribution mistakes sometimes occur here, starting with missing redistribution metrics.

Please refer to the Building Scalable Cisco Networks (BSCN) and Cisco Internetworking Troubleshooting (CIT) courses for more information on route redistribution troubleshooting.



Last but not least, the routes redistributed into the PE-CE routing protocol have to be propagated to CE routers (or the CE routers need a default route toward PE routers). Use standard routing protocol troubleshooting techniques in this step.

Note	When using a default route on the CE routers, verify that the CE routers use classless routing configured with the **ip classless** command.



After you've verified a proper route exchange, start MPLS/VPN data flow troubleshooting using the checks listed in the slide.

Practice

Q1) What are the preliminary MPLS VPN troubleshooting steps?

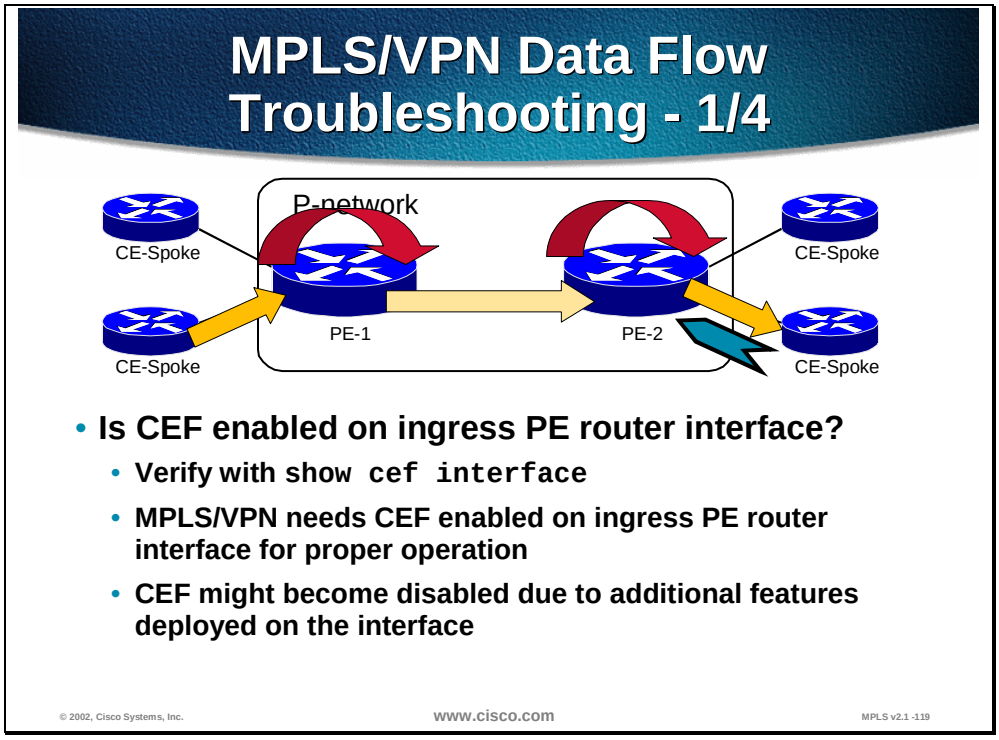
A) Verify that fast-switching or optimum switching is used, that there are labels for the BGP next-hop addresses and that there is there is no MTU issues in the transit path.

B) Verify that IS-IS or OSPF is running, that there are labels for all BGP derived routes and that you can traceroute the transit path.

C) Verify that CEF is enabled, that there are labels for the BGP next-hop addresses and that there is no MTU issues in the transit path.

D) Verify that CEF is enabled, that there are labels for all BGP derived routes and that there is no MTU issues in the transit path.

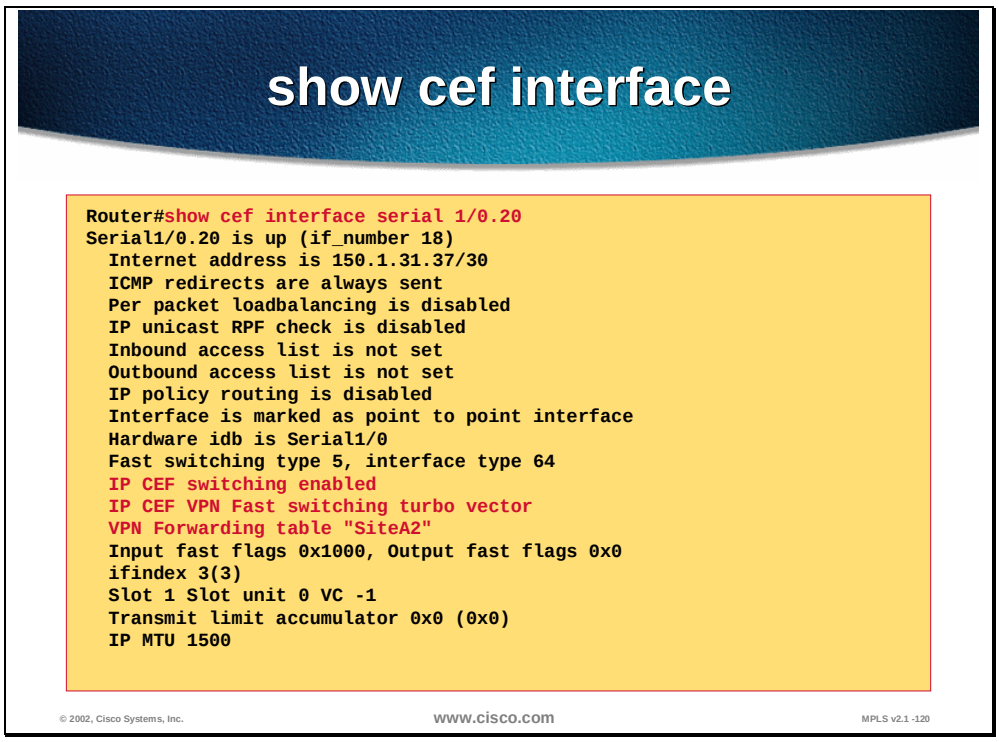
Troubleshooting VPN Data Forwarding



One of the most common data-flow related configuration mistakes is the failure to enable CEF in ingress PE router interface, which can be verified with the **show cef interface** command. CEF is the only switching method that can perform per-VRF lookup and thus support MPLS/VPN architecture.

There are three common reasons for this problem (assuming that CEF is enabled on the router):

- CEF is manually disabled on an interface
- The interface is using an encapsulation method that is not supported by CEF, for example, X.25 or multi-link PPP with interleaving
- Another feature has been configured on the interface that disables CEF (for example, IP precedence accounting)



show cef interface

To display Cisco Express Forwarding (CEF) related interface information, use the **show cef interface** command in EXEC mode.

show cef interface *type number* [**detail**]

Syntax Description

type number	Interface type and number for displaying CEF-related information.
detail	(Optional) Displays detailed CEF information for the specified interface type and number.

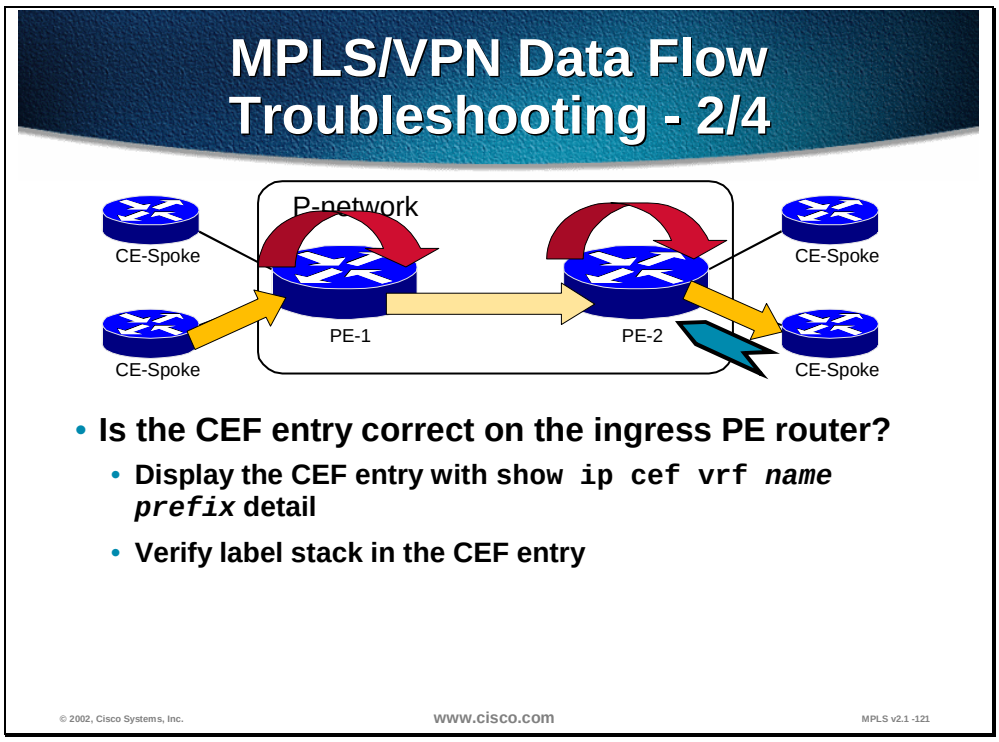
Usage Guidelines

This command is available on routers that have RP cards and line cards.
The **detail** keyword displays more CEF-related information for the specified interface. You can use this command to show the CEF state on an individual interface.

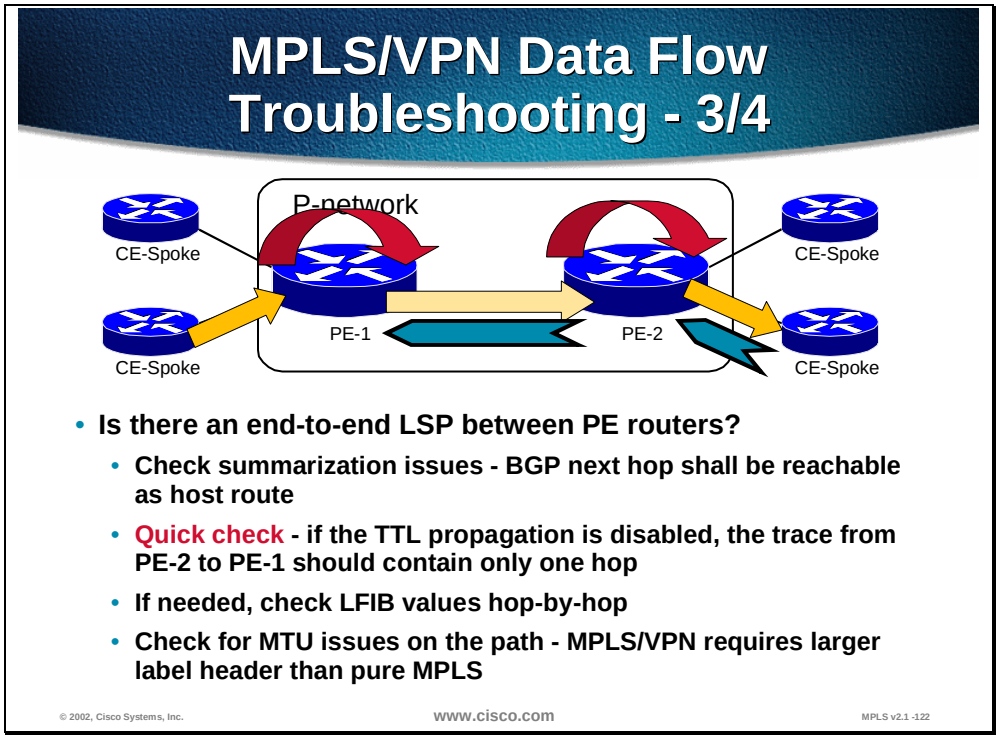
The following table describes the fields shown in the output.

Table: show cef interface detail Field Descriptions

Field	Description
interface type number is {up | down}	Indicates status of the interface.
Internet address	Internet address of the interface.
ICMP packets are {always sent | never sent}	Indicates how packet forwarding is configured.
Per-packet load balancing	Status of load balancing in use on the interface (enabled or disabled).
Inbound access list {# | Not set}	Number of access lists defined for the interface.
Outbound access list	Number of access lists defined for the interface.
Hardware idb is *type number*	Interface type and number configured.
Fast switching type	Used for troubleshooting; indicates switching mode in use.
IP Distributed CEF switching {enabled | disabled}	Indicates the switching path used.
Slot *n* Slot unit *n*	The slot number.
Hardware transmit queue	Indicates the number of packets in the transmit queue.
Transmit limit accumulator	Indicates the maximum number of packets allowed in the transmit queue.
IP MTU	The value of the MTU size set on the interface.

