



# MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

---

**First Published:** February 19, 2007

**Last Updated:** February 19, 2007

This document describes how to configure a Virtual Private Network (VPN) routing and forwarding (VRF) instance for IPv4 and IPv6 VPNs and describes how to upgrade your existing single-protocol IPv4-only VRF to a multiprotocol VRF configuration.

The MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature introduces Cisco IOS command-line interface (CLI) commands that allow you to enable an IPv4 and IPv6 VPN in the same VRF instance and to simplify the migration from a single-protocol VRF configuration to a multiprotocol VRF configuration.

## Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the [“Feature Information for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs”](#) section on [page 27](#).

## Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

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

## Contents

- [Prerequisites for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs, page 2](#)
- [Restrictions for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs, page 2](#)
- [Information About MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs, page 2](#)
- [How to Configure MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs, page 6](#)
- [Configuration Examples for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs, page 14](#)
- [Additional References, page 17](#)
- [Command Reference, page 18](#)



- [Feature Information for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs, page 27](#)
- [Glossary, page 29](#)

## Prerequisites for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

The MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature has the following prerequisites:

- For migration—An IPv4 Multiprotocol Label Switching (MPLS) VPN VRF must exist.
- For a new VRF configuration—Cisco Express Forwarding and an MPLS label distribution method, either Label Distribution Protocol (LDP) or MPLS Traffic Engineering (TE), must be enabled on all routers in the core, including the provider edge (PE) routers.

## Restrictions for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

The MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature has the following restrictions:

- Once you have converted to a multiprotocol VRF, you cannot convert the VRF back to an IPv4-only single-protocol VRF.
- You can associate an interface with only one VRF. You cannot configuring a VRF for IPv4 and a different VRF for IPv6 on the same interface.
- You can configure only IPv4 and IPv6 address families in a multiprotocol VRF. Other protocols (IPX, Appletalk, and the like) are not supported.

## Information About MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

Before you use the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature to migrate from a single-protocol VRF to a multiprotocol VRF, you should understand the following:

- [VRF Concepts Similar for IPv4 and IPv6 VPNs, page 2](#)
- [Single-Protocol VRF to Multiprotocol VRF Migration, page 3](#)
- [Multiprotocol VRF Configurations Characteristics and Examples, page 4](#)

## VRF Concepts Similar for IPv4 and IPv6 VPNs

VPNs for IPv6 use the same VRF concepts that IPv4 VPNs use, such as address families, route distinguishers, route targets, and VRF identifiers. Customers that use both IPv4 and IPv6 VPNs might want to share VRF policies between address families. They might want a way to define applicable VRF policies for all address families, instead of defining VRF policies for an address family individually as they do for or a single-protocol IPv4-only VRF.

Prior to Cisco IOS Release 12.2(33)SRB, a VRF applied only to an IPv4 address family. A one-to-one relationship existed between the VRF name and a routing and forwarding table identifier, between a VRF name and a route distinguisher (RD), and between a VRF name and a VPN ID. This configuration is called a single-protocol VRF.

Cisco IOS Release 12.2(33)SRB introduces support for a multiple address-family (multi-AF) VRF structure. The multi-AF VRF allows you to define multiple address families under the same VRF. A given VRF, identified by its name and a set of policies, can apply to both an IPv4 VPN and an IPv6 VPN at the same time. This VRF can be activated on a given interface, even though the routing and forwarding tables are different for the IPv4 and IPv6 protocols. This configuration is called a multiprotocol VRF.

## Single-Protocol VRF to Multiprotocol VRF Migration

Prior to Cisco IOS Release 12.2(33)SRB, you could create a single protocol IPv4-only VRF. You created a single-protocol VRF by entering the **ip vrf** command. To activate the single-protocol VRF on an interface, you entered the **ip vrf forwarding** (interface configuration) command.

After the introduction of the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature in Cisco IOS Release 12.2(33)SRB, you create a multiprotocol VRF by entering the **vrf definition** command. To activate the multiprotocol VRF on an interface, you enter the **vrf forwarding** command.

The MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature introduces the **vrf upgrade-cli multi-af-mode {common-policies | non-common-policies} [vrf vrf-name]** command that forces VRF configuration migration from a single-protocol VRF model to a multiprotocol VRF model:

- If the route-target policies apply to all address families configured in the multi-AF VRF, select the **common-policies** keyword.
- If the route-target policies apply only to the IPv4 address family that you are migrating, select the **non-common-policies** keyword.

After you enter the **vrf upgrade-cli** command and save the configuration to NVRAM, the single-protocol VRF configuration is saved as a multiprotocol VRF configuration. In the upgrade process, the **ip vrf** command is converted to the **vrf definition** command (global configuration commands) and the **ip vrf forwarding** command is converted to the **vrf forwarding** command (interface configuration command). The **vrf upgrade-cli** command has a one-time immediate effect.

You might have both IPv4-only VRFs and multiprotocol VRFs on your router. Once you create a VRF, you can edit it using only the commands in the mode in which it was created. For example, you created a VRF named vrf2 with the following multiprotocol VRF commands:

```
Router# configure terminal
Enter configuration command, one per line. End with CNTL/Z
Router(config)# vrf definition vrf2
Router(config-vrf)# rd 2:2
Router(config-vrf)# route-target import 2:2
Router(config-vrf)# route-target export 2:2
Router(config-vrf)# end
```

If you try to edit VRF vrf2 with IPv4-only VRF commands, you receive the following message:

```
Router# configure terminal
Enter configuration command, one per line. End with CNTL/Z
Router(config)# ip vrf vrf2
```

% Use 'vrf definition vrf2' command

If you try to edit an IPv4-only VRF with the multiprotocol VRF commands, you would receive this message, where <vrf-name> is the name of the IPv4-only VRF:

% Use 'ip vrf <vrf-name>' command

The **ip vrf name** and **ip vrf forwarding** (interface configuration) *name* commands will be available for a period of time before they are removed. Use the **vrf upgrade** command to migrate your older IPv4-only VRFs to the new multiprotocol VRF configuration. When you need to create a new VRF—whether the VRF is for an IPv4 VPN, or IPv6 VPN, or both—use the multiprotocol VRF **vrf definition** and **vrf forwarding** commands that support a multi-AF configuration.

