VRF Protocol Configuration Guide, Cisco Catalyst IE9300 Rugged Series Switches

First Published: 2025-09-09

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

Preface

This preface describes the conventions of this document and information on how to obtain other documentation. It also provides information on what's new in Cisco product documentation.

Document Conventions

This document uses the following conventions:

Convention	Description
^ or Ctrl	Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination ^**D** or **Ctrl-D** means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
bold font	Commands and keywords and user-entered text appear in **bold** font.
Italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
`Courier` font	Terminal sessions and information the system displays appear in `courier` font.
`Bold Courier` font	**`Bold Courier`** font indicates text that the user must enter.
[x]	Elements in square brackets are optional.
...	An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.
|	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
[x | y]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Convention	Description
{x | y}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Reader Alert Conventions

This document may use the following conventions for reader alerts:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Tip Means *the following information will help you solve a problem.*

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time.* You can save time by performing the action described in the paragraph.

Take note of the following general safety warnings:

Warning IMPORTANT SAFETY INSTRUCTIONS

Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Read the installation instructions before using, installing, or connecting the system to the power source. Use the statement number at the beginning of each warning statement to locate its translation in the translated safety warnings for this device.

SAVE THESE INSTRUCTIONS

Related Documentation

Note Before installing or upgrading the , refer to the release notes.

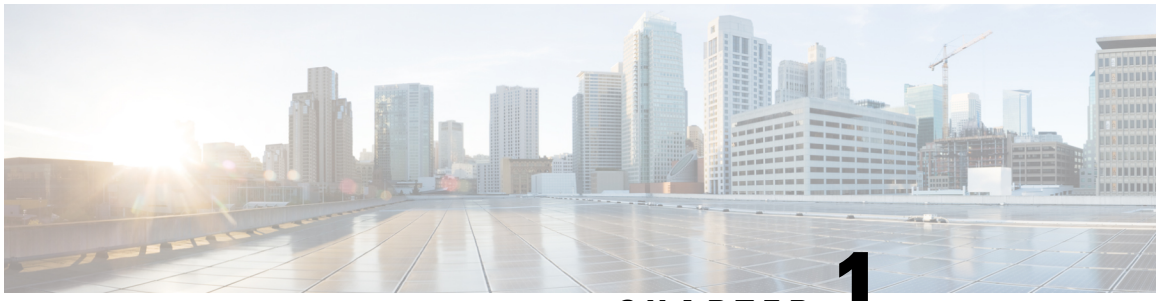
- Cisco Validated Designs documents, located at:

  http://www.cisco.com/go/designzone

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see What's New in Cisco Product Documentation.

You can also subscribe to the *What's New in Cisco Product Documentation* RSS feed, which delivers lists and content of new and revised Cisco technical documentation directly to your desktop, using any RSS reader application. This RSS feed is a free service.



CHAPTER 1

VRF

VRF-lite

Virtual Routing and Forwarding Lite is a feature that enables a service provider to:

- support two or more VPNs,
- allow IP addresses to overlap among VPNs, and
- distinguish routes for different VPNs using input interfaces.

VRF-lite forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be either physical, such as Ethernet ports, or logical, such as VLAN SVIs. A Layer 3 interface cannot belong to more than one VRF at any time.

VRF-lite interfaces must be Layer 3 interfaces.

Key devices in VRF-lite deployments

- Customer edge (CE) devices: Provide customer access to the service provider network over a data link to one or more provider edge routers. CE devices advertise local routes to the provider edge router and learn remote VPN routes from it. A switch can function as a CE device.
- Provider edge (PE) routers: Exchange routing information with CE devices using static routing or protocols like BGP, RIPv1, or RIPv2. A PE router maintains a VRF for each directly connected site and is only required to maintain VPN routes for those VPNs to which it is directly attached. Multiple interfaces on a PE router can be associated with a single VRF if all these sites participate in the same VPN. After learning local VPN routes from CE devices, a PE router exchanges VPN routing information with other PE routers using internal BGP (iBGP).
- Provider routers (core routers): Any routers in the service provider network that do not attach to CE devices.