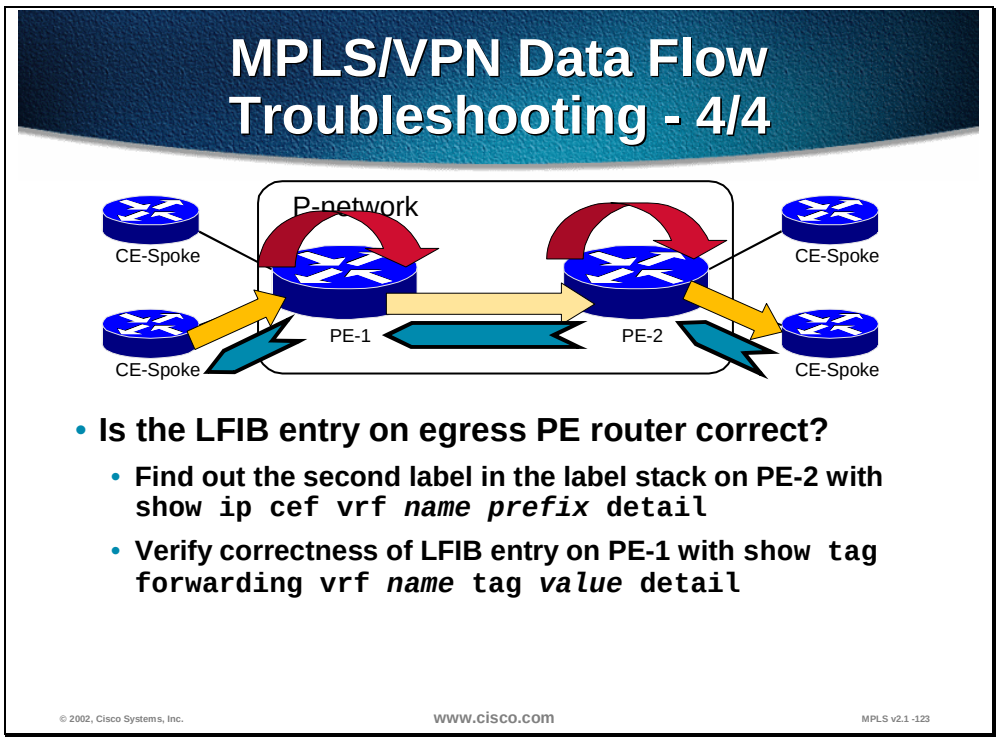


If the CEF switching is enabled on ingress interface, you can verify the validity of CEF entry and the associated label stack with the **show ip cef vrf *name prefix* detail** command. The top label in the stack should correspond to the BGP next-hop label as displayed by the **show tag forwarding** command on the ingress router and the second label in the stack should correspond to the label allocated by the egress router as displayed by the **show tag forwarding** command on the egress router.



If the CEF is enabled on the ingress interface and the CEF entry contains proper labels, the data flow problem might lie inside the MPLS core. Two common mistakes include summarization of BGP next hops inside the core IGP and MTU issues.

The quickest check on a potential summarization problem can be done by disabling IP TTL propagation into the MPLS label header by using the **no tag-switching ip ttl-propagate** command. The traceroute command toward BGP next-hop shall display no intermediate hops when the TTL propagation is disabled. If the intermediate hops are displayed, the label switched path between PE routers is broken at those hops and the VPN traffic cannot flow.



As a last troubleshooting measure (usually not needed), you can verify the contents of Label Forwarding Information Base (LFIB) on the egress PE router and compare it with the second label in the label stack on the ingress PE router. A mismatch indicates an internal IOS error that has to be reported to Cisco Technical Assistance Center (TAC).

Practice

Q1) How would you test end-to-end data flow between PE routers?

A) Use traceroute from PE router to PE router and verify that each hop is tag-switched (except for the final hop due to penultimum-hop-popping)

B) Use ping from PE router to PE router

C) The end-to-end data flow cannot be tested

D) Use telnet from PE router to PE router

Summary

After completing this section, you should be able to perform the following tasks:

- Verify proper PE-to-PE connectivity
- Verify proper redistribution of VPN routes and creation of MPLS labels
- Verify VPN route propagation and data forwarding

Next Steps

After completing this lesson, go to:

- Advanced VRF Import/Export Features

Lesson Review

Instructions

Answer the following questions:

1. What are the preliminary MPLS/VPN troubleshooting steps?
2. How would you verify routing information exchange between PE routers?
3. How would you verify that the VPNv4 routes are entered in the proper VRF?
4. How would you verify redistribution of VPNv4 routes into PE-CE routing protocol?
5. How would you test end-to-end data flow between PE routers?
6. How would you verify that the CE routes get redistributed into MP-BGP with proper route targets?
7. How would you check for potential MTU size issues on the path taken by PE-to-PE LSP?
8. How would you verify that the PE router ingress interface supports CEF switching?

Advanced VRF Import/Export Features

Overview

This lesson describes advanced features available to control VRF import and export processes.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Configure import and export route maps within VRFs
- Configure limits on the number of routes accepted from a BGP neighbor
- Configure limits on the total number of routes in a VRF

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

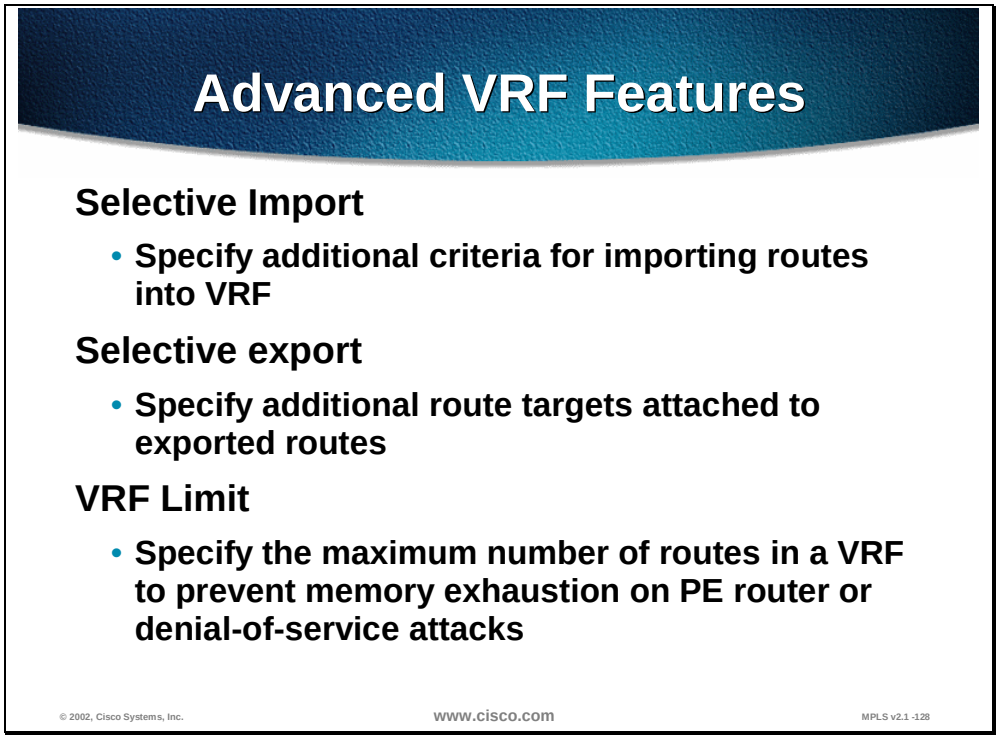
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- Advanced VRF Features
- Configuring Import Route-Map
- Configuring Export Route-Map
- Configuring BGP Per-Neighbor Limits
- Configuring Per-VRF Route Limit
- Summary
- Lesson Review

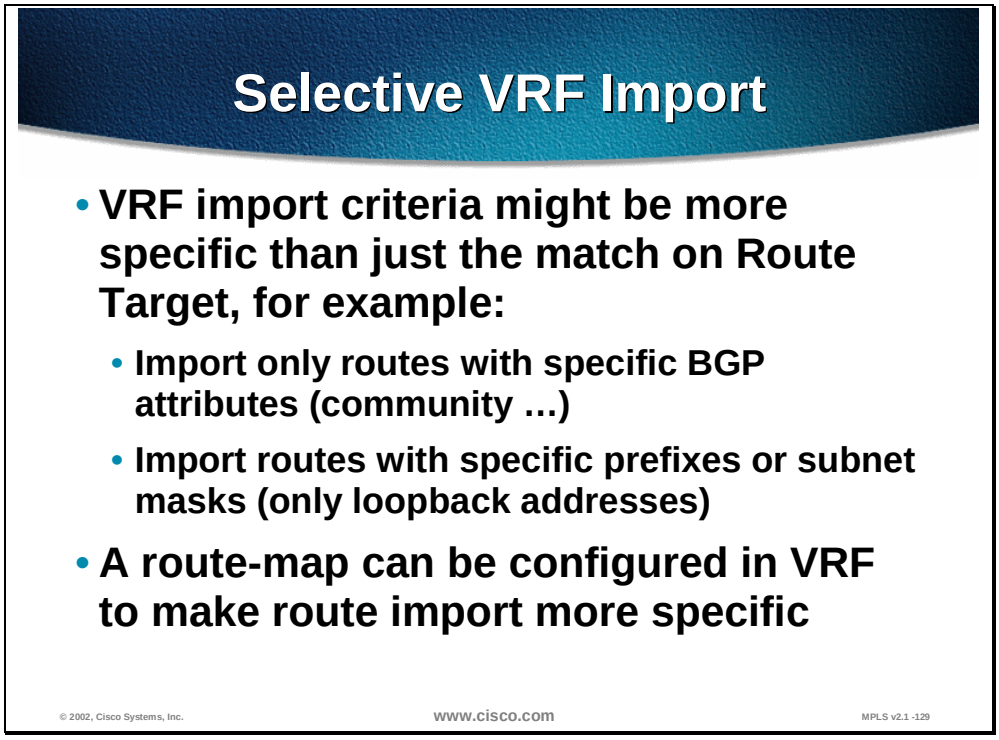
Advanced VRF Features



There are a number of advanced VRF features that allow you to deploy advanced MPLS/VPN topologies or to increase the stability of your MPLS/VPN backbone:

- The **selective import** feature allows you to select routes to be imported into a VRF based on criteria other than route target
- The **selective export** feature allows you to attach specific route targets only to a subset of routes exported from a VRF (by default, the same route targets get attached to all exported routes)
- The **VRF route limit** feature allows you to limit the number of routes the customer (or other PE routers) can insert in the VRF, therefore preventing fatal consequences of configuration errors or denial-of-service attacks.

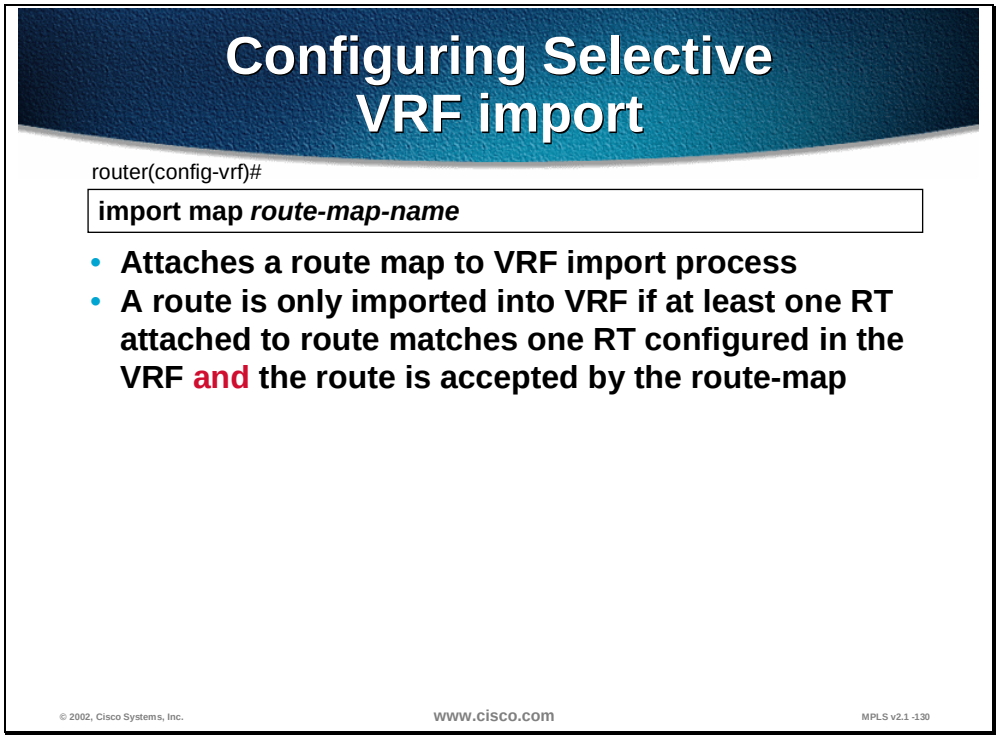
Configuring import route-map



Selective route import into a VRF allows you to narrow the route import criteria by using a route-map that can filter the routes selected by the route-target import filter. The routes imported into a VRF are BGP routes, so you can use match conditions in a route-map to match any BGP attribute of a route, including, for example, communities, local-preference, MED, AS-path, etc.

The import route-map filter is combined with the route-target import filter – a route has to pass the route-target import filter first and then the import route map. The necessary conditions for a route to be imported into a VRF are thus:

- At least one of the route-targets attached to the route matches one of the import route targets configured in the VRF
- The route is permitted by the import route-map.



import map

To configure an import route map for a VRF, use the **import map** command in VRF submode.