## Multiprotocol VRF Configurations Characteristics and Examples

In a multiprotocol VRF, you can configure both IPv4 VRFs and IPv6 VRFs under the same address family or configure separate VRFs for each IPv4 or IPv6 address family. The multiprotocol VRF configuration has the following characteristics:

- The VRF name identifies a VRF, which might have both IPv4 and IPv6 address families. On the same interface, you cannot have IPv4 and IPv6 address families using different VRF names.
- The RD, VPN ID, and SNMP context are shared by both IPv4 and IPv6 address families for a given VRF.
- The policies (route target, for example) specified in multi-AF VRF mode, outside the address-family configuration, are defaults to be applied to each address family. Route targets are the only VRF characteristics that can be defined inside and outside an address family.

The following is also true when you associate a multiprotocol VRF with an interface:

- Binding an interface to a VRF (**vrf forwarding vrf-name** command) removes all IPv4 and IPv6 addresses configured on that interface.
- Once you associate a VRF with a given interface, all active address families belong to that VRF. The exception is when no address of the address-family type is configured, in which case the protocol is disabled.
- Configuring an address on an interface that is bound to a VRF requires that the address family corresponding to the address type is active for that VRF. Otherwise, an error message is issued stating that the address family must be activated first in the VRF.

Backward compatibility with the single-protocol VRF CLI is supported in Cisco IOS Release 12.2(33)SRB. This means that you might have single-protocol and multiprotocol CLI on the same router, but not in the same VRF configuration.

The single-protocol CLI continues to allow you to define an IPv4 address within a VRF and an IPv6 address in the global routing table on the same interface.

The following sections have multiprotocol VRF configuration examples:

- [Multiprotocol VRF Configuration: Single Protocol with Noncommon Policies Example, page 4](#)
- [Multiprotocol VRF Configuration: Multiprotocol with Noncommon Policies Example, page 5](#)
- [Multiprotocol VRF Configuration: Multiprotocol with Common Policies Example, page 5](#)
- [Multiprotocol VRF Configuration: Multiprotocol with Common and Noncommon Policies, page 5](#)

### Multiprotocol VRF Configuration: Single Protocol with Noncommon Policies Example

The following is an example of a multiprotocol VRF configuration for a single protocol (IPv4) with route-target policies in the address-family configuration:

```
vrf definition vrf2
rd 2:2
!
address-family ipv4
```

```
route-target export 2:2
route-target import 2:2
exit-address-family
```

The RD (2:2) applies to all address families defined for VRF vrf2.

## Multiprotocol VRF Configuration: Multiprotocol with Noncommon Policies Example

The following is an example of a multiprotocol VRF configuration for IPv4 and IPv6 VPNs in which the route-target policies are defined in the separate address-family configurations:

```
vrf definition vrf2
 rd 2:2
 !
 address-family ipv4
  route-target export 2:2
  route-target import 2:2
 exit-address-family
 !
 address-family ipv6
  route-target export 3:3
  route-target import 3:3
 exit-address-family
```

## Multiprotocol VRF Configuration: Multiprotocol with Common Policies Example

The following is an example of a multiprotocol VRF configuration for IPv4 and IPv6 VPNs with route-target policies defined in the global part of the VRF:

```
vrf definition vrf2
 rd 2:2
 route-target export 2:2
 route-target import 2:2
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
```

The route-target policies are defined outside the address-family configurations. Therefore, the policies apply to all address families defined in VRF vrf2.

## Multiprotocol VRF Configuration: Multiprotocol with Common and Noncommon Policies

The following is an example of a multiprotocol VRF with route-target policies defined in both global and address-family areas:

- For IPv6, the route-target definitions are defined under the address family. These definitions are used and the route-target definitions in the global area are ignored. Therefore, the IPv6 VPN ignores import 100:2.
- For IPv4, no route-target policies are defined under the address family, therefore, the global definitions are used.

```
vrf definition vrf1
 route-target export 100:1
 route-target import 100:1
 route-target import 100:2
 !
```

```

address-family ipv4
exit-address-family
!
address-family ipv6
route-target export 100:1
route-target import 100:1
route-target import 100:3
exit-address-family

```

## How to Configure MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

This feature provides Cisco IOS CLI commands that allow you to configure a multiprotocol VRF (IPv4 and IPv6 VPNs in the same VRF) and to migrate a single-protocol VRF configuration (IPv4-only VRF) to a multiprotocol VRF configuration.

A multiprotocol VRF allows you to share route targets policies (import and export) between IPv4 and IPv6 or to configure separate route-target policies for IPv4 and IPv6 VPNs.

Perform the following tasks to configure or migrate to the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature:

- [Configuring a VRF for IPv4 and IPv6 VPNs, page 6](#) (required)
- [Associating a Multiprotocol VRF with an Interface, page 8](#) (required)
- [Verifying the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs Configuration, page 10](#) (optional)

Perform the following task to migrate from a single-protocol VRF to a multiprotocol VRF configuration:

- [Migrating from a Single-Protocol IPv4-Only VRF to a Multiprotocol VRF Configuration, page 13](#)

## Configuring a VRF for IPv4 and IPv6 VPNs

Perform the following task to configure a VRF for IPv4 and IPv6 VPNs. When you configure a VRF for both IPv4 and IPv6 VPNs (a multiprotocol VRF), you can choose to configure route-target policies that apply to all address-families in the VRF or you can configure route-target policies that apply to individual address families in the VRF.