Guidelines for configuring VRF-lite

- Customer sharing: Multiple customers can share one CE device. The shared CE device maintains separate VRF tables for each customer and switches or routes packets based on each customer's routing table.
- Privacy and security: VRF-lite allows a CE device to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.
- IP address reuse: Because customers use different VRF tables, you can reuse the same IP addresses across different VRFs.
- Physical link sharing: VRF-lite allows multiple customers to share the same physical link between the PE and the CE.
- Interface types: The switch supports configuring VRF using physical ports, VLAN SVIs, or a combination of both. You can connect SVIs through an access port or a trunk port.
- VLAN usage: A customer can use multiple VLANs as long as they do not overlap with those of other customers. A customer's VLANs are mapped to a specific routing table ID that identifies the appropriate routing tables stored on the switch.
- TCAM resource sharing: The Layer 3 TCAM resource is shared between all VRFs. To ensure that any one VRF has sufficient CAM space, use the `maximum routes` command.
- Route limits: A switch using VRF can support one global network and multiple VRFs. The total number of routes supported is limited by the size of the TCAM.
- IPv4 and IPv6 support: A single VRF can be configured for both IPv4 and IPv6.
- Packet handling: If an incoming packet's destination address is not found in the VRF table, the packet is dropped.
- Hardware switching: If insufficient TCAM space exists for a VRF route, hardware switching for that VRF is disabled, and the corresponding data packets are sent to software for processing.
- IPv4 specific protocols: The switch supports PIM-SM and PIM-SSM protocols for IPv4.
- IPv6 specific protocols: VRF-aware OSPFv3, EIGRPv6, and IPv6 static routing are supported.
- IPv6 VRF-aware applications: VRF-aware IPv6 route applications include ping, telnet, ssh, tftp, ftp, and traceroute.

VRF-lite deployment and operational management

Effective configuration and management of Virtual Routing and Forwarding Lite (VRF-lite) involves these tasks to ensure proper operation.

- VRF definition and interface association
- Routing protocol configuration
- Multicast configuration
- VRF-aware service configuration

VRF definition and interface association

To enable and configure essential network services to operate within specific VRF instances, perform these tasks:

- Configure per-VRF for TACACS+ servers
- Configure the user interface for uRPF
- Configure the user interface for NTP
- Configure the user interface for Telnet and SSH

Routing protocol configuration

To set up routing within VRF instances, including static routes and dynamic protocols, requires these configurations.

- Configure VRF static routes
- Configure OSPFv3 router process
- Enable OSPFv3 on an interface
- Configure EIGRPv6 routing process

Multicast configuration

- Configure multicast VRFs

Configure VRF-aware service

Establishing VRF instances and connecting them to network interfaces requires these configurations.

- Configure IPv4 VRFs
- Configure IPv6 VRFs
- Associate interfaces to defined VRFs

Configure per-VRF for TACACS+ servers

Configure per-virtual route forwarding (per-VRF) authentication, authorization, and accounting (AAA) on TACACS+ servers.

This configuration allows TACACS+ servers to operate within specific VRF instances, providing isolated AAA services for different VPNs.

Perform these steps to configure per-VRF for TACACS+ servers.

Before you begin

- Ensure AAA is configured.

- Ensure a server group is configured.

Procedure

Step 1 Configure a VRF table.

a) Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

b) Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

c) Use the **vrf definition** command to configure VRF and enter VRF configuration mode.

Example:

```
Switch(config)# vrf definition cisco
```

Note
You must have a Network Advantage license to configure VRF Definition.

d) Use the **rd** command to create routing and forwarding tables for a VRF instance by specifying a route distinguisher.

Example:

```
Switch(config-vrf)# rd 100:1
```

e) Use the **exit** command to exit VRF configuration mode.

Example:

```
Switch(config-vrf)# exit
```

Step 2 Configure an interface for VRF

a) Use the **interface** command to configure an interface and enter interface configuration mode.

Example:

```
Switch(config)# interface Loopback0
```

b) Use the **vrf forwarding** command to configure a VRF for the interface.

Example:

```
Switch(config-if)# vrf forwarding cisco
```

c) Use the **ip address** command to set a primary or secondary IP address for an interface.

Example:

```
Switch(config-if)# ip address 10.0.0.2 255.0.0.0
```

d) Use the **exit** command to exit interface configuration mode.

Example:

```
Switch(config-if)# exit
```

Step 3 Configure a TACACS+ server group with VRF

a) Use the **aaa group server tacacs+** command to group different TACACS+ server hosts into distinct lists and methods, and enter server-group configuration mode.

Example:

```
Switch(config)# aaa group server tacacs+ cisco
```

b) Use the **server-private** command to configure the IP address of the private TACACS+ server for the group server.

Example:

```
Switch(config-sg-tacacs+)# server-private 10.0.0.3
```

c) Use the **vrf forwarding** command to configure the VRF reference of a AAA TACACS+ server group.

Example:

```
Switch(config-sg-tacacs+)# vrf forwarding cisco
```

d) Use the **ip tacacs source-interface** command to use the IP address of a specified interface for all outgoing TACACS+ packets.

Example:

```
Switch(config-sg-tacacs+)# ip tacacs source-interface Loopback0
```

e) Use the **exit** command to exit server-group configuration mode.