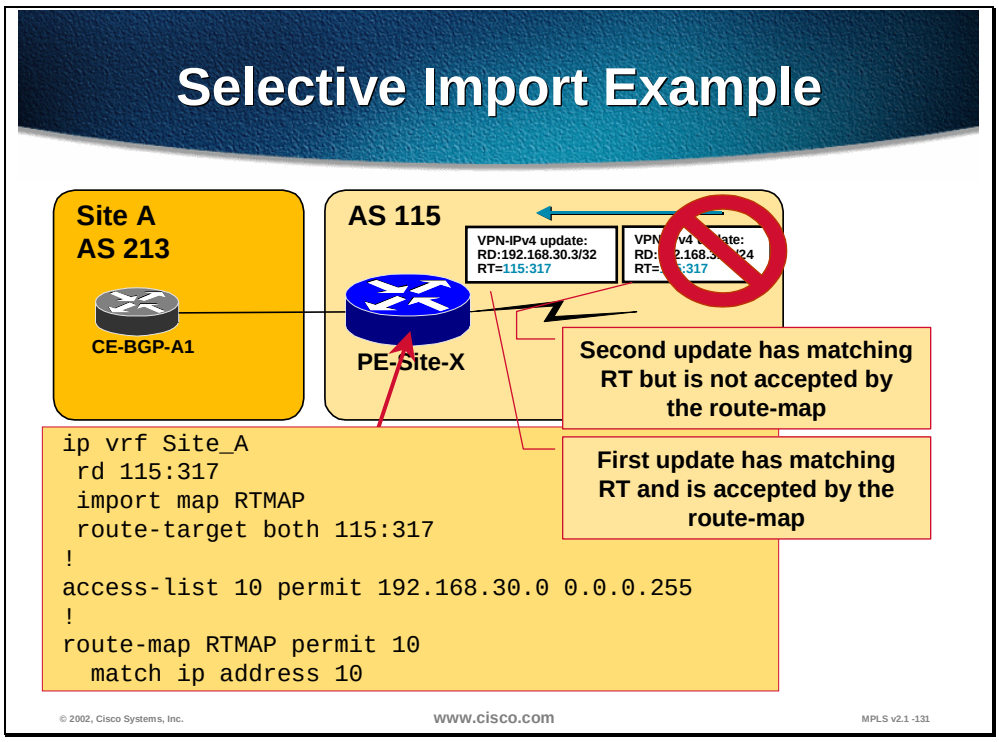
import map *route-map*

Syntax Description

route-map	Specifies the route map to be used as an import route map for the VRF.

Defaults

There is no default. A VRF has no import route map unless one is configured using the **import map** command.



The slide shows an example where an import route-map is used to match the IPv4 portion incoming of VPNv4 routes and import only routes matching a certain prefix into the VRF. A configuration similar to this one could be used to:

- Deploy advanced MPLS/VPN topologies (for example, managed router services topology – see the **MPLS/VPN Topologies** chapter of the **MPLS VPN Solutions** lesson for more details or
- Increase the security of extranet VPN by allowing only predefined subnets to be inserted into a VRF, thus preventing an extranet site from inserting unapproved subnets into the extranet.

Note A similar function is usually not needed in an intranet scenario, because all the customer routers in an intranet are usually under common administration.

Practice

Q1) Why would you need selective VRF import?

A) Selective VRF import is never needed

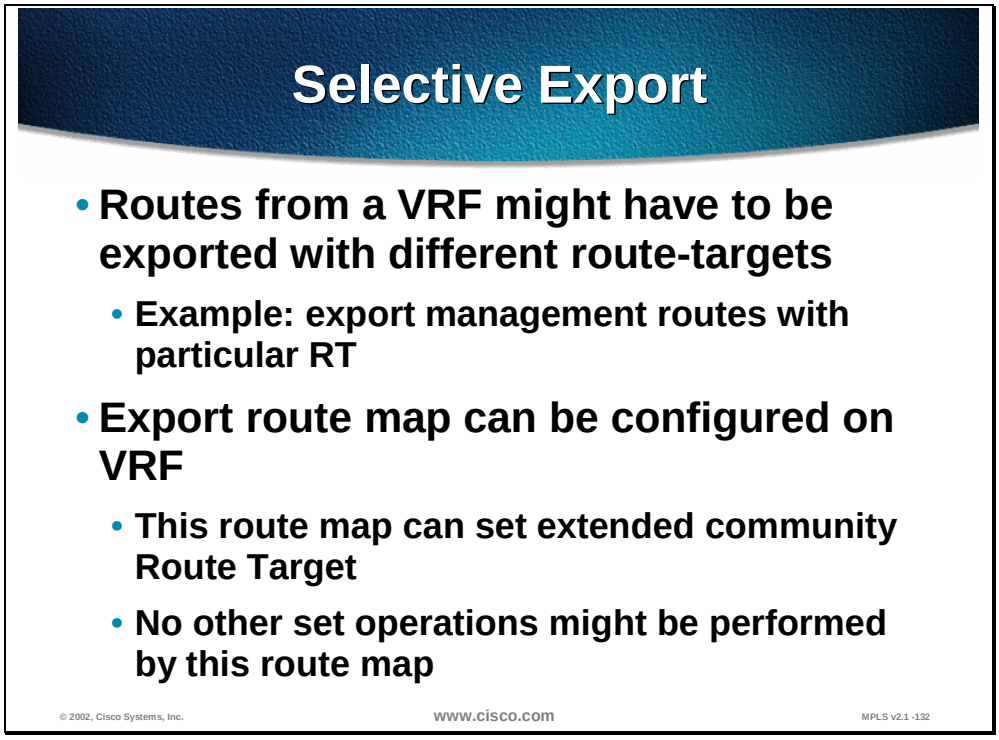
B) To allow customer sites with different forwarding rules to share the same VRF

C) To save bandwidth by limiting the number of routes exchanged using BGP

D) To save CPU by limiting the number of updates send and received by BGP

E) To import only a subset of the otherwise imported routes

Configuring export route-map

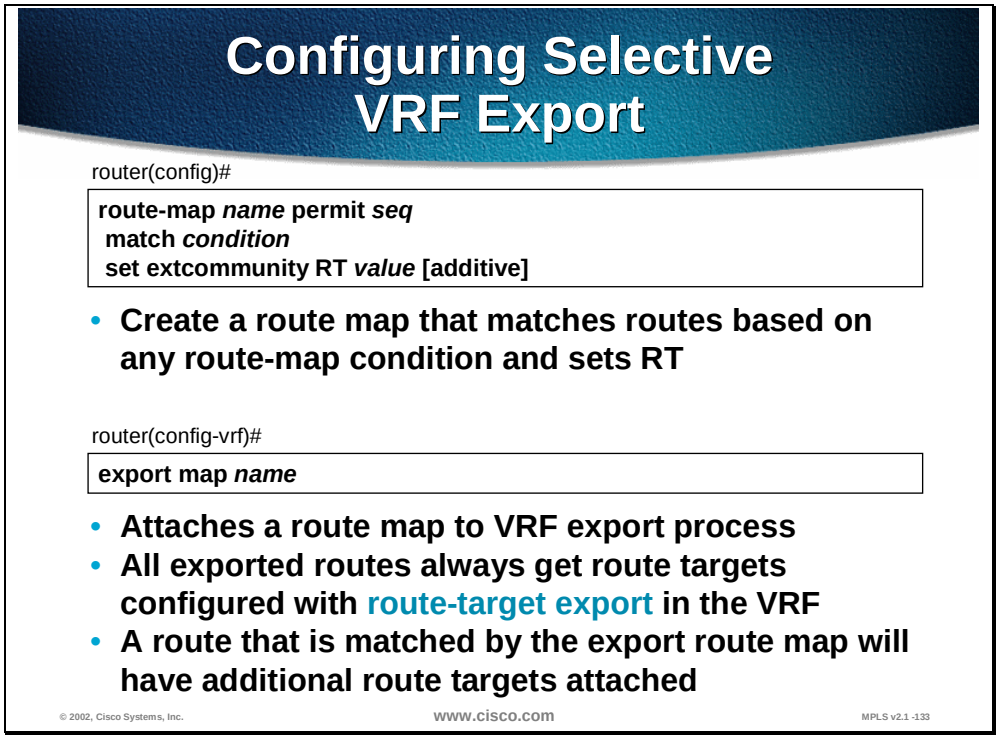


Some advanced MPLS/VPN topologies are easiest to implement if you can attach a variety of route targets to routes exported from the same VRF, so that only a subset of the routes exported from a VRF is imported into another VRF. Most of the services where the customer routers need to connect to a common server, be it a network management station, voice gateway or an application server, fall into this category.

The **export route-map** function provides exactly this functionality – a route map can be specified for each VRF to attach additional route targets to routes exported from a VRF. The export route-map performs only the attachment of route targets, it does not perform any filtering function and you cannot change any other route attributes with this route-map.

Attributes attached to a route with an export route-map are combined with the export route-target attributes. If you specify export route-targets in a VRF and set route targets with an export route-map, all of the specified route targets are attached to the exported route.

Note Export route-map provides functionality that is almost identical to the import route-map, but applied to a different VRF. Any requirement that can be implemented with an export route-map can also be implemented with an import route-map, but usually in a more awkward manner.



set extcommunity

To set the extended communities attribute, use the **set extcommunity** route-map configuration command. To delete the entry, use the **no** form of this command.

set extcommunity *extcommunity-type community-number* [**additive**]
no set extcommunity *extcommunity-type community-number* [**additive**]

Syntax Description

extcommunity-type	Valid parameters are **rt** (Route Target) and **soo** (Site of Origin).
extcommunity-number	Valid parameter is entered in a *x:y* format where *x* can either be an AS number (1-65535) and *y* is in the range from 1 to 4294967200 or *x* is an IP address where *y* is in the range from 1 to 65535.
additive	(Optional) Adds the extended community to the already existing extended communities.

Default

No BGP extended community attributes are set by the route map.

export map

To apply a route map to filter and modify exported routes, use the **export map** VRF configuration command. To remove the route map from the VRF, use the **no** form of this command.

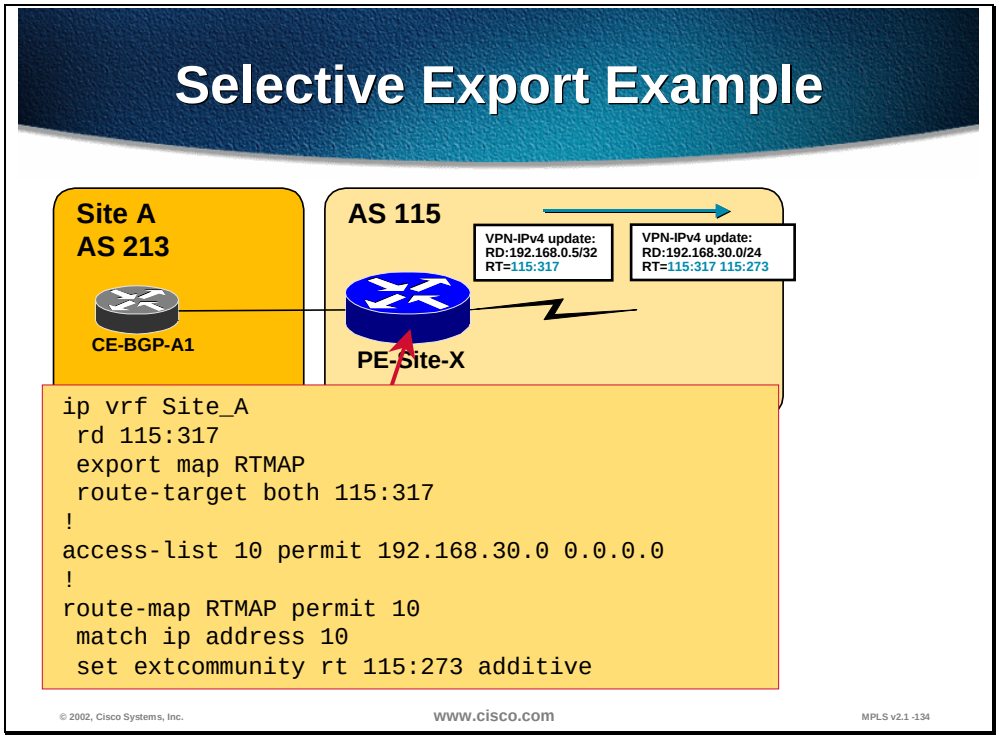
export map *route-map-name*
no export map *route-map-name*

Syntax Description

route-map-name	specify the name of the route map to be used.

Default

No route map is used.



This example mirrors the example from page 162, this time implemented with an export-map. In the example on page 162, the selective import of routes into a VRF was achieved with an import route-map in the receiving VRF that allowed only routes from a certain address block to be inserted into the VRF. In this example, routes from certain address block are marked with an additional route-target in the originating VRF and are automatically inserted into the receiving VRF based on their route target.

The main difference between import and export route-map is therefore the deployment point:

- Import route-map is deployed in the receiving VRF
- Export route-map is deployed in the originating VRF

Based on your network design, one or the other functionality might be preferred.

Practice

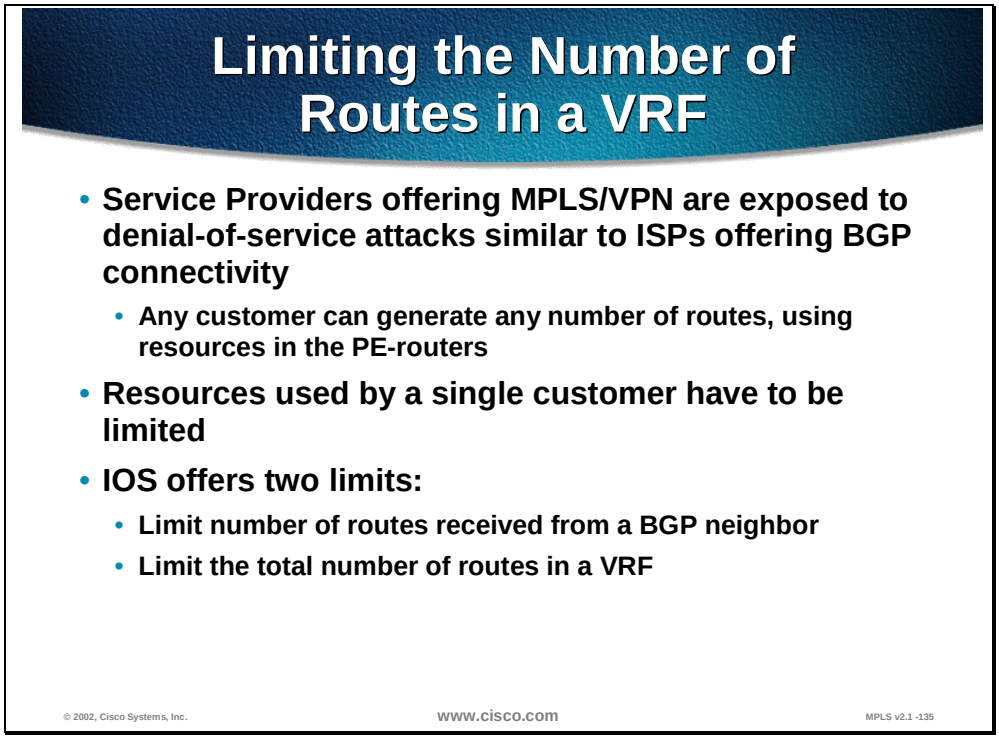
Q1) Why would you need selective VRF export?