The following task shows how to configure a VRF that has that has route-target policies defined for IPv4 and IPv6 VPNs in separate VRF address families.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **rd** *route-distinguisher*
5. **address-family** {**ipv4** | **ipv6**}
6. **route-target** {**import** | **export** | **both**} *route-target-ext-community*
7. **exit-address-family**
8. **address-family** {**ipv4** | **ipv6**}
9. **route-target** {**import** | **export** | **both**} *route-target-ext-community*

10. end

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b>  <b>Example:</b> Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b>  <b>Example:</b> Router# configure terminal	Enters global configuration mode.
Step 3	<b>vrf definition vrf-name</b>  <b>Example:</b> Router(config)# vrf definition vrf1	Configures a VRF routing table and enters VRF configuration mode. <ul style="list-style-type: none"> <li>The <b>vrf-name</b> argument is the name of the VRF.</li> </ul>
Step 4	<b>rd route-distinguisher</b>  <b>Example:</b> Router(config-vrf)# rd 100:1	Creates routing and forwarding tables for a VRF. <ul style="list-style-type: none"> <li>The <i>route-distinguisher</i> argument specifies to add an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix. You can enter a route distinguisher in either of these formats: <ul style="list-style-type: none"> <li>16-bit ASN (autonomous system number): your 32-bit number For example, 101:3.</li> <li>32-bit IP address: your 16-bit number For example, 192.168.122.15:1.</li> </ul> </li> </ul>
Step 5	<b>address-family {ipv4   ipv6}</b>  <b>Example:</b> Router(config-vrf) address-family ipv4	Enters VRF address family configuration mode. <ul style="list-style-type: none"> <li>The <b>ipv4</b> keyword specifies an address family for an IPv4 VPN.</li> <li>The <b>ipv6</b> keyword specifies an address family for an IPv6 VPN.</li> </ul>
Step 6	<b>route-target {import   export   both}</b> <b>route-target-ext-community</b>  <b>Example:</b> Router(config-vrf-af)# route-target both 100:2	Creates a route-target extended community for a VRF. <ul style="list-style-type: none"> <li>The <b>import</b> keyword specifies to import routing information from the target VPN extended community.</li> <li>The <b>export</b> keyword specifies to export routing information to the target VPN extended community.</li> <li>The <b>both</b> keyword specifies to import both import and export routing information to the target VPN extended community.</li> <li>The <i>route-target-ext-community</i> argument adds the route-target extended community attributes to the VRF's list of import, export, or both (import and export) route-target extended communities.</li> </ul>

	Command or Action	Purpose
Step 7	<b>exit-address-family</b>  <b>Example:</b> Router(config-vrf-af)# exit-address-family	Exits from VRF address family configuration mode.
Step 8	<b>address-family {ipv4   ipv6}</b>  <b>Example:</b> Router(config-vrf) address-family ipv6	Enters VRF address family configuration mode. <ul style="list-style-type: none"> <li>The <b>ipv4</b> keyword specifies an address family for an IPv4 VPN.</li> <li>The <b>ipv6</b> keyword specifies an address family for an IPv6 VPN.</li> </ul>
Step 9	<b>route-target {import   export   both}</b> <i>route-target-ext-community</i>  <b>Example:</b> Router(config-vrf-af)# route-target both 100:3	Creates a route-target extended community for a VRF. <ul style="list-style-type: none"> <li>The <b>import</b> keyword specifies to import routing information from the target VPN extended community.</li> <li>The <b>export</b> keyword specifies to export routing information to the target VPN extended community.</li> <li>The <b>both</b> keyword specifies to import both import and export routing information to the target VPN extended community.</li> <li>The <i>route-target-ext-community</i> argument adds the route-target extended community attributes to the VRF's list of import, export, or both (import and export) route-target extended communities.</li> </ul>
Step 10	<b>end</b>  <b>Example:</b> Router(config-vrf-af)# end	Exits to privileged EXEC mode.

## Associating a Multiprotocol VRF with an Interface

Perform the following task to associate a multiprotocol VRF with an interface. Associating the VRF with an interface activates the VRF.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **vrf forwarding** *vrf-name*
5. **ip address** *ip-address mask* [**secondary**]
6. **ipv6 address** {*ipv6-address/prefix-length* | *prefix-name sub-bits/prefix-length*}
7. **end**



## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>enable</b>  <b>Example:</b> Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul>
Step 2	<b>configure terminal</b>  <b>Example:</b> Router# configure terminal	Enters global configuration mode.
Step 3	<b>interface</b> <i>type number</i>  <b>Example:</b> Router(config)# interface Ethernet 0/1	Configures an interface type and enters interface configuration mode. <ul style="list-style-type: none"> <li>The <b>type</b> argument identifies the type of interface to be configured.</li> <li>The <b>number</b> argument identifies the port, connector, or interface card number.</li> </ul>
Step 4	<b>vrf forwarding</b> <i>vrf-name</i>  <b>Example:</b> Router(config-if)# vrf forwarding vrf1	Associates a VRF with an interface or subinterface. <ul style="list-style-type: none"> <li>The <i>vrf-name</i> argument is the name of the VRF.</li> </ul>
Step 5	<b>ip address</b> <i>ip-address mask</i> [ <b>secondary</b> ]  <b>Example:</b> Router(config-if)# ip address 10.24.24.24 255.255.255.255	Sets a primary or secondary IP address for an interface. <ul style="list-style-type: none"> <li>The <i>ip-address</i> argument is the IP address.</li> <li>The <i>mask</i> argument is the mask of the associated IP subnet.</li> <li>The <b>secondary</b> keyword specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.</li> </ul>

Command or Action	Purpose
<p><b>Step 6</b></p> <p><b>ipv6 address</b> {<i>ipv6-address/prefix-length</i>   <i>prefix-name sub-bits/prefix-length</i>}</p> <p><b>Example:</b>  Router(config-if)# ipv6 address  2001:0DB8:0300:0201::/64</p>	<p>Configures an IPv6 address based on an IPv6 general prefix and enables IPv6 processing on an interface.</p> <ul style="list-style-type: none"> <li>The <i>ipv6-address</i> argument is the IPv6 address to be used.</li> <li>The <i>prefix-length</i> argument is the length of the IPv6 prefix, which is a decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value.</li> <li>The <i>prefix-name</i> argument is a general prefix that specifies the leading bits of the network to be configured on the interface.</li> <li>The <i>sub-bits</i> argument is the subprefix bits and host bits of the address to be concatenated with the prefixes provided by the general prefix specified with the <i>prefix-name</i> argument.</li> </ul> <p>The <i>sub-bits</i> argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.</p>
<p><b>Step 7</b></p> <p><b>end</b></p> <p><b>Example:</b>  Router(config-if) end</p>	<p>Exits to privileged EXEC mode.</p>

## Verifying the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs Configuration

Perform the following task to verify the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature configuration, that is, show that the VRF configuration is upgraded to a multi-AF multiprotocol VRF.