Example:

```
Switch(config-sg-tacacs+)# exit
```

Configure the user interface for uRPF

You can enable Unicast Reverse Path Forwarding (uRPF) on an interface assigned to a VRF.

Configuring uRPF on a VRF interface enhances security by verifying the source IP address of incoming packets, ensuring they are reachable via the interface they arrived on. Source lookup is performed in the VRF table.

Perform these steps to enable the user interface for uRPF.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **interface** command to enter interface configuration mode and specify the Layer 3 interface.

Example:

```
Switch(config)# interface GigabitEthernet0/1
```

Step 4 Use the **no switchport** command to remove the interface from Layer 2 configuration mode.

Example:

```
Switch(config)# no switchport
```

Step 5 Use the **vrf forwarding** command to configure a VRF for the interface.

Example:

```
Switch(config-if)# vrf forwarding multiVrfA
```

Step 6 Use the **ipv6 address** command to assign IPv6 address to the interface.

Example:

```
Switch(config-if)# ipv6 address 2001:DB8:1::1/64
```

Step 7 Use the **ipv6 verify unicast** command to enable uRPF on the interface.

Example:

```
Switch(config-if)# ipv6 verify unicast source reachable-via rx allow-default
```

Step 8 Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-if)# end
```

uRPF is enabled on the specified VRF interface, enhancing packet source validation.

Configure the user interface for NTP

You can configure an NTP server or peer for time synchronization within a specified VRF.

This task ensures that devices within a VRF maintain accurate time synchronization by using an NTP server or peer that is also part of that VRF.

Perform these steps to configure the user interface for NTP.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **ntp server vrf** *vrf-name* command to configure the NTP server in the specified VRF.

Example:

```
Switch(config)# ntp server vrf MyVRF 2001:DB8::10
```

Step 4 Use the **ntp peer vrf** *vrf-name*command to configure the NTP peer in the specified VRF.

Example:

```
Switch(config)# ntp server vrf MyVRF 2001:DB8::10
```

The NTP server or peer is configured for the specified VRF, enabling time synchronization for devices within that VRF.

Configure the user interface for Telnet and SSH

You can connect to an IPv6 host via Telnet or SSH within a specified VRF.

This task allows secure remote access to devices that are part of a specific VRF instance.

Perform these steps to configure the user interface for Telnet and SSH.

Procedure

Step 1 Use the **telnet <ipv6-address>/vrf** command to connect through Telnet to an IPv6 host or address in the specified VRF.

Example:

```
Switch# telnet 2001:DB8::1/vrf VRF-A
```

Step 2 Use the **ssh -l username -vrf** *vrf-name ipv6-host* command to connect through SSH to an IPv6 host or address in the specified VRF.

Example:

```
Switch# ssh -l admin -vrf VRF-B 2001:DB8::2
```

A Telnet or SSH connection is established to the specified IPv6 host within the VRF.

Configure VRF static routes

You can configure static routes specific to a VRF.

This task allows you to define explicit paths for traffic within a VRF, which is useful for directing traffic to specific destinations or next-hops that are part of that VRF.

Perform these steps to configure VRF static routes.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **ipv6 route** command to configure static routes specific to VRF.

Example:

```
Switch(config)# ipv6 route vrf v6a 7000::/64 GigabitEthernet 1/0/1 4000::2
```

The syntax includes ipv6 route [vrf vrf-name] ipv6-prefix/prefix-length {ipv6-address | interface-type interface-number [ipv6-address]}

Table 1: Syntax description

Keyword	Description
vrf	Virtual routing table instance for network segregation.
vrf-name	Identifier name assigned to a VRF instance.
ipv6-prefix/prefix-length	IPv6 network prefix with subnet mask length.
ipv6-address	IPv6 address assigned to interface.
interface-type	Type of physical or logical network interface.
interface-number	Numerical identifier of the network interface.

A static route is added to the specified VRF's routing table.

Configure OSPFv3 router process

You can configure the OSPFv3 router process for IPv6 within a VRF.

This task enables dynamic routing using OSPFv3 for IPv6 traffic within a specific VRF, allowing the VRF to exchange routing information with other OSPFv3-enabled devices.

Perform these steps to configure OSPFv3 router process.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **router ospfv3** command to enable OSPFv3 router configuration mode for the IPv6 address family.

Example:

```
Switch(config)# router ospfv3 1
```

Step 4 Use the **area** command to configure OSPFv3 area.

Example:

```
Switch(config-router)# area 1
```

Step 5 Use the **router-id** command to configure router ID.

Example:

```
Switch(config-router)# router-id 1.1.1.1
```

Step 6 Use the **address-family** command to enter IPv6 address family configuration mode for OSPFv3 in VRF.

Example:

```
Switch(config-router)# address-family ipv6 unicast
```

Step 7 Use the **exit-address-family** command to exit from address-family configuration mode.

Example:

```
Switch(config-router-af)# exit-address-family
```

Step 8 Use the **end** command to privileged EXEC mode.

Example:

```
Switch(config-router)# end
```

The OSPFv3 router process is configured for IPv6 unicast within the specified VRF.

Enable OSPFv3 on an interface

You can enable OSPFv3 on a specific interface with IPv6 address family.

This task activates OSPFv3 on a Layer 3 interface, allowing it to participate in OSPFv3 routing for IPv6 within its configured area.

Perform these steps to enable OSPFv3 on an interface.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **interface** command to enter into interface configuration mode.

Example:

```
Switch(config)# interface GigabitEthernet2/1
```

Step 4 Use the **no switchport** command to remove the interface from Layer 2 configuration mode.

Example:

```
Switch(config)# no switchport
```

Step 5 Use the **ipv6 enable** command to enable IPv6 on the interface.

Example:

```
Switch(config-if)# ipv6 enable
```

Step 6 Use the **ipv6 address** command to assign IPv6 address to the interface.

Example:

```
Switch(config-if)# ipv6 address 2001:DB8:1::1/64
```

Step 7 Use the **ipv6 ospf** command to enable OSPFv3 on an IPv6 interface with address-family (AF).

Example:

```
Switch(config-if)# ipv6 ospf 1 area 0
```

Syntax includes **ipv6 { ospfv3 <process-id> area <area-ID>}**

Table 2: Syntax description

Keyword	Description
process-id	Locally assigned identifier for OSPFv3 routing process.
area	Logical grouping of routers within OSPFv3 for routing.
area-ID	Unique identifier for an OSPFv3 area, decimal or IPv6 prefix.

Step 8 Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-if)# end
```

OSPFv3 is enabled on the specified interface for IPv6.

Configure EIGRPv6 route process

You can configure the EIGRPv6 routing process within a VRF-Lite environment.

This task enables dynamic routing using EIGRP for IPv6 traffic within a specific VRF, allowing the VRF to exchange routing information with other EIGRPv6-enabled devices.

Perform these steps to configure EIGRPv6 router process.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **router eigrp** command to enter into EIGRP configuration mode.

Example:

```
Switch(config)# router eigrp test
```

Step 4 Use the **address-family** command to enter IPv6 address family configuration mode for OSPFv3 in VRF

Example:

```
Switch(config-router)# address-family ipv6 unicast
```

Step 5 Use the **topology** command to configure an EIGRP process to route IP traffic under the specified topology instance and enter address family topology configuration mode.

Example:

```
Switch(config-router-af)# topology base
```

Step 6 Use the **exit-af-topology** command to exit from address family topology configuration mode.

Example:

```
Switch(config-router-af-topology)# exit-af-topology
```

Step 7 Use the **eigrp router-id** command to configure the EIGRP router ID for a specific router.

Example:

```
Switch(config-router)# eigrp router-id 2.3.4.5
```

Step 8 Use the **exit-address-family** command to exit from address family configuration mode.

Example:

```
Switch(config-router)# exit-address-family
```

Step 9 Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-router)# end
```

The EIGRPv6 routing process is configured for the specified VRF.

Configure multicast VRFs

You can configure multicast routing within a specific VRF table.

This task enables you to isolate multicast traffic and routing within a VRF, which is essential for supporting multicast applications in a multi-VPN environment.

Perform these steps to configure multicast VRFs.

Procedure

Step 1 Configure a VRF table with multicast routing

a) Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

b) Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

c) Use the **ip routing** command to enable IP routing.

Example:

```
Switch(config)# ip routing
```

d) Use the **vrf definition** command to configure a VRF table and enter VRF configuration mode.

Example:

```
Switch(config)# vrf definition multiVrfA
```

e) Use the **ip multicast-routing vrf** command to optionally enabling multicast routing within the VRF.

Example:

```
Switch(config-vrf)# ip multicast-routing vrf multiVrfA
```

f) Use the **rd** command to create routing and forwarding tables for a VRF instance by specifying a route distinguisher.

Example:

```
Switch(config-vrf)# rd 100:1
```

g) Use the **route-target** command to create a list of import, export, or import and export route target communities for the specified VRF.

Example:

```
Switch(config-vrf)# route-target export 100:1
```

Enter route target either an AS number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y).

Note
The route-target-ext-community value should be the same as the route-distinguisher value.

h) Use the **import map** command to optionally associate a route map with the VRF.

Example:

```
Switch(config-vrf)# import map route-map
```

Step 2 Configure an interface for VRF and PIM

a) Use the **interface** command to configure an interface and enter interface configuration mode.

Example:

```
Switch(config)# interface GigabitEthernet1/1
```

b) Use the **vrf forwarding** command to configure a VRF for the interface.

Example:

```
Switch(config-if)# vrf forwarding multiVrfA
```

c) Use the **ip address** command to set a primary or secondary IP address for an interface.

Example:

```
Switch(config-if)# ip address 172.21.200.203 255.255.255.0
```

d) Use the **ip pim sparse-mode** command to enable PIM on the VRF-associated Layer 3 interface.

Example:

```
Switch(config-if)# ip pim sparse-mode
```

e) Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-if)# end
```

Step 3 (Optional) Use the **show vrf definition** command to verify the configuration by displaying information about the configured VRFs.

Example:

```
Switch# show vrf definition
```

Multicast routing is enabled and configured within the specified VRF.

Configure IPv4 VRFs

You can configure a Virtual Routing and Forwarding (VRF) instance for IPv4.

This task enables the creation of separate routing and forwarding tables for different customer networks or services, allowing for IP address overlap and network isolation.

Perform these steps to configure IPv4 VRFs.

Procedure

Step 1 Configure a VRF

a) Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

b) Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

c) Use the **vrf definition** command to configure a VRF table and enter VRF configuration mode.

Example:

```
Switch(config)# vrf definition multiVrfA
```

d) Use the **rd** command to create routing and forwarding tables for a VRF instance by specifying a route distinguisher.

Example:

```
Switch(config-vrf)# rd 100:1
```

e) Use the **route-target** command to create a list of import, export, or import and export route target communities for the specified VRF.

Example:

```
Switch(config-vrf)# route-target export 100:1
```

Enter route target either an AS number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The route-target-ext-community value should be the same as the route-distinguisher value.

f) Use the **import map** command to optionally associate a route map with the VRF.

Example:

```
Switch(config-vrf)# import map route-map
```

Step 2 Configure an interface for VRF

a) Use the **interface** command to configure an interface and enter interface configuration mode.

Example:

```
Switch(config)# interface GigabitEthernet1/1
```

b) Use the **vrf forwarding** command to configure a VRF for the interface.

Example:

```
Switch(config-if)# vrf forwarding multiVrfA
```

c) Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-if)# end
```

Step 3 (Optional) Use the **show vrf definition** command to verify the configuration by displaying information about the configured VRFs.

Example:

```
Switch# show vrf definition
VRF multiVrfA; default RD 100:1; default VPNID <not set>
Interfaces:
  GigabitEthernet1/1
VRF Table ID = 1
Export VPN route-target communities
  RT:100:1
No import route-map
```

```
Import route-map route-map
No export route-map
VRF label distribution protocol: not configured
VRF label allocation mode: per-prefix
```

An IPv4 VRF is configured and associated with a Layer 3 interface.

What to do next

- To delete a VRF and remove all interfaces from it, use the **no vrf definition** global configuration command.
- To remove an interface from the VRF, use the **no vrf forwarding** interface configuration command.

Configure IPv6 VRFs

You can configure a Virtual Routing and Forwarding (VRF) instance for IPv6.

This task enables the creation of separate routing and forwarding tables for different customer networks or services, allowing for IP address overlap and network isolation for IPv6 traffic.

Perform these steps to configure IPv6 VRFs.

Procedure

Step 1 Configure a VRF

a) Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

b) Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

c) Use the **vrf definition** command to name the VRF and enter VRF configuration mode.

Example:

```
Switch# vrf definition red
```

d) (Optional) Use the **rd** command to create routing and forwarding tables for a VRF instance by specifying a route distinguisher.

Example:

```
Switch(config-vrf)# rd 100:1
```

e) (Optional) Use the **address-family** command to specify the address family within the VRF.

Example:

```
Switch(config-vrf)# address-family ipv6
```

f) (Optional) Use the **route-target** command to create a list of import, export, or import and export route target communities for the specified VRF.

Example:

```
Switch(config-vrf-af)# route-target both 65000:1
```

Enter either an AS system number and an arbitrary number (`xxx:y`) or an IP address and an arbitrary number (`A.B.C.D:y`).

Note
This command is effective only if BGP is running.

g) Use the **end** command to return to privileged EXEC mode

Example:

```
Switch(config-vrf-af)# end
```

Step 2 Address-Family and multicast configuration for VRF.

a) Use the **vrf definition** command to name the VRF and enter VRF configuration mode

Example:

```
Switch# vrf definition red
```

b) Use the **exit-address-family** command to exit VRF address-family configuration mode and return to VRF configuration mode.

Example:

```
Switch(config-vrf-af)# exit-address-family
```

c) Use the **ipv6 multicast multitopology** command to enable multicast specific RPF topology within the VRF.

Example:

```
Switch(config-vrf)# ipv6 multicast multitopology
```

d) Use the **address-family ipv6 multicast** command to enter multicast IPv6 address-family configuration mode within the VRF.

Example:

```
Switch(config-vrf)# address-family ipv6 multicast
```

e) Use the **end** command to return to privileged EXEC mode

Example:

```
Switch(config-vrf-af)# end
```

An IPv6 VRF is configured, optionally with multicast capabilities, and ready for interface association.

Associate interfaces to defined VRFs

You can associate Layer 3 interfaces with previously defined VRF instances.

This task links physical or logical interfaces to specific VRFs, ensuring that traffic entering or exiting these interfaces is routed according to the VRF's routing table.

Perform these steps to associte interfaces to defined VRFs.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **interface** command to enter interface configuration mode and specify the Layer 3 interface.

Example:

```
Switch(config)# interface GigabitEthernet0/1
```

Step 4 Use the **no switchport** command to remove the interface from Layer 2 configuration mode.

Example:

```
Switch(config)# no switchport
```

Step 5 Use the **vrf forwarding** command to configure a VRF for the interface.

Example:

```
Switch(config-if)# vrf forwarding multiVrfA
```

Step 6 Use the **ipv6 enable** command to enable IPv6 on the interface.

Example:

```
Switch(config-if)# ipv6 enable
```

Step 7 Use the **ipv6 address** command to assign IPv6 address to the interface.

Example:

```
Switch(config-if)# ipv6 address 2001:DB8:1::1/64
```

Step 8 Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-if)# end
```

Step 9 (Optional) Use the **show ipv6 vrf** command to monitor the configured VRF.

Example:

```
Switch# show ipv6 vrf
VRF Name                           VRF ID   State      Interfaces
---------------------------------  -------  ---------  -------------------------
red                                1        Up         GigabitEthernet0/1
                                                         Loopback0
blue                               2        Up         GigabitEthernet0/2
                                                         Vlan10
```

The specified interface is associated with the VRF, and IPv6 is enabled on it.

VRF-aware services for IPv6

IPv6 services can be configured on global interfaces and within the global routing instance. These services are enhanced to run on multiple routing instances, making them VRF-aware. Any configured VRF in the system can be specified for a VRF-aware service.

VRF-aware services are implemented in platform-independent modules. VRF provides multiple routing instances in Cisco IOS. Each platform has its own limit on the number of VRFs it supports.

Key characteristics of VRF-aware IPv6 services

- Users can ping a host within a user-specified VRF.
- Neighbor Discovery entries are learned in separate VRFs. Users can display Neighbor Discovery (ND) entries for specific VRFs.

These services are VRF-aware for IPv6:

- Ping
- Unicast Reverse Path Forwarding (uRPF)
- Traceroute
- FTP and TFTP
- Telnet and SSH
- NTP

VRF-aware services characteristics

IP services can be configured on global interfaces and within the global routing instance. These services are enhanced to run on multiple routing instances, making them VRF-aware. Any configured VRF in the system can be specified for a VRF-aware service.

VRF-aware services are implemented in platform-independent modules. VRF provides multiple routing instances in Cisco IOS. Each platform has its own limit on the number of VRFs it supports.

Key characteristics of VRF-aware services

- Users can ping a host within a user-specified VRF.
- ARP entries are learned in separate VRFs. Users can display Address Resolution Protocol (ARP) entries for specific VRFs.

Commands for verification and troubleshooting

To verify the operational status, connectivity, and configuration of VRF instances, perform these tasks:

- Configure the user interface for ARP
- Configure the user interface for ping

- Configure the user interface for traceroute
- Display IPv4 VRF-lite status

Configure the user interface for ARP

Display or create static ARP entries within a specified VRF

This task allows you to manage ARP entries for specific VRF instances, which helps in troubleshooting and ensuring proper network communication within a VRF.

Perform these steps to configure the user interface for ARP.

Procedure

Step 1 Use the **enable** command to enter privileged EXEC mode.

Example:

```
Switch> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Switch# configure terminal
```

Step 3 Use the **arp vrf** command to create a static ARP entry in the specified VRF.

Example:

```
Switch(config)# arp vrf multiVrfA 192.168.1.10 0800.0900.1234 ARPA
```

Step 4 Use the **end** command to return to privileged EXEC mode.