A) To allow customer sites with different forwarding rules to share the same VRF

B) To save bandwidth by limiting the number of routes exchanged using BGP

C) Selective VRF exportis never needed

D) To assign a route-target to only a subset of the otherwise imported routes



MPLS/VPN architecture achieves a very tight coupling of customer and the service provider network, resulting in a number of advantages. The tight coupling might also result in a few disadvantages because the service provider network is all of a sudden exposed to design and configuration errors in customer networks, as well as to a number of new denial-of-service attacks based on routing protocol behavior.

To limit the effect of configuration errors as well as malicious user behavior, Cisco IOS offers two features that limit the number of routes (and consequently resource consumption at a PE router) that a VPN user can have:

- The BGP maximum-prefix feature limits the number of routes that an individual BGP peer can send
- The VRF route limit limits the total number of routes in a VRF, regardless of whether they are received from CE routers or from other PE routers via MP-IBGP

Practice

Q1) Why would you need VRF route limit?

A) To protect the PE router from running out of CPU resources due to intensive updates from the CE router

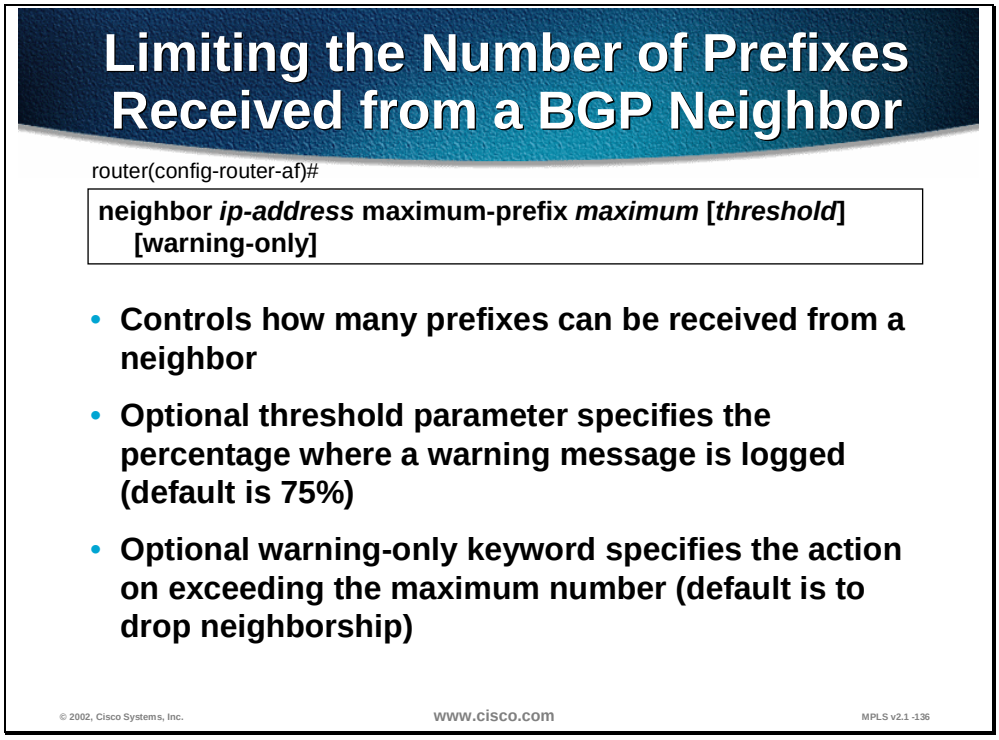
B) To advice Cisco IOS not to reserve too much memory

C) To avoid installing the 96 bit large VPNv4 routes in the VRF routing table

D) To protect the PE router from running out of memory due to misconfiguration in the customer network

E) To protect the CE router from running out of memory due to misconfiguration in the customer network

Configuring BGP Per-Neighbor Limits



neighbor maximum-prefix

To control how many prefixes can be received from a neighbor, use the **neighbor maximum-prefix** router configuration command. To disable this function, use the **no** form of this command.

neighbor {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum* [*threshold*]
[**warning-only**]
no neighbor {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum*

Syntax Description

ip-address	IP address of the neighbor.
peer-group-name	Name of a BGP peer group.
maximum	Maximum number of prefixes allowed from this neighbor.
threshold	(Optional) Integer specifying at what percentage of *maximum* the router starts to generate a warning message. The range is 1 to 100; the default is 75 (percent).
warning-only	(Optional) Allows the router to generate a log message when the *maximum* is exceeded, instead of terminating the peering.

Default

Disabled; there is no limit on the number of prefixes.

Practice

Q1) When would you want to use BGP maximum-prefix parameter?

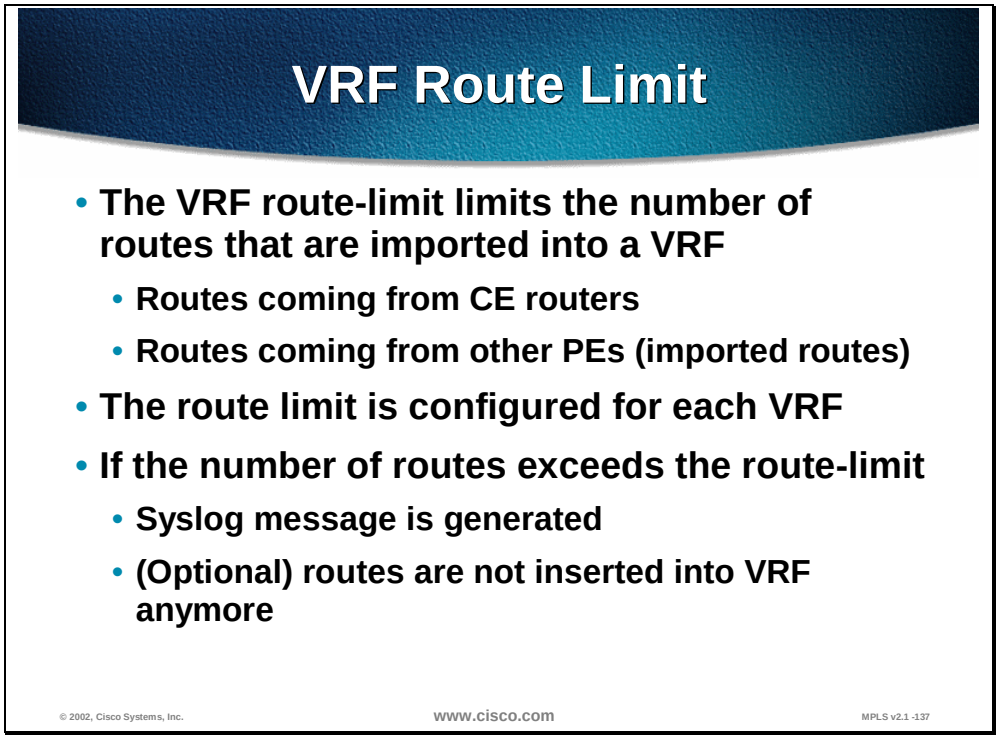
A) To protect the PE router from being overwhelmed with BGP routes from a misconfigured customer router

B) To protect the P routers from being overwhelmed with BGP routes from a misconfigured customer router

C) To avoid sending too many BGP prefixes to a neighboring router due to misconfiguration

D) To stop Internet from growing out of control

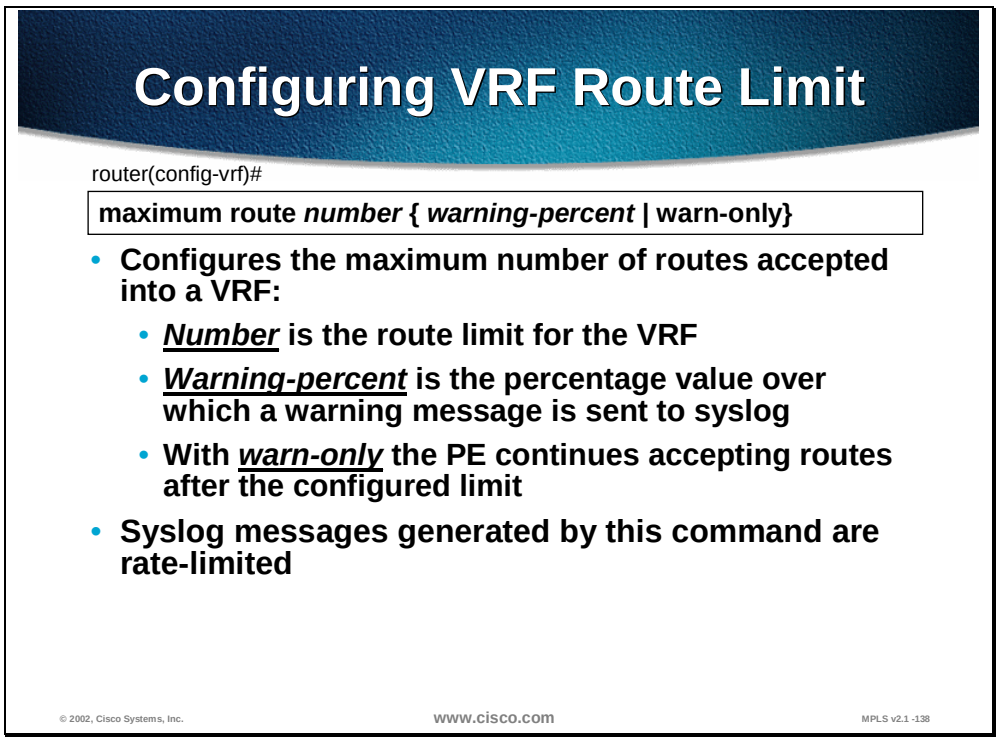
Configuring Per-VRF Route Limit



The VRF route limit, contrary to the BGP maximum-prefix limit, limits the overall number of routes in a VRF, regardless of their origin. Similar to BGP maximum-prefix, the network operator might be warned when the number of routes exceeds a certain threshold via the syslog mechanism. Additionally, you can configure IOS to ignore new VRF routes when the total number of routes exceeds the maximum configured limit.

The route limit is configured for each individual VRF, giving you maximum design and configuration flexibility.

Note The per-VRF limit could be used to implement add-on MPLS/VPN services, where a user paying for a better service might be able to insert more VPN routes into the network.



maximum routes

To limit the maximum number of routes in a VRF to prevent a PE router from importing too many routes, use the **maximum routes** command in VRF submode. To remove the limit on the maximum number of routes allowed, use the **no** form of this command.

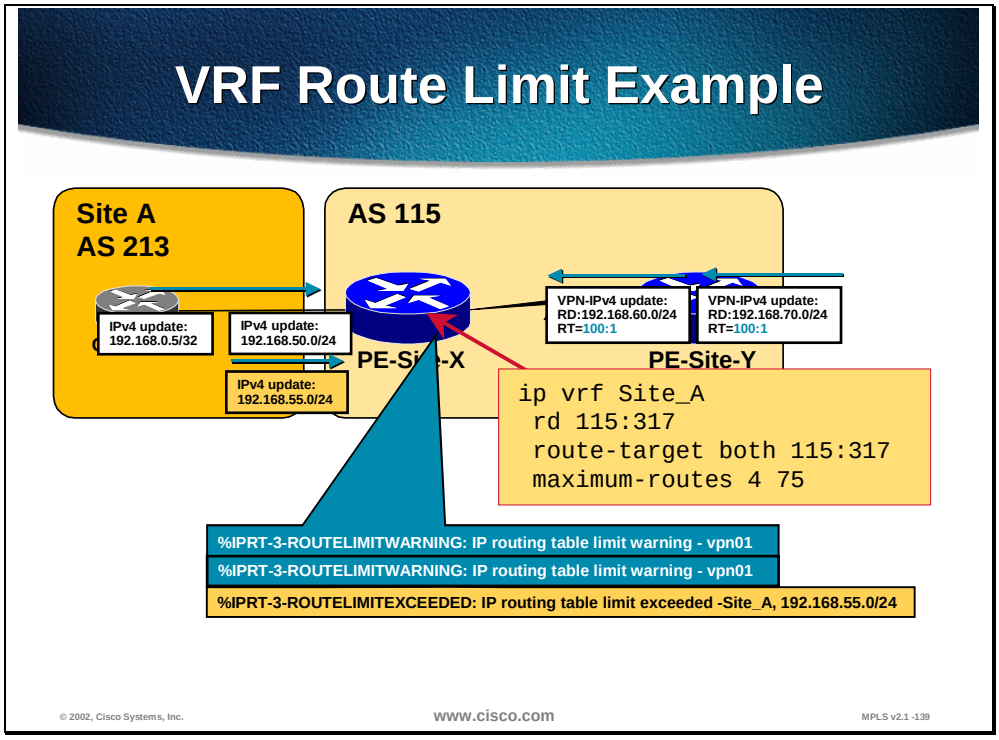
maximum routes *limit* {*warn threshold* | **warn-only**}
no maximum routes

Syntax Description

limit	Specifies the maximum number of routes allowed in a VRF. You may select from 1 to 4,294,967,295 routes to be allowed in a VRF.
warn threshold	Rejects routes when the threshold limit is reached. The threshold limit is a percentage of the limit specified, from 1 to 100.
warn-only	Issues a SYSLOG error message when the maximum number of routes allowed for a VRF exceeds the threshold. However, additional routes are still allowed.