### SUMMARY STEPS

1. **enable**
2. **show running-config vrf** [*vrf-name*]
3. **show vrf**
4. **show vrf detail** [*vrf-name*]
5. **end**

### DETAILED STEPS

#### Step 1 **enable**

Use this command to enable privileged EXEC mode. Enter your password, if prompted. For example:

```
Router> enable
Router#
```

**Step 2** **show running-config vrf** [*vrf-name*]

Use this command to verify that the upgrade to a multi-AF multiprotocol VRF configuration was successful. The following is sample command output before the upgrade to a multi-AF multiprotocol VRF:

```
Router# show running-config vrf vpn2

Building configuration...

Current configuration : 604 bytes
ip vrf vpn2
  rd 1:1
  route-target export 1:1
  route-target import 1:1
!
!
interface Loopback1
  ip vrf forwarding vpn2
  ip address 10.43.43.43 255.255.255.255
!
```

The following is sample command output after you upgrade to a multi-AF multiprotocol VRF with common policies for all address families:

```
Router# show running-config vrf vpn1

Building configuration...

Current configuration : 604 bytes
vrf definition vpn1
  rd 1:1
  route-target export 1:1
  route-target import 1:1
!
  address-family ipv4
  exit-address-family
!
!
interface Loopback1
  ip vrf forwarding vpn1
  ip address 10.43.43.43 255.255.255.255
!
```

This configuration contains the **vrf definition** command. The **vrf definition** command replaces the **ip vrf** command in the multi-AF multiprotocol VRF configuration.

**Step 3** **show vrf**

Use this command to verify that the upgrade to a multi-AF multiprotocol VRF configuration was successful. The **show vrf** command replaces the **show ip vrf** command when a VRF configuration is updated to a multi-AF multiprotocol VRF configuration. The **show vrf** command displays the protocols defined for a VRFs. The following command shows sample output after you upgrade a single-protocol VRF configuration to a multi-AF multiprotocol VRF configuration:

```
Router# show vrf vpn1
```

Name	Default RD	Protocols	Interfaces
vpn1	1:1	ipv4	Lo1/0

The following is sample output from the **show ip vrf vp1** command. Compare this to the output of the **show vrf vp1** command. The protocols under the VRF are not displayed.

```
Router# show ip vrf vrf1
```

Name	Default RD	Interface
vp1	1:1	Loopback1

The following is sample output from the **show vrf** command for multiprotocol VRFs, one of which contains both IPv4 and IPv6 protocols:

```
Router# show vrf
```

Name	Default RD	Protocols	Interfaces
vp1	1:1	ipv4	Lo1/0
vp2	100:3	ipv4	Lo23 AT3/0/0.1
vp4	100:2	ipv4, ipv6	

#### Step 4 **show vrf detail [vrf-name]**

Use this command to display all characteristics of the defined VRF to verify that the configuration is as you expected. For example, if your VRF configuration for VRF vp1 is as follows:

```
vrf definition vp1
 route-target export 100:1
 route-target import 100:1
 route-target import 100:2
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 route-target export 100:1
 route-target import 100:1
 route-target import 100:3
 exit-address-family
```

This command would display the following:

```
Router# show vrf detail vp1
```

```
VRF vp1 (VRF Id = 3); default RD <not set>; default VPNID <not set>
  No interfaces
Address family ipv4 (Table ID = 3 (0x3)):
  Connected addresses are not in global routing table
  Export VPN route-target communities
    RT:100:1

  Import VPN route-target communities
    RT:100:1          RT:100:2

  No import route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
Address family ipv6 (Table ID = 503316483 (0x1E000003)):
  Connected addresses are not in global routing table
  Export VPN route-target communities
    RT:100:1

  Import VPN route-target communities
    RT:100:1          RT:100:3

  No import route-map
  No export route-map
```

```
VRF label distribution protocol: not configured
VRF label allocation mode: per-prefix
```

**Step 5** **exit**

Use this command to exit to user EXEC mode. For example:

```
Router# exit
Router>
```

## Migrating from a Single-Protocol IPv4-Only VRF to a Multiprotocol VRF Configuration

Perform the following task to force migration from a single-protocol IPv4-only VRF to a multiprotocol VRF configuration.

The multiprotocol VRF configuration allows you to define multiple address families under the same VRF. A given VRF, identified by its name and a set of policies, can apply to both an IPv4 VPN and an IPv6 VPN at the same time. This VRF can be activated on a given interface, even though the routing and forwarding tables are different for the IPv4 and IPv6 protocols.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vrf upgrade-cli multi-af-mode { common-policies | non-common-policies } [vrf vrf-name]**
4. **exit**
5. **show running-config vrf vrf-name**