Example:

```
Switch(config-if-vrrp)# end
```

Step 5 (Optional) Use the **show ip arp vrf** command to display the ARP table in the specified VRF.

Example:

```
Switch# show ip arp vrf vrf-nameSwitch# show ip arp vrf MyVRF
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.0.0.1         -          0001.0001.0001  ARPA   GigabitEthernet0/0 (static)
Internet  10.0.0.10        1          000a.000a.000a  ARPA   GigabitEthernet0/0
Internet  10.0.0.11        0          000b.000b.000b  ARPA   GigabitEthernet0/0
Internet  10.0.0.20        5          0014.0014.0014  ARPA   GigabitEthernet0/1
```

You can view existing ARP entries or add new static ARP entries for a specific VRF.

Configure the user interface for ping

You can ping an IPv6 host or address within a specified VRF.

This task allows you to perform network reachability tests for devices within a particular VRF instance.

Procedure

(Optional) Use the **ping vrf** command to ping an IPv6 host or address in the specified VRF.

Example:

```
Switch# ping vrf cisco ipv6-host
```

The system attempts to ping the specified IPv6 host within the VRF, and the results are displayed.

Configure the user interface for traceroute

You can trace the route to an IPv6 address within a specified VRF.

This task diagnoses network path issues for devices located within a particular VRF instance.

Procedure

(Optional) Use the **traceroute vrf** command to trace the route to an IPv6 address in the specified VRF.

Example:

```
Switch# traceroute vrf cisco 2001:DB8::1
```

The system attempts to ping the specified IPv6 host within the VRF, and the results are displayed.

Monitor IPv4 VRF-lite status

You can monitor VRF-lite configuration and status for IPv4.

This task helps in verifying the operational status and routing information of configured IPv4 VRF-lite instances.

Procedure

Step 1 Use the **show ip protocols** command to display routing protocol information associated with a VRF.

Example:

```
Switch# show ip protocols vrf MyVRF
Routing Protocol is "eigrp 1"
  EIGRP-IPv6 VRF(MyVRF) (AS 65000)
  Router ID 10.10.10.1
  Metric weights: K1=1, K2=0, K3=1, K4=0, K5=0
  Metric of this router is 25600
  Maximum path: 4
  Maximum hopcount 100
  Maximum metric variance 1
```

```
  Redistributing:
    connected (metric 2000000 1000 255 1 1500)
  Interfaces:
    GigabitEthernet0/1
    Loopback0
  Passive Interface(s):
    Loopback0
  Automatic network summarization is not in effect
  EIGRP-IPv6 VRF(MyVRF) for AS 65000
    Neighbors:
      2001:DB8:1:1::2 (GigabitEthernet0/1)
```

Step 2 Use the **show ip route vrf** command to display IP routing table information associated with a VRF.

Example:

Switch# **show ip route vrf MyVRF**

```
Routing Table: MyVRF
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/1
L        192.168.1.1/32 is directly connected, GigabitEthernet0/1
D     10.0.0.0/8 [90/307200] via 192.168.1.2, 00:01:30, GigabitEthernet0/1
S     172.16.0.0/16 [1/0] via 192.168.1.3
```

Step 3 Use the **show vrf definition** command to display information about the defined VRF instances.

Example:

```
Switch# show vrf definition
VRF Name                          VRF ID   State      Interfaces
--------------------------------  -------  ---------  -------------------------
MyVRF                             1        Up         GigabitEthernet0/1
                                                        Loopback0
AnotherVRF                        2        Up         GigabitEthernet0/2
                                                        Vlan10
```

Step 4 Use the **bidir vrf** command to display bidirectional forwarding information associated with a VRF instance.

Example:

```
Switch# bidir vrf MyVRF active
PIM Bidir-RPFs for VRF MyVRF
  Group: 2001:DB8:2::/64, RP: 2001:DB8:2:2::1
    Active: Yes
    Uptime: 00:15:23
    Incoming Interface: GigabitEthernet0/1
    Outgoing Interface: Null0
    Flags: S (Sparse-mode), B (Bidir)
  Group: 2001:DB8:3::/64, RP: 2001:DB8:3:3::1
    Active: Yes
    Uptime: 00:08:45
    Incoming Interface: GigabitEthernet0/2
```

```
Outgoing Interface: Null0
Flags: S (Sparse-mode), B (Bidir)
```

IPv6 VRF-lite configuration example

This example illustrates how to use OSPFv3 for CE-PE routing in an IPv6 VRF-lite setup. The topology involves these three switches with VRF instances `v1` and `v2` configured.