Defaults

No default behavior or values.



In this example, the network designer has decided to limit the number of routes in a VRF to four, with the warning threshold being set at 75% (or three routes).

When the first two routes are received and inserted in the VRF, the router accepts them. When the third route is received, a warning message is generated and the message is repeated with the insertion of the fourth route.

Note The SYSLOG messages are rate-limited to prevent indirect denial-of-service attacks on the network management station

When the PE router receives the fifth route, the maximum route limit is exceeded and the route is ignored. The network operator is notified through another syslog message.

Practice

Q1) When would you want to use VRF route-limit?

A) To avoid consuming too much CPU and bandwidth resources by limiting the number of routes being exchanged with a VRF per second

B) To instruct the show ip route vrf command to paginate the output

C) To protect the P routers from being overwhelmed with customer routes due to misconfiguration

D) To avoid sending too many routes from the PE routers into the customer network due to misconfiguration

E) To avoid too many routes in the PE router's VRF due to misconfiguration in the customer network

Summary

After completing this section, you should be able to perform the following tasks:

- Configure import and export route maps within VRFs
- Configure limits on the number of routes accepted from a BGP neighbor
- Configure limits on the total number of routes in a VRF

Next Steps

After completing this lesson, go to:

- Advanced PE-CE BGP Configuration

Lesson Review

Instructions

Answer the following questions:

1. Why would you need the selective VRF import command?
2. How does the import route-map affect VRF import process?
3. Why would you need the selective VRF export command?
4. How does the export route-map affect VRF export process?
5. Which BGP attributes can be set with an export route-map?
6. Why would you need the VRF route limit command?
7. How many VRF route-limiting options does IOS offer?
8. When would you want to use the BGP maximum-prefix parameter?
9. When would you want to use the VRF route-limit?

Advanced PE-CE BGP Configuration

Overview

This lesson describes advanced BGP features that are sometimes needed in environments where BGP is used between PE and CE routers.

Importance

This lesson gives the student information on configuring, monitoring and troubleshooting MPLS/VPN technology on Cisco IOS platform and is a mandatory prerequisite for the MPLS/VPN Service Solution lesson.

Objectives

Upon completion of this lesson, the learner will be able to perform the following tasks:

- Describe and properly use the AS-Override feature
- Describe and properly use the AllowAS-in feature
- Configure Site-Of-Origin (SOO) on incoming interface or BGP neighbor

Learner Skills and Knowledge

To fully benefit from this lesson, you must have these prerequisite skills and knowledge:

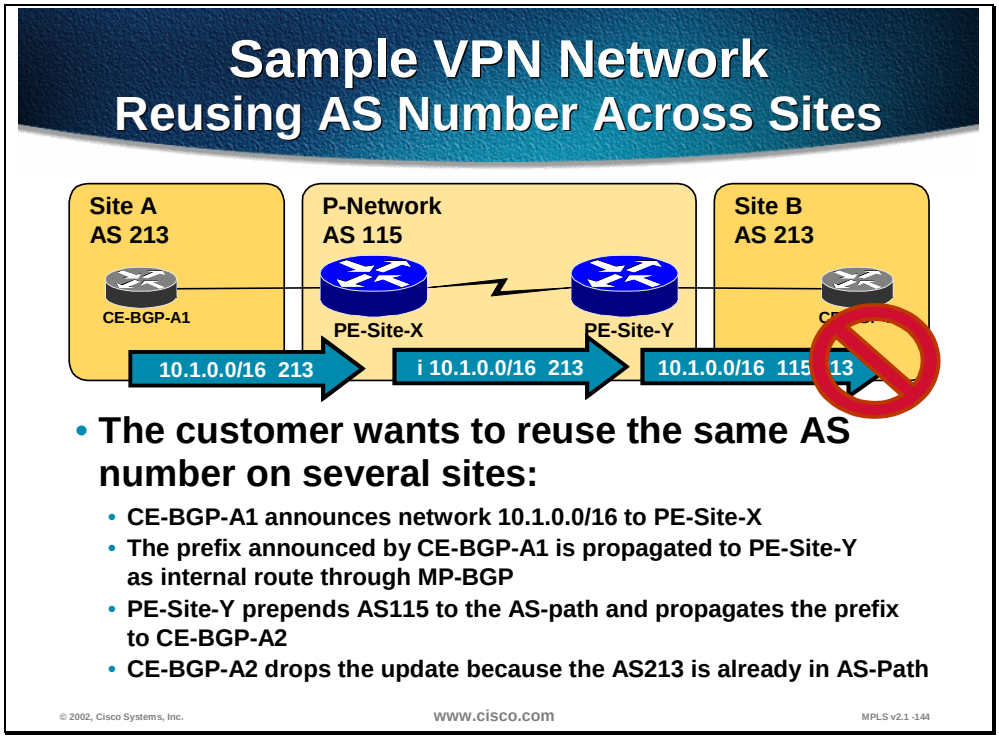
- T_MPLS_VPN module and all associated prerequisites

Outline

This lesson includes these sections:

- Overview
- BGP Issues Arising When the Customer Uses the Same AS Number on Multiple Sites
- Configuring AS-Override Feature
- BGP Issues Arising in Multihomed Customer Sites
- Configuring AllowAS-In Feature
- Configuring Site-of-Origin
- Summary
- Lesson Review

BGP Issues Arising When the Customer Uses the Same AS Number on Multiple Sites

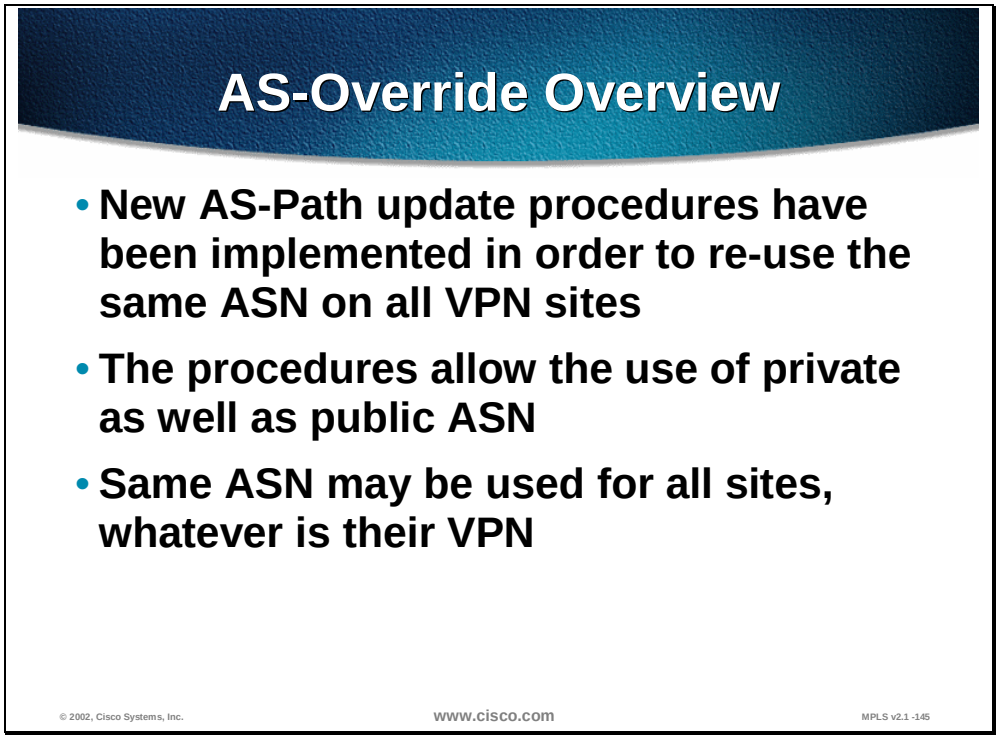


There are two ways an MPLS/VPN customer can deploy the BGP as the routing protocol between the PE and the CE routers:

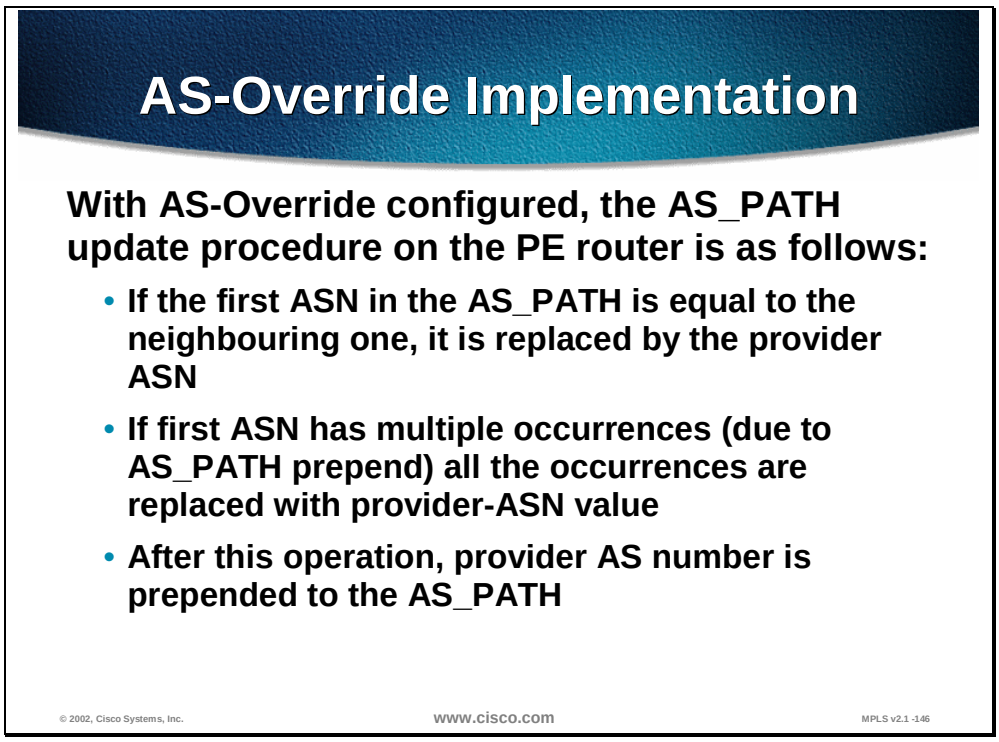
- If the customer has used any other routing protocol in the traditional overlay VPN network before, there are no limitations on the numbering of customer's autonomous systems; every site could be a separate autonomous system
- If, however, the customer has been using BGP as the routing protocol before, there is a good chance that all the sites (or a subset of the sites) were using the same autonomous system number

BGP loop prevention rules disallow discontiguous autonomous systems – in other words, two customer sites with the identical AS number cannot be linked by another autonomous system. If such a setup happens (as in the example above), the routing updates from one site would be dropped when the other site receives them and there would be no connectivity between the sites.

Configuring AS-Override Feature

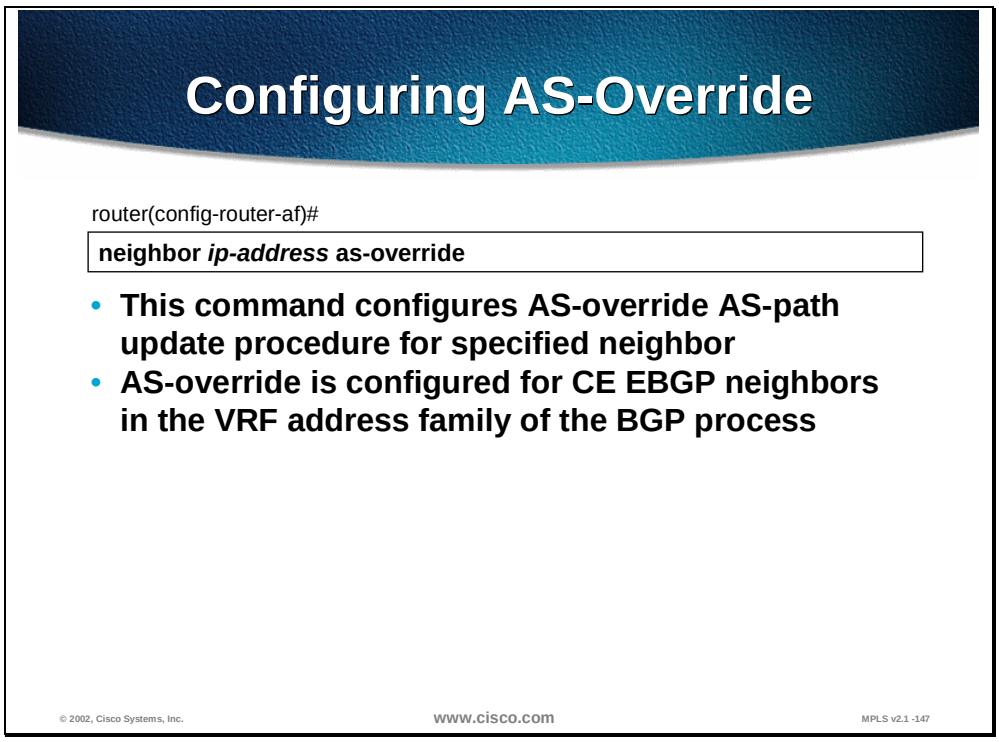


To support customer topologies where the same customer AS number is used at more than one site, the AS-path update procedure in BGP has been modified to overcome the loop prevention rules of BGP. The new AS-path update procedure supports usage of one AS number at many sites (even between several overlapping VPNs) and does not rely on distinction between private or public AS numbers.