### DETAILED STEPS

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

	Command or Action	Purpose
Step 3	<p><b>vrf upgrade-cli multi-af-mode</b> {<b>common-policies</b>   <b>non-common-policies</b>} [<b>vrf</b> <i>vrf-name</i>]</p> <p><b>Example:</b> Router(config)# vrf upgrade-cli multi-af-mode common-policies vrf vpn4</p>	<p>Upgrades a VRF instance or all VRFs configured on the router to support multiple address families under the same VRF.</p> <ul style="list-style-type: none"> <li>The <b>multi-af-mode</b> keyword specifies an upgrade of a single-protocol VRF or all VRFs to a multiprotocol VRF that supports multi-AFs configuration.</li> <li>The <b>common-policies</b> keyword specifies to copy the route-target policies to the common part of the VRF configuration so that the policies apply to all address families configured in the multi-AF VRF.</li> <li>The <b>non-common-policies</b> keyword specifies to copy the route-target policies to the IPv4 address family part of the VRF configuration so that the policies apply only to IPv4.</li> <li>The <b>vrf</b> keyword specifies a VRF for the upgrade to a multi-AF VRF configuration.</li> <li>The <i>vrf-name</i> argument is the name of the single-protocol VRF to upgrade to a multi-AF VRF configuration.</li> </ul>
Step 4	<p><b>exit</b></p> <p><b>Example:</b> Router(config)# exit</p>	Exits to privileged EXEC mode.
Step 5	<p><b>show running-config vrf</b> [<i>vrf-name</i>]</p> <p><b>Example:</b> Router# show running-config vrf vpn4</p>	<p>Displays the subset of the running configuration of a router that is linked to a specific VRF instance or to all VRFs configured on the router.</p> <ul style="list-style-type: none"> <li>The <i>vrf-name</i> argument is the name of the VRF of which you want to display the configuration.</li> </ul> <p><b>Note</b> The Cisco IOS image that supports the multiprotocol VRF commands might not support the <b>show running-config vrf</b> command. You can use the <b>show running-config</b> command instead.</p>

## Configuration Examples for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

The following examples show how to use the VRF CLI provided by the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature to migrate from a single-protocol VRF to a multiprotocol VRF configuration:

- [Configuring a VRF for IPv4 and IPv6 VPNs: Example, page 15](#)
- [Associating a Multiprotocol VRF with an Interface: Example, page 15](#)
- [Migrating from a Single-Protocol IPv4-Only VRF Configuration to a Multiprotocol VRF Configuration: Example, page 16](#)

## Configuring a VRF for IPv4 and IPv6 VPNs: Example

The following example shows how to configure a VRF for IPv4 and IPv6 VPNs:

```
configure terminal
!
vrf definition vrf1
  rd 100:1
!
  address-family ipv4
    route-target both 100:2
  exit-address-family
!
  address-family ipv6
    route-target import 100:3
    route-target export 100:3
  exit-address-family
end
```

In this example, noncommon policies are defined in the address-family configuration.

The following is an example of a VRF for IPv4 and IPv6 that has common policies defined in the global part of the VRF configuration:

```
configure terminal
!
vrf definition vrf2
  rd 200:1
  route-target import 200:2
  route-target export 200:2
!
  address-family ipv4
  exit-address-family
!
  address-family ipv6
  exit-address-family
end
```

## Associating a Multiprotocol VRF with an Interface: Example

The following example shows how to associate a multiprotocol VRF with an interface:

```
configure terminal
!
interface Ethernet 0/1
  vrf forwarding vrf1
  ip address 10.24.24.24 255.255.255.255
  ipv6 address 2001:0DB8:0300:0201::/64
end
```

## Migrating from a Single-Protocol IPv4-Only VRF Configuration to a Multiprotocol VRF Configuration: Example

This section contains examples that show how to migrate from a single-protocol IPv4-only VRF to a multiprotocol VRF configuration.

This example shows a single-protocol IPv4-only VRF before the Cisco IOS VRF CLI for IPv4 and IPv6 is entered on the router:

```
ip vrf vrf1
 rd 1:1
 route-target export 1:1
 route-target import 1:1

interface Loopback1
 ip vrf forwarding V1
 ip address 10.3.3.3 255.255.255.255
```

This example shows how to force the migration of the single-protocol VRF vrf1 to a multiprotocol VRF configuration:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
!
Router(config)# vrf upgrade-cli multi-af-mode common-policies vrf vrf1
```

```
You are about to upgrade to the multi-AF VRF syntax commands.
You will loose any IPv6 address configured on interfaces
belonging to upgraded VRFs.
Are you sure ? [yes]: yes
```

```
Number of VRFs upgraded: 1
```

```
Router(config)# exit
```

This example shows the multiprotocol VRF configuration after the forced migration:

```
vrf definition vrf1
 rd 1:1
 route-target export 1:1
 route-target import 1:1
 !
 address-family ipv4
 exit-address-family
 !

interface Loopback1
 vrf forwarding V1
 ip address 10.3.3.3 255.255.255.255
```

The following is another example of a multi-AF multiprotocol VRF configuration:

```
vrf definition vrf2
 rd 100:1
 address family ipv6
 route-target 200:1 both
 exit-address-family
 !
ip vrf vrf1
 rd 200:1
 route-target 200:1 both
 !
```



