Configuring Layer 3 Virtualization

This chapter describes how to configure Layer 3 virtualization on the Cisco NX-OS device.

This chapter includes the following sections:

About Layer 3 Virtualization

Cisco NX-OS supports multiple virtual routing and forwarding instances (VRFs). Each VRF contains a separate address space with unicast and multicast route tables for IPv4 and IPv6 and makes routing decisions independent of any other VRF.

Each router has a default VRF and a management VRF:

Management VRF