The modified AS-path update procedure (also called AS-override) is extremely simple:

- The procedure is only used if the first AS number in the AS-path is equal to the AS-number of the receiving BGP router
- In this case, all the leading occurrences of the AS number of the receiving BGP router are replaced with the AS number of the sending BGP router. Any other occurrences (further down the AS path) of the receiving router's AS number are not replaced because they indicate a real routing information loop
- An extra copy of the sending router's AS number is prepended to the AS-path (standard AS number prepending procedure that occurs on every EBGP update)



neighbor as-override

To configure a PE router to override a site's ASN with a provider's ASN, use the **neighbor as-override** router configuration command. To remove VPN IPv4 prefixes from a specified router, use the **no** form of this command.

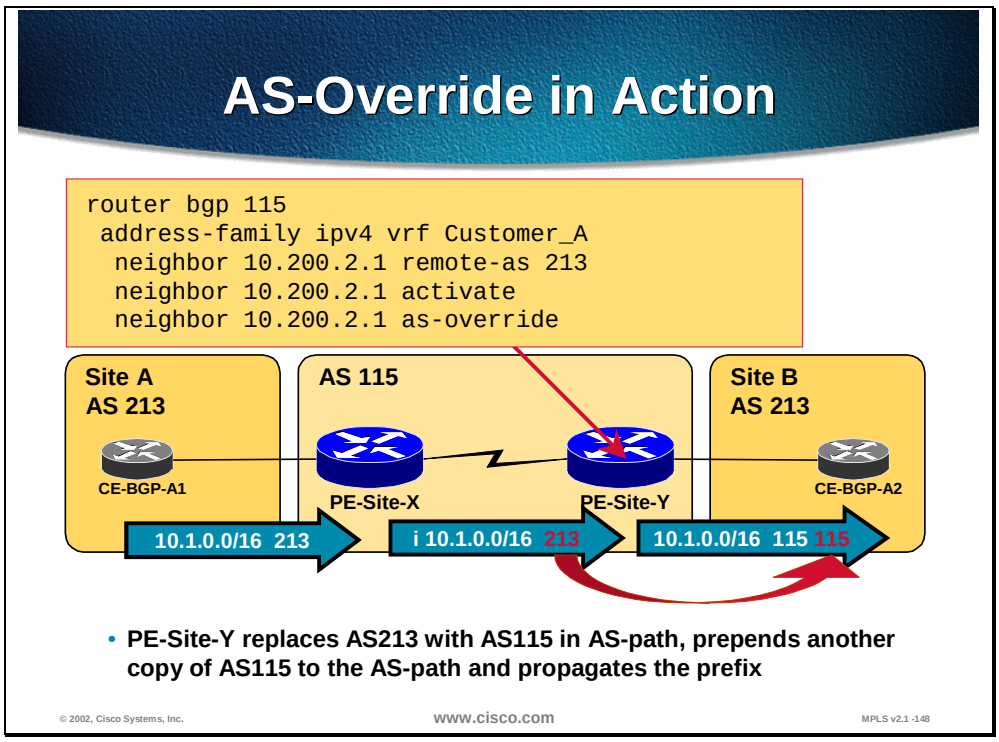
neighbor *ip-address* as-override
no neighbor *ip-address* as-override

Syntax Description

ip-address	Specifies the router's IP address to override with the ASN provided.

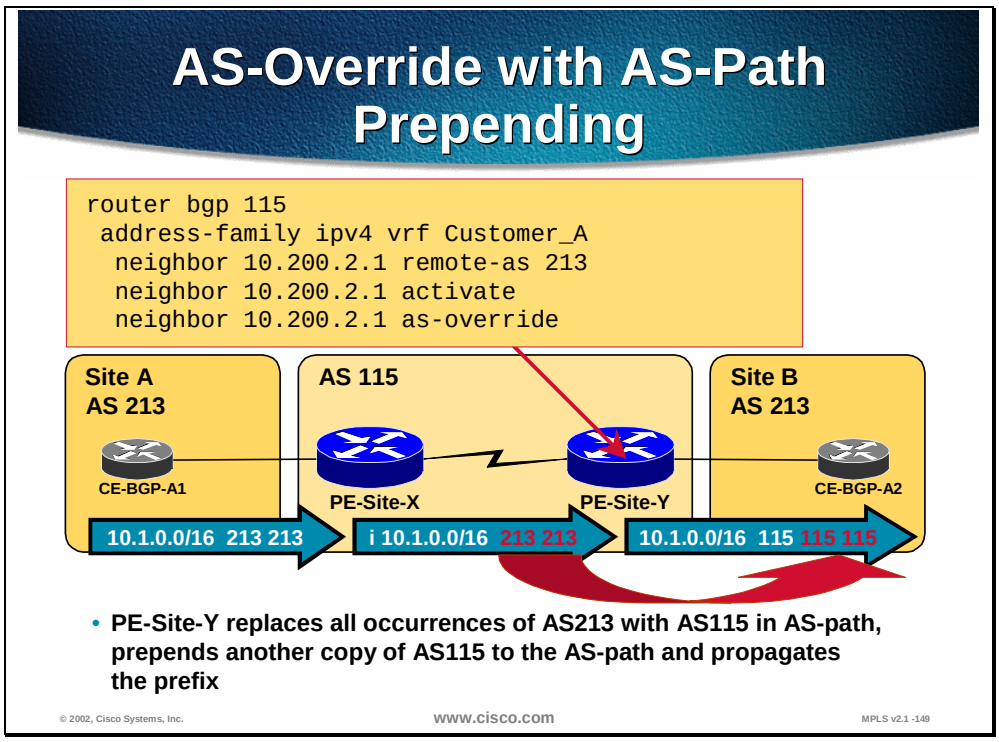
Defaults

No default behavior or values.



In the example above, two customer sites (Site A and Site B) use BGP to communicate with the MPLS/VPN backbone. Both sites use AS 213 and Site B would drop the update sent by Site A without the AS-override mechanism.

The AS-override mechanism, configured on PE-Site-Y router, replaces the customer AS number (213) with the provider AS number (115) before sending the update to the customer site. An extra copy of the provider AS number is prepended to the AS-path during the standard EBGP update processing.



If the customer is using AS prepending to influence BGP path selection within the MPLS/VPN backbone, the PE router has to send a route with an AS path containing multiple copies of the customer AS number to the CE router. In this case, all the leading copies of the customer AS number are replaced with the provider AS number (resulting in two occurrences of the provider AS number in the example above) and the third occurrence of the provider AS number is prepended to the BGP update before it's sent out to the CE router.

Practice

Q1) How does the AS-override feature work?

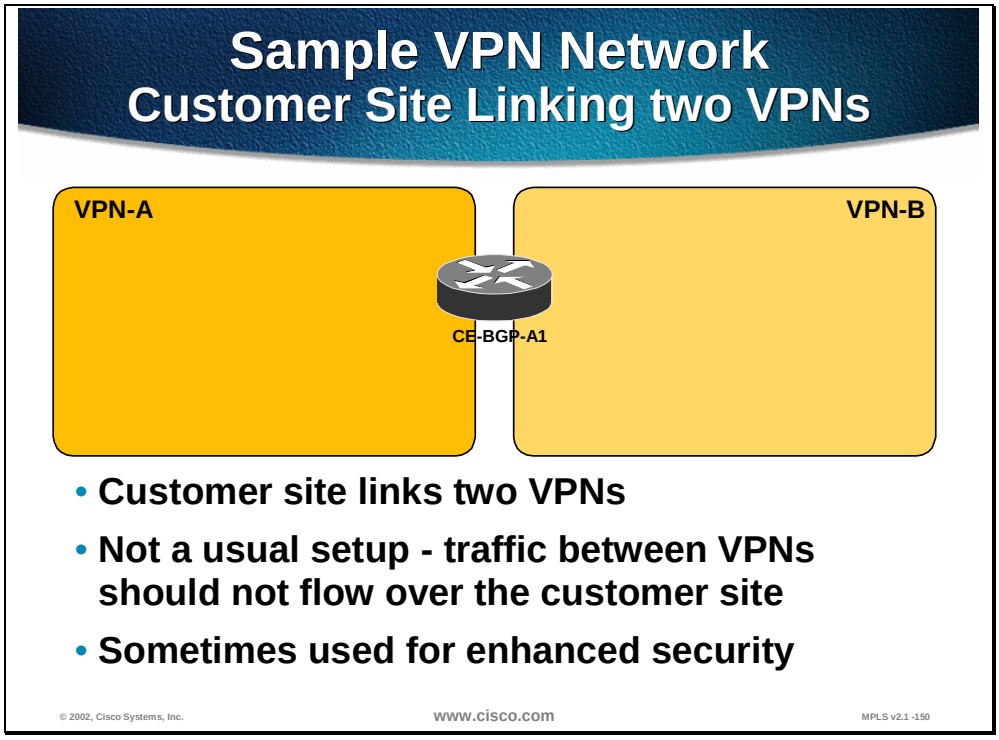
A) The PE router removes the customers AS number on routes recevied from the CE router.

B) The PE router does not prepend its own AS number before sending the routes to the CE router.

C) The PE router does not perform the standard loop prevention test for its own AS in the AS path on received routes.

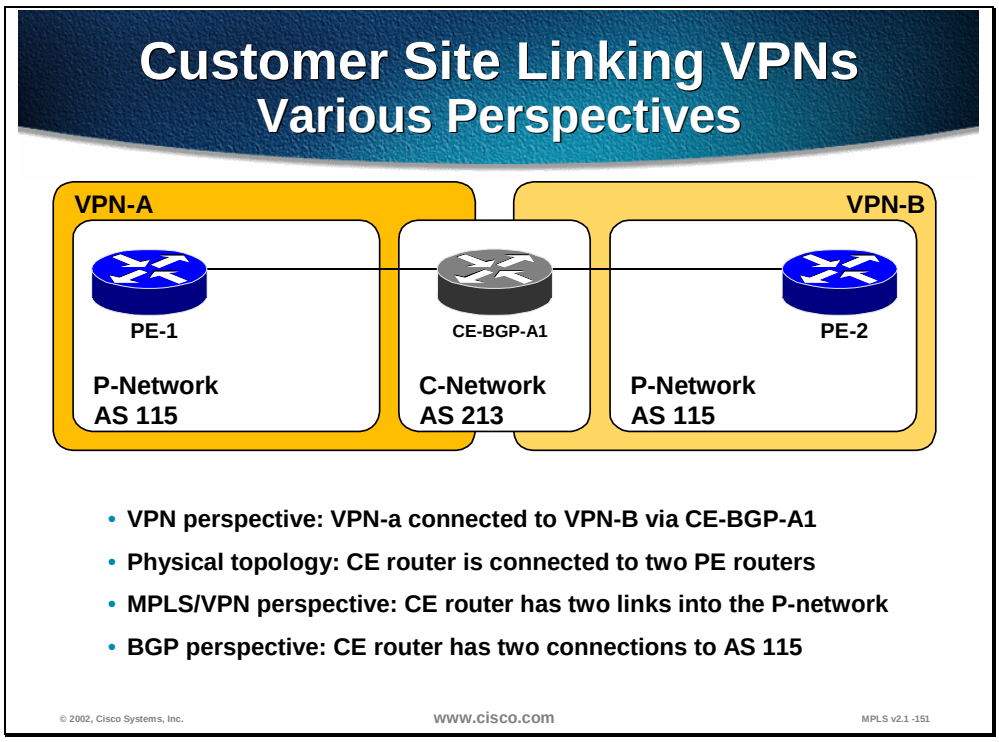
D) The PE router replaces leading occurences of the customer AS number in the AS-path with the provider AS number before sending the routes to the CE router.

BGP Issues Arising in Multihomed Customer Sites



In some security-conscious implementations, customer VPNs are linked by a customer router that performs security functions like access filters or access logging.

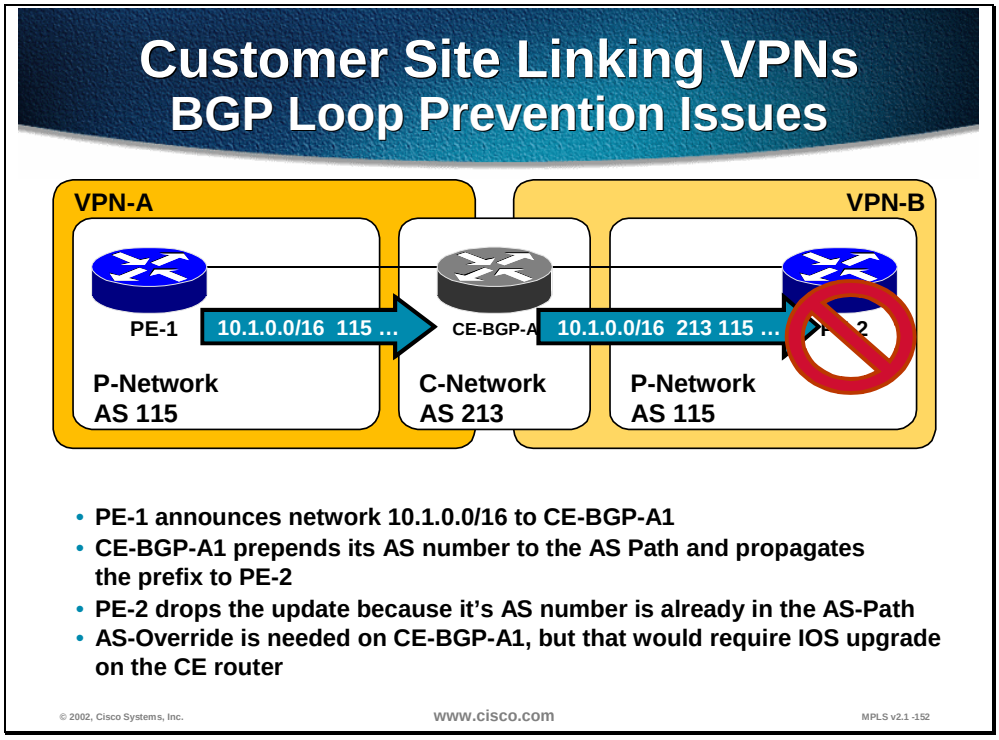
Note	This setup is not a usual setup because it deviates from the basic goal of MPLS/VPN – replace hub-and-spoke routing of traditional overlay VPN with optimum any-to-any routing.