```

interface Ethernet0/0
 vrf forwarding vrf2
 ip address 10.50.1.2 255.255.255.0
 ipv6 address 2001:0DB8:0:1::/64
!
interface Ethernet0/1
 ip vrf forwarding vrf1
 ip address 10.60.1.2 255.255.255.0
 ipv6 address 2001:0DB8:1:1::/64

```

In this example, all addresses (IPv4 and IPv6) defined for interface Ethernet0/0 are in VRF vrf2. For the interface Ethernet0/1, the IPv4 address is defined in VRF vrf1 but the IPv6 address is in the global IPv6 routing table.

## Additional References

The following sections provide references related to the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature.

## Related Documents

Related Topic	Document Title
Configuration tasks for MPLS VPNs	<a href="#">“Part 4: MPLS Virtual Private Networks,” <i>Cisco IOS Multiprotocol Label Switching Configuration Guide</i>, Release 4</a>
Commands for configuring MPLS and MPLS VPNs	<a href="#">Cisco IOS Multiprotocol Label Switching Command Reference</a> , Release 12.2SR
Configuration tasks for MPLS	“Configuring Multiprotocol Label Switching” chapter in the <a href="#">Cisco IOS Multiprotocol Label Switching Configuration Guide</a> , Release 4

## Standards

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

## MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a>

## RFCs

RFC	Title
RFC 1771	<i>A Border Gateway Protocol 4 (BGP-4)</i>
RFC 4364	<i>BGP MPLS/IP Virtual private Networks (VPNs)</i>

## Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register on Cisco.com.	<a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a>

## Command Reference

This section documents only commands that are new or modified.

- [show vrf](#)
- [vrf definition](#)
- [vrf forwarding](#)
- [vrf upgrade-cli](#)

# show vrf

To display the defined Virtual Private Network (VPN) routing and forwarding (VRF) instances, use the **show vrf** command in user EXEC or privileged EXEC mode.

**show vrf** [**ipv4** | **ipv6**] [**interface** | **brief** | **detail** | **id** | **select**] [*vrf-name*]

Syntax Description	<b>ipv4</b>	(Optional) Displays IPv4 address-family type VRF instances.
	<b>ipv6</b>	(Optional) Displays IPv6 address-family type VRF instances.
	<b>interface</b>	(Optional) Displays the interface associated with the specified VRF instances.
	<b>brief</b>	(Optional) Displays brief information about the specified VRF instances.
	<b>detail</b>	(Optional) Displays detailed information about the specified VRF instances.
	<b>id</b>	(Optional) Displays VPN-ID information for the specified VRF instances.
	<b>select</b>	(Optional) Displays selection information for the specified VRF instances.
	<i>vrf-name</i>	(Optional) Name assigned to a VRF.

**Command Default** If you do not specify any arguments or keywords, the command displays concise information about all configured VRFs.

**Command Modes** User EXEC  
Privileged EXEC

Command History	Release	Modification
	12.2(33)SRB	This command was introduced.

**Usage Guidelines** Use the **show vrf** command to display information about specified VRF instances or all VRF instances. Specify no arguments or keywords to display information on all VRF instances.

**Examples** The following is sample output from the **show vrf** command that displays brief information about all configured VRF instances:

Router# **show vrf**

Name	Default RD	Protocols	Interfaces
N1	100:0	ipv4, ipv6	
V1	1:1	ipv4	Lo1

V2	2:2	ipv4, ipv6	Et0/1.1 Et0/1.2 Et0/1.3
V3	3:3	ipv4	Lo3 Et0/1.4

Table 1 describes the significant fields shown in the display.

**Table 1** *show vrf Field Descriptions*

Field	Description
Name	Name of the VRF instance.
Default RD	The default route distinguisher for the specified VRF instances.
Protocols	The address-family protocol type for the specified VRF instance.
Interfaces	The network interface associated with the VRF instance.

The following example displays output from the **show vrf** command with the **detail** keyword. The information shown is for a VRF named cisco1.

Router# **show vrf detail**

```

VRF cisco1; default RD 100:1; default VPNID <not set>
  Interfaces:
    Ethernet0/0          Loopback10
Address family ipv4 (Table ID = 0x1):
  Connected addresses are not in global routing table
  Export VPN route-target communities
    RT:100:1
  Import VPN route-target communities
    RT:100:1
  No import route-map
  No export route-map
  VRF label distribution protocol: not configured
Address family ipv6 (Table ID = 0xE000001):
  Connected addresses are not in global routing table
  Export VPN route-target communities
    RT:100:1
  Import VPN route-target communities
    RT:100:1
  No import route-map
  No export route-map
  VRF label distribution protocol: not configured

```

Table 2 describes the significant fields shown in the display.

**Table 2** *show vrf detail Field Descriptions*

Field	Description
default RD 100:1	The route distinguisher (RD) given to this VRF.
Interfaces:	Interfaces to which the VRF is attached.

**Table 2** *show vrf detail Field Descriptions (continued)*

Field	Description
Export VPN route-target communities RT:100:1	Route target VPN extended communities to be exported.
Import VPN route-target communities RT:100:1	Route target VPN extended communities to be imported.

**Related Commands**

Command	Description
<b>vrf definition</b>	Configures a VRF routing table instance and enters VRF configuration mode.
<b>vrf forwarding</b>	Associates a VRF instance with an interface or subinterface.

## vrf definition

To configure a Virtual Private Network (VPN) routing and forwarding (VRF) routing table instance and enter VRF configuration mode, use the **vrf definition** command in global configuration mode. To remove a VRF routing table, use the **no** form of this command.

**vrf definition** *vrf-name*

**no vrf definition** *vrf-name*

<b>Syntax Description</b>	<i>vrf-name</i>	Name assigned to a VRF.
---------------------------	-----------------	-------------------------

<b>Command Default</b>	No VRFs are defined. No import or export lists are associated with a VRF. No route maps are associated with a VRF.
------------------------	--

<b>Command Modes</b>	Global configuration
----------------------	----------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.2(33)SRB	This command was introduced.

<b>Usage Guidelines</b>	Use the <b>vrf definition</b> command to give a VRF a name and to enter VRF configuration mode. Once the router is in VRF configuration mode, use the <b>rd</b> command to give the VRF a route distinguisher. The <b>rd</b> command creates the routing and forwarding tables and associates the route distinguisher (RD) with the VRF instance named in the <i>vrf-name</i> argument.
-------------------------	---

Users can configure shared route targets (import and export) between IPv4 and IPv6. This feature is useful in a migration scenario, where IPv4 policies already are configured and IPv6 policies should be the same as the IPv4 policies. You can configure separate route-target policies for IPv4 and IPv6 VPNs in address family configuration mode. Enter address family configuration mode from VRF configuration mode.

In VRF configuration mode, you can also associate a Simple Network Management Protocol (SNMP) context with the named VRF and configure or update a VPN ID.

<b>Examples</b>	The following example assigns the name vrf1 to a VRF, enters VRF configuration mode, and configures a route distinguisher, 100:20:
-----------------	--