- CE1,
- PE, and
- CE2

Configure CE1 switch

```
ipv6 unicast-routing
vrf definition v1
 rd 100:1
 !
 address-family ipv6
 exit-address-family
 !
vrf definition v2
 rd 200:1
 !
 address-family ipv6
 exit-address-family
 !
interface Vlan100
 vrf forwarding v1
 ipv6 address 1000:1::1/64
 ospfv3 100 ipv6 area 0
 !
interface Vlan200
 vrf forwarding v2
 ipv6 address 2000:1::1/64
 ospfv3 200 ipv6 area 0
 !
interface GigabitEthernet 1/0/1
 switchport access vlan 100
 end
interface GigabitEthernet 1/0/2
 switchport access vlan 200
 end
interface GigabitEthernet 1/0/24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 end
router ospfv3 100
 router-id 10.10.10.10
 !
 address-family ipv6 unicast vrf v1
  redistribute connected area 0 normal
 exit-address-family
 !
router ospfv3 200
 router-id 20.20.20.20
 !
 address-family ipv6 unicast vrf v2
```

```
  redistribute connected area 0 normal
 exit-address-family
 !
```

Configure PE switch

```
ipv6 unicast-routing
vrf definition v1
 rd 100:1
 !
 address-family ipv6
 exit-address-family
 !
vrf definition v2
 rd 200:1
 !
 address-family ipv6
 exit-address-family
 !
interface Vlan600
 vrf forwarding v1
 no ipv6 address
 ipv6 address 1000:1::2/64
 ospfv3 100 ipv6 area 0
 !
interface Vlan700
 vrf forwarding v2
 no ipv6 address
 ipv6 address 2000:1::2/64
 ospfv3 200 ipv6 area 0
 !
interface Vlan800
 vrf forwarding v1
 ipv6 address 3000:1::7/64
 ospfv3 100 ipv6 area 0
 !
interface Vlan900
 vrf forwarding v2
 ipv6 address 4000:1::7/64
 ospfv3 200 ipv6 area 0
 !
interface GigabitEthernet 1/0/1
 switchport trunk encapsulation dot1q
 switchport mode trunk
 exit
interface GigabitEthernet 1/0/2
 switchport trunk encapsulation dot1q
 switchport mode trunk
 exit
router ospfv3 100
 router-id 30.30.30.30
 !
 address-family ipv6 unicast vrf v1
  redistribute connected area 0 normal
 exit-address-family
 !
 address-family ipv6 unicast vrf v2
  redistribute connected area 0 normal
 exit-address-family
 !
```

Configure CE2 switch

```
ipv6 unicast-routing
vrf definition v1
```

```
 rd 100:1
 !
 address-family ipv6
 exit-address-family
 !
vrf definition v2
 rd 200:1
 !
 address-family ipv6
 exit-address-family
 !
interface Vlan100
 vrf forwarding v1
 ipv6 address 1000:1::3/64
 ospfv3 100 ipv6 area 0
 !
interface Vlan200
 vrf forwarding v2
 ipv6 address 2000:1::3/64
 ospfv3 200 ipv6 area 0
 !
interface GigabitEthernet 1/0/1
 switchport access vlan 100
 end
interface GigabitEthernet 1/0/2
 switchport access vlan 200
 end
interface GigabitEthernet 1/0/24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 end
router ospfv3 100
 router-id 40.40.40.40
 !
 address-family ipv6 unicast vrf v1
  redistribute connected area 0 normal
 exit-address-family
 !
router ospfv3 200
 router-id 50.50.50.50
 !
 address-family ipv6 unicast vrf v2
  redistribute connected
 area 0 normal
 exit-address-family
 !
```

VPN co-existence between IPv4 and IPv6

Backward compatibility exists between the "older" CLI for configuring IPv4 and the "new" CLI for IPv6. This means that a configuration might contain both CLIs. The IPv4 CLI retains the ability to have an IP address defined within a VRF and an IPv6 address defined in the global routing table on the same interface.

Consider this configuration snippet:

```
vrf definition red
 rd 100:1
 address family ipv6
  route-target both 200:1
 exit-address-family
!
vrf definition blue
```

```
 rd 200:1
 route-target both 200:1
!
interface GigabitEthernet1/1
 vrf forwarding red
 ip address 50.1.1.2 255.255.255.0
 ipv6 address 4000::72B/64
!
interface GigabitEthernet1/2
 vrf forwarding blue
 ip address 60.1.1.2 255.255.255.0
 ipv6 address 5000::72B/64
```

In this example:

- For `GigabitEthernet1/1`, all addresses (IPv4 and IPv6) refer to VRF `red`.
- For `GigabitEthernet1/2`, the IPv4 address refers to VRF `blue`, but the IPv6 address refers to the global IPv6 routing table.