The setup where a customer router links two VPN networks in an MPLS/VPN backbone can be viewed from several different perspectives:

- From the VPN perspective, a CE router links two VPNs
- From the physical perspective, the CE router is connected through two separate links (physical or logical interface) to one or two PE routers. In MPLS/VPN terms, the CE router has two links into the P-network

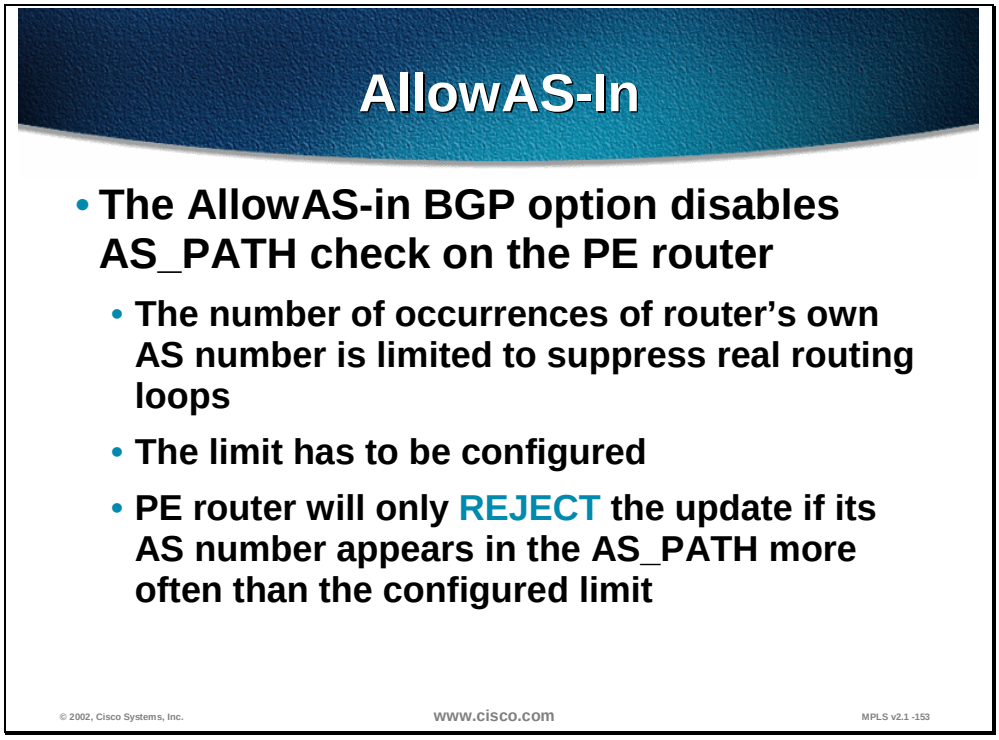
There is no problem with the proposed customer setup if analyzed through these perspectives – they all represent valid connectivity or routing options. The problem occurs when we analyze the BGP perspective, where the CE router has to propagate routes between two PE routers, which are both in the same autonomous system.



Similar to the situation where two customer sites were using the same AS number, BGP loop prevention rules prevent a PE router from accepting the routing update sent by the CE router if that routing update already contains the AS number of the MPLS/VPN backbone (which it will if the CE router is propagating routes between two VPNs).

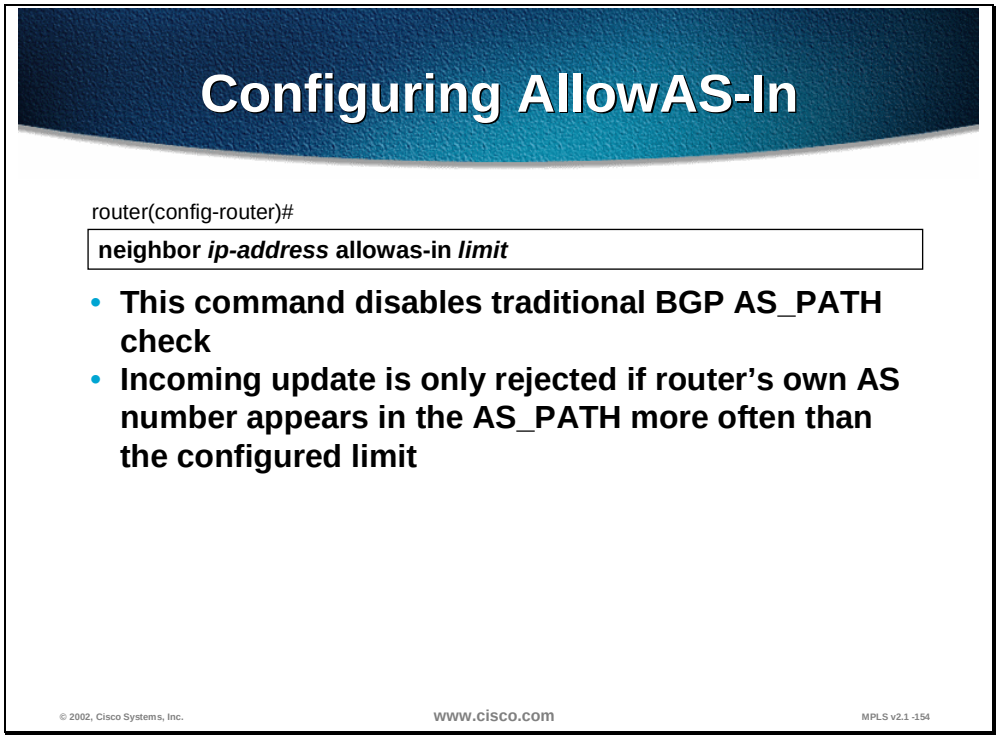
The solution to this BGP routing problem could be identical to the previous one – AS-override has to be used on the CE router. This solution would, however, require a very recent IOS version (12.0T or 12.1 IOS release) on the CE router and is therefore not enforceable in every customer situation.

Configuring AllowAS-In Feature



To support topologies where a CE router with no AS-override support links two VPNs, the BGP loop prevention mechanism on the PE routers was modified to support the situations where the PE router would receive routes with its own AS number already in the AS path.

With the **allowas-in** feature configured on a BGP neighbor of the PE router, the PE router would not drop incoming BGP updates with its AS number in the AS path if they are received from that neighbor. To prevent real BGP routing information loops, the number of occurrences of the MPLS/VPN backbone AS number can be limited and the incoming updates that exceed the limit are dropped.



neighbor allowas-in

To configure PE routers to allow readvertisement of all prefixes containing duplicate ASNs, use the **neighbor allowas-in** command in router configuration mode. To disable the readvertisement of a PE router's ASN, use the **no** form of this command.

neighbor allowas-in *number*
no neighbor allowas-in *number*

Syntax Description

number	Specifies the number of times to allow the advertisement of a PE router's ASN. Valid values are from 1 to 10 times.

Defaults

No default behavior or values.

Practice

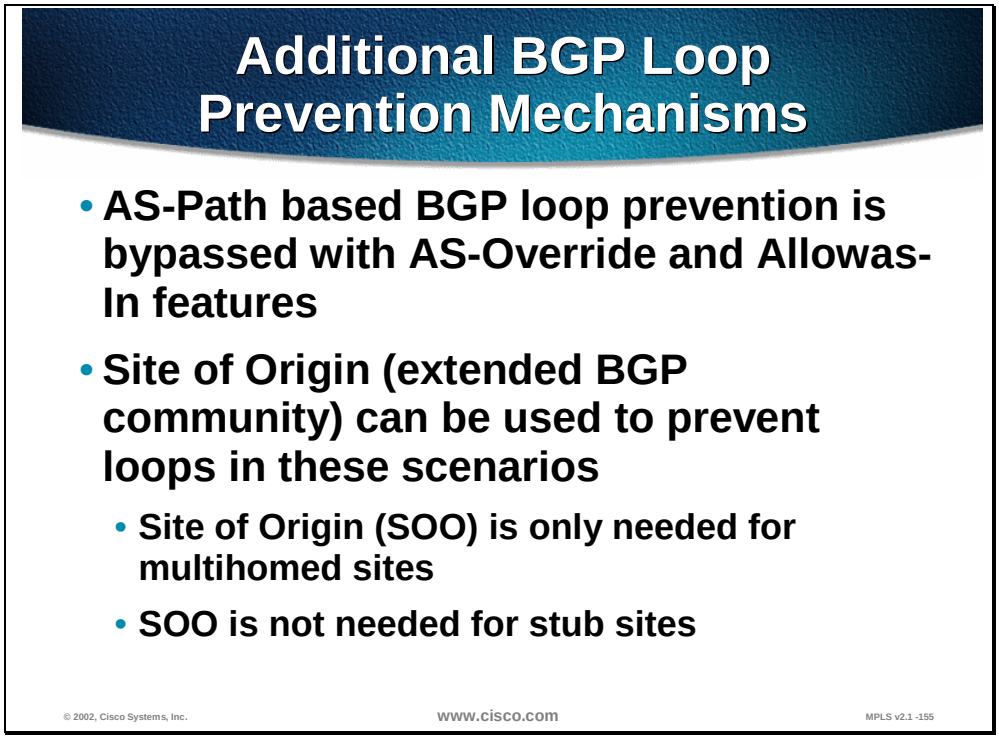
Q1) Why can't you use the AS-override feature instead of the AllowAS-In feature?

A) AS-override might not detect all possble BGP loops which AllowAS-in does.

B) AS-override and AllowAS-in can be used interchangable.

C) AS-override operates on outgoing routes and hence would have to applied on the CE router.

D) AS-override is less efficiently implemented in the router.



Most aspects of BGP loop prevention are bypassed when you're using either **as-override** or **allowas-in** features. Although the routing information loops can still be detected by counting occurrences of an autonomous system number in the AS path in end-to-end BGP routing scenario, the situation can get worse when BGP is mixed with other PE-CE routing protocols. The **Site-of-origin** extended BGP community can be used as an additional loop prevention mechanism in these scenarios.

Note	Site-of-origin and any other loop prevention mechanisms are only needed for customer networks with multi-homed sites where the PE-CE routing protocol does not have any loop detection mechanisms (RIP) or have them turned off (BGP allow-as and/or as-override). Loops can never occur in customer networks that only have stub sites.

Practice

Q1) When would you have to use Site-of-Origin?

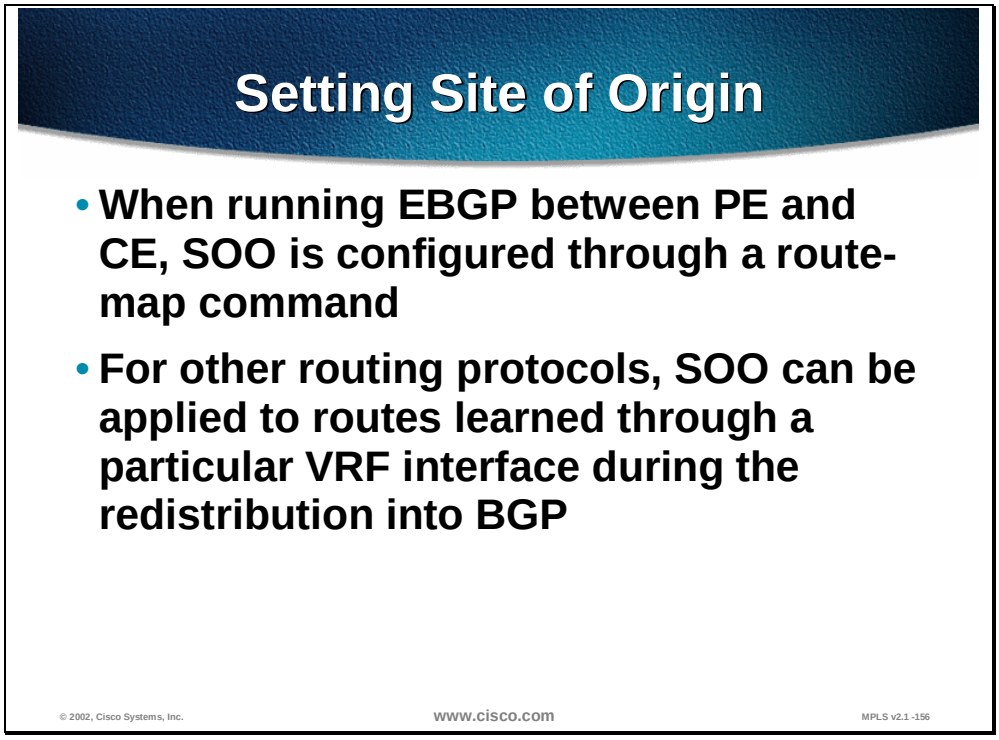
A) You always have to use site-of-origin when using MPLS/VPN.

B) Site-of-origin is used for cosmetical reasons only.

C) To prevent BGP loops when using AS-override and/or AllowAS-in.

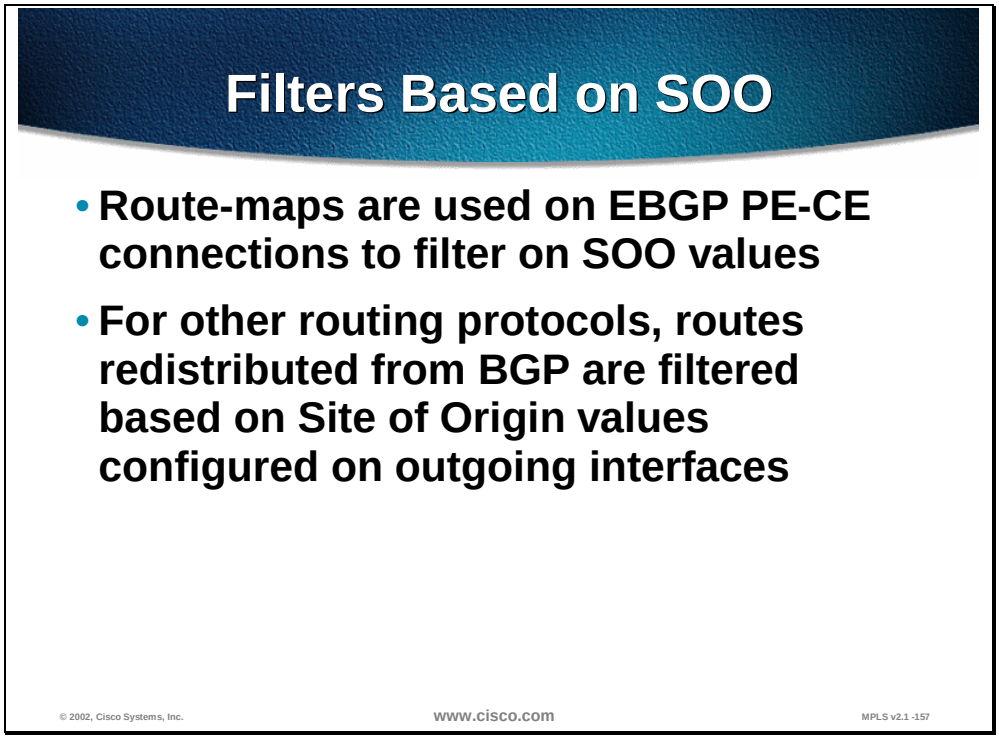
D) To maintain information about customer sites.

Configuring Site-of-Origin



There are two ways to set site-of-origin attribute on a BGP route:

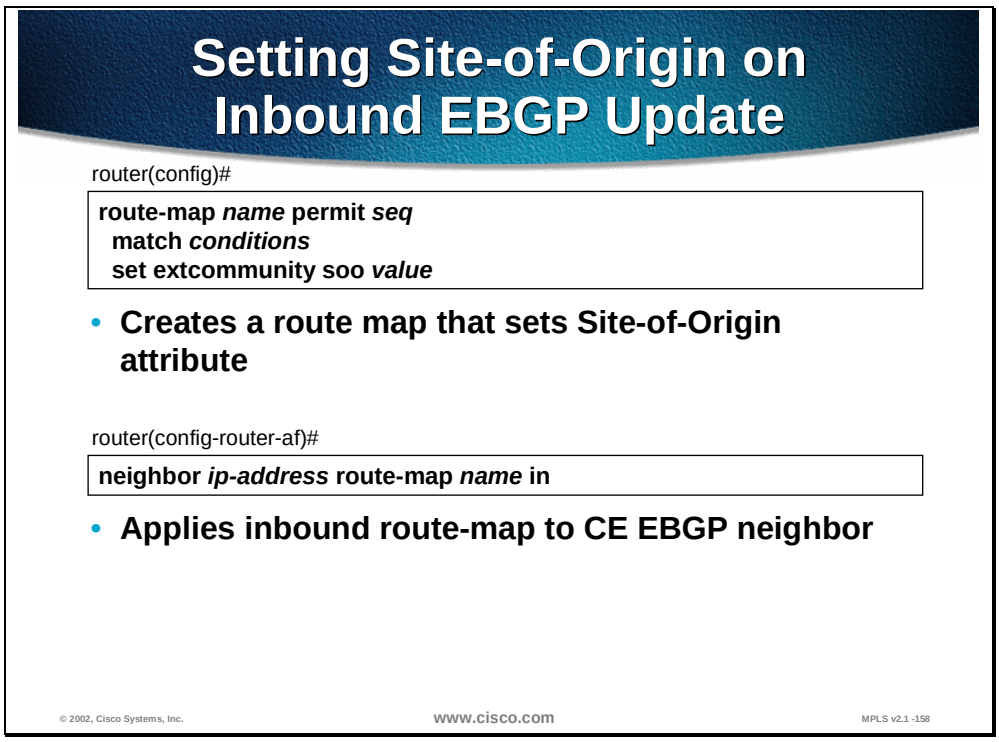
- For routes received from the BGP-speaking CE routers, the site-of-origin is set by incoming route map on the PE-router
- For all other routes, a route-map setting site-of-origin is applied to the incoming interface and the site-of-origin as set by the route-map is attached to the BGP route when an IGP route received through that interface is redistributed into BGP.



Outgoing filters based on site-of-origin attribute also depend on the routing protocol used:

- For situations where BGP is used as the PE-CE routing protocol, outbound route maps can be used on the PE router to deny routes matching particular value of site-of-origin
- For all other routing protocols, filtering is performed based on site-of-origin route-map configured on the outgoing interface before the update is sent across that interface to the CE router

Note SOO is not useful when OSPF is used as the PE-CE routing protocol. OSPF has other loop detection mechanisms for multihomed sites.



set extcommunity

To set the extended communities attribute, use the **set extcommunity** route-map configuration command. To delete the entry, use the **no** form of this command.

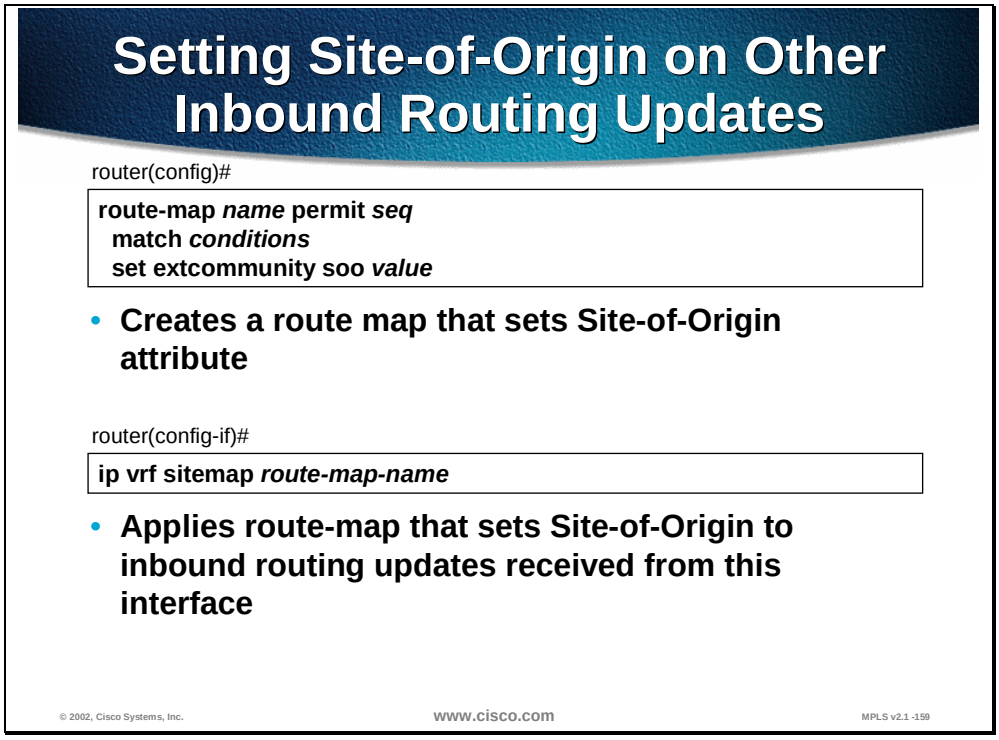
set extcommunity *extcommunity-type community-number* [**additive**]
no set extcommunity *extcommunity-type community-number* [**additive**]

Syntax Description

extcommunity-type	Valid parameters are **rt** (Route Target) and **soo** (Site of Origin).
extcommunity-number	Valid parameter is entered in a *x:y* format where *x* can either be an AS number (1-65535) and *y* is in the range from 1 to 4294967200 or *x* is an IP address where *y* is in the range from 1 to 65535.
additive	(Optional) Adds the extended community to the already existing extended communities.

Default

No BGP extended community attributes are set by the route map.



The **sitemap** concept is useful only when RIP is used as the PE-CE routing protocol.

ip vrf sitemap

To set the Site of Origin extended community attribute, use the **ip vrf sitemap** interface configuration command. To delete the entry, use the **no** form of this command.

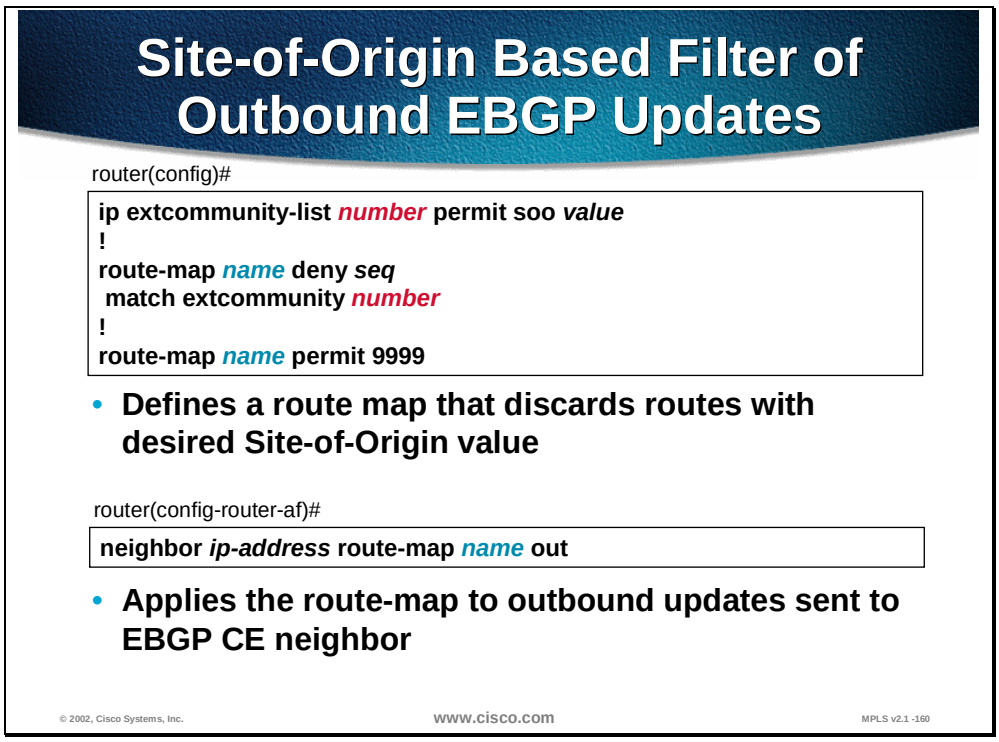
ip vrf sitemap *route-map-name*
no ip vrf sitemap *route-map-name*

Syntax Description

route-map-name	Set the name of the route map to be used.

Default

No route map is used to set the Site of Origin extended community.



In this example, a route-map matching a specific site-of-origin value was defined using **ip extcommunity-list** to establish a site-of-origin filter and **route-map** command to define the route-map based on the filter.

The newly defined route-map is then applied to a BGP neighbor (CE router) on the PE router.

Note	When prefixes are being advertised to the CE site, the SOO present in the attribute is checked against the SOO that belongs to the CE site. If a loop exists (if the SOO values are the same), prefixes are not advertised and the above outbound route-map is no longer needed.

Practice

Q1) What is Site-of-Origin?

A) An extended community BGP attribute.

B) A unique numerical value which has to be assigned to each customer site.

C) A MIB variable readable using SNMP.

D) A BGP session parameter.

Summary

After completing this section, you should be able to perform the following tasks:

- Describe and properly use the AS-Override feature
- Describe and properly use the AllowAS-in feature
- Configure Site-Of-Origin (SOO) on incoming interface or BGP neighbor

Lesson Review

Instructions

Answer the following questions:

1. When would you need the AS-override feature?
2. How does the AS-override feature work?
3. When would you need the AllowAS-In feature?
4. Why can’t you use the AS-override feature instead of AllowAS-In feature?
5. How do you prevent BGP loops when using AS-override?
6. How do you prevent BGP loops when using AllowAS-in?
7. When would you have to use Site-of-Origin?
8. What is Site-of-Origin?
9. Where can you set the Site-of-Origin?
10. How do you implement filters based on Site-of-Origin?

Summary

After completing this lesson, you should be able to perform the following tasks:

- Configure Virtual Routing and Forwarding tables
- Configure Multi-protocol BGP in MPLS/VPN backbone
- Configure PE-CE routing protocols
- Configure advanced MPLS/VPN features
- Monitor MPLS/VPN operations
- Troubleshoot MPLS/VPN implementation