```
Router(config)# vrf definition vrf1
Router(config-vrf)# rd 100:20
```

**Related Commands**

Command	Description
<b>address-family</b>	Enters address family configuration mode for configuring routing protocols such as BGP, RIP, and static routing.
<b>context</b>	Associates an SNMP context with a particular VRF.
<b>rd</b>	Specifies a route distinguisher.
<b>route-target</b>	Creates a route-target extended community for a VPN VRF.
<b>vpn id</b>	Sets or updates a VPN ID on a VRF.
<b>vrf forwarding</b>	Associates a VRF instance with an interface or subinterface.

# vrf forwarding

To associate a Virtual Private Network (VPN) routing and forwarding (VRF) instance with an interface or subinterface, use the **vrf forwarding** command in interface configuration mode. To disassociate a VRF from an interface, use the **no** form of this command.

**vrf forwarding** *vrf-name*

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

Syntax Description	<i>vrf-name</i> Name assigned to a VRF.					
Command Default	The default for an interface is the global routing table.					
Command Modes	Interface configuration					
Command History	<table><tr><th>Release</th><th>Modification</th></tr><tr><td>12.2(33)SRB</td><td>This command was introduced.</td></tr></table>		Release	Modification	12.2(33)SRB	This command was introduced.
Release	Modification					
12.2(33)SRB	This command was introduced.					
Usage Guidelines	Use the <b>vrf forwarding</b> command to associate an interface with a VRF. When the interface is bound to a VRF, previously configured IPv4 and IPv6 addresses are removed, and they must be reconfigured.					
Examples	<p>The following example shows how to associate a VRF named site1 to serial interface 0/0 and configure an IPv6 and an IPv4 address:</p> <pre>interface Serial0/0  vrf forwarding site1  ipv6 address 2001:100:1:1000::72b/64  ip address 10.11.11.1 255.255.255.0</pre>					
Related Commands	<table><tr><th>Command</th><th>Description</th></tr><tr><td><b>vrf definition</b></td><td>Configures a VRF routing table instance and enters VRF configuration mode.</td></tr></table>		Command	Description	<b>vrf definition</b>	Configures a VRF routing table instance and enters VRF configuration mode.
Command	Description					
<b>vrf definition</b>	Configures a VRF routing table instance and enters VRF configuration mode.					



# vrf upgrade-cli

To upgrade a Virtual Private Network (VPN) routing and forwarding (VRF) instance or all VRFs on the router to support multiple address families (multi-AFs) for the same VRF, use the **vrf upgrade-cli** command in global configuration mode.

**vrf upgrade-cli multi-af-mode { common-policies | non-common-policies } [vrf vrf-name]**

Syntax Description	<b>multi-af-mode</b>	Specifies an upgrade of a single-protocol VRF or all VRFs to a multiprotocol VRF that supports multi-AFs configuration.
	<b>common-policies</b>	Specifies to copy the route-target policies to the common part of the VRF configuration so that the policies apply to all address families configured in the multi-AF VRF.
	<b>non-common-policies</b>	Specifies to copy the route-target policies to the IPv4 address family part of the VRF configuration so that the policies apply only to an IPv4 VRF.
	<b>vrf</b>	(Optional) Specifies a VRF for the upgrade to a multi-AF VRF configuration.
	<i>vrf-name</i>	(Optional) The name of the single-protocol VRF to upgrade to a multi-AF VRF configuration.

Command Default	If you do not enter the name of a specific single-protocol VRF, all VRFs defined on the router are upgraded to the multi-AF VRF configuration.
-----------------	--

Command Modes	Global configuration
---------------	----------------------

Command History	Release	Modification
	12.2(33)SRB	This command was introduced.

Usage Guidelines	The <b>vrf upgrade-cli</b> command is used to upgrade a specified single-protocol VRF (IPv4-only VRF) configuration or all single-protocol VRF configurations on the router to a multiprotocol VRF that supports multi-AF configuration.
	The upgrade is automatic and does not require any further configuration. After you enter the <b>vrf upgrade-cli</b> command, the single-protocol VRF configuration is lost when you save the configuration to NVRAM. A multiprotocol VRF configuration is saved.
	If your configuration requires that all route-target policies (import, export, both) apply to all address families, you enter the <b>vrf upgrade-cli multi-af-mode common-policies</b> command. If your configuration requires that these policies apply to IPv4 VPNs only, enter the <b>vrf upgrade-cli multi-af-mode non-common-policies</b> command.
	After the upgrade to a multiprotocol VRF is complete, you can edit the VRF only with multiprotocol VRF configuration commands.

## Examples

The following example shows how to upgrade a single-protocol VRF configuration named `vrf1` to a multi-AF VRF configuration and apply the common policies of `vrf1` to all address families defined for the VRF:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
!
Router(config)# vrf upgrade-cli multi-af-mode common-policies vrf vrf1

You are about to upgrade to the multi-AF VRF syntax commands.
You will loose any IPv6 address configured on interfaces
belonging to upgraded VRFs.
Are you sure ? [yes]: yes
Number of VRFs upgraded: 1
Router(config)# exit
```

The following is an example of the single-protocol VRF configuration for VRF `vrf1` before you enter the **vrf upgrade-cli** command to upgrade to a multi-AF multiprotocol VRF configuration:

```
!
ip vrf vrf1
 rd 1:1
  route-target export 1:1
  route-target import 1:1

interface Loopback1
 ip vrf forwarding vrf1
 ip address 10.3.3.3 255.255.255.255
```

This is an example of the multi-AF multiprotocol VRF configuration for VRF `vrf1` after you enter the **vrf upgrade-cli** command with the **common-policies** keyword:

```
!
vrf definition vrf1
 rd 1:1
  route-target export 1:1
  route-target import 1:1
!
 address-family ipv4
  exit-address-family
!

interface Loopback1
 vrf forwarding vrf1
 ip address 10.3.3.3 255.255.255.255
```

## Related Commands

Command	Description
<b>show vrf</b>	Displays the defined VRF instances.
<b>vrf definition</b>	Configures a VRF routing table instance and enters VRF configuration mode.
<b>vrf forwarding</b>	Associates a VRF instance with an interface or subinterface.

# Feature Information for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs

Table 3 lists the release history for this feature.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

**Note**

Table 3 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

**Table 3**      *Feature Information for MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs*

Feature Name	Releases	Feature Information
MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs	12.2(33)SRB	<p>This document describes how to configure a multiprotocol Virtual Private Network (VPN) routing and forwarding (VRF) instance for IPv4 and IPv6 VPNs and describes how to upgrade your existing single-protocol IPv4-only VRF to a multiprotocol VRF configuration.</p> <p>The MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs feature introduces Cisco IOS command-line interface (CLI) commands that allow you to enable an IPv4 and IPv6 VPN in the same Multiprotocol Label Switching (MPLS) VRF instance and to simplify the migration from a single-protocol VRF configuration to a multiprotocol VRF configuration.</p> <p>In 12.2(33)SRB, this feature was introduced on the Cisco 7600 router.</p> <p>The following sections provide information about this feature:</p> <ul style="list-style-type: none"> <li>• <a href="#">VRF Concepts Similar for IPv4 and IPv6 VPNs, page 2</a></li> <li>• <a href="#">Single-Protocol VRF to Multiprotocol VRF Migration, page 3</a></li> <li>• <a href="#">Multiprotocol VRF Configurations Characteristics and Examples, page 4</a></li> <li>• <a href="#">Configuring a VRF for IPv4 and IPv6 VPNs, page 6</a></li> <li>• <a href="#">Associating a Multiprotocol VRF with an Interface, page 8</a></li> <li>• <a href="#">Verifying the MPLS VPN—VRF CLI for IPv4 and IPv6 VPNs Configuration, page 10</a></li> <li>• <a href="#">Migrating from a Single-Protocol IPv4-Only VRF to a Multiprotocol VRF Configuration, page 13</a></li> </ul> <p>The following commands were introduced or modified by this feature: <b>vrf definition</b>, <b>vrf forwarding</b>, <b>vrf upgrade-cli</b>, and <b>show vrf</b>.</p>

# Glossary

**6PE**—IPv6 provider edge router or a Multiprotocol Label Switching (MPLS) label switch router (LSR) edge router using IPv6.

**6VPE**—IPv6 Virtual Private Network (VPN) provider edge router.

**AF**—address family. Set of related communication protocols in which all members use a common addressing mechanism to identify endpoints. Also called protocol family.

**AFI**—Address Family Identifier. Carries the identity of the network-layer protocol that is associated with the network address.

**BGP**—Border Gateway Protocol. A routing protocol used between autonomous systems. It is the routing protocol that makes the internet work. BGP is a distance-vector routing protocol that carries connectivity information and an additional set of BGP attributes. These attributes allow for a rich set of policies for deciding the best route to use to reach a given destination. BGP is defined by RFC 1771.

**CE**—customer edge router. A service provider router that connects to Virtual Private Network (VPN) customer sites.

**FIB**—Forwarding Information Base. Database that stores information about switching of data packets. A FIB is based on information in the Routing Information Base (RIB). It is the optimal set of selected routes that are installed in the line cards for forwarding.

**HA**—high availability. High availability is defined as the continuous operation of systems. For a system to be available, all components—including application and database servers, storage devices, and the end-to-end network—need to provide continuous service.

**IP**—Internet Protocol. Network-layer protocol in the TCP/IP stack offering a connectionless internetwork service. IP provides features for addressing, type-of-service specification, fragmentation and reassembly, and security.

**IPv4**—IP Version 4. Network layer for the TCP/IP protocol suite. IPv4 is a connectionless, best-effort packet switching protocol.

**IPv6**—IP Version 6. Replacement for IPv4. IPv6 is a next-generation IP protocol. IPv6 is backward compatible with and designed to fix the shortcomings of IPv4, such as data security and maximum number of user addresses. IPv6 increases the address space from 32 to 128 bits, providing for an unlimited number of networks and systems. It also supports quality of service (QoS) parameters for real-time audio and video.

**MFI**—MPLS Forwarding Infrastructure. In the Cisco MPLS subsystem, the data structure for storing information about incoming and outgoing labels and associated equivalent packets suitable for labeling.

**MPLS**—Multiprotocol Label Switching. MPLS is a method for forwarding packets (frames) through a network. It enables routers at the edge of a network to apply labels to packets (frames). ATM switches or existing routers in the network core can switch packets according to the labels with minimal lookup overhead.

**PE**—provider edge router. A router that is part of a service provider's network and that is connected to a customer edge (CE) router. The PE router function is a combination of an MLS edge label switch router (LSR) function with some additional functions to support Virtual Private Networks (VPNs).

**RD (IPv4)**—route distinguisher. An 8-byte value that is concatenated with an IPv4 prefix to create a unique VPN IPv4 (VPNv4) prefix.

**RD (IPv6)**—route distinguisher. A 64-bit value that is prepended to an IPv6 prefix to create a globally unique VPN-IPv6 address.

**RIB**—Routing Information Base. The set of all available routes from which to choose the Forwarding Information Base (FIB). The RIB essentially contains all routes available for selection. It is the sum of all routes learned by dynamic routing protocols, all directly attached networks (that is, networks to which a given router has interfaces connected), and any additional configured routes, such as static routes.

**RT**—route target. Extended community attribute used to identify the Virtual Private Network (VPN) routing and forwarding (VRF) routing table into which a prefix is to be imported.

**VPN**—Virtual Private Network. Enables IP traffic to travel securely over a public TCP/IP network by encrypting all traffic from one network to another. A VPN uses “tunneling” to encrypt all information at the IP level.

**VRF**—Virtual Private Network (VPN) routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a PE router.

**VRF table**—A routing and a forwarding table associated to a Virtual Private Network (VPN) routing and forwarding (VRF) instance. This is a customer-specific table, enabling the provider edge (PE) router to maintain independent routing states for each customer.

**Note**

See [Internetworking Terms and Acronyms](#) for terms not included in this glossary.

CCVP, the Cisco logo, and Welcome to the Human Network are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, LightStream, Linksys, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PIX, ProConnect, ScriptShare, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0711R)

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

©2007 Cisco Systems, Inc. All rights reserved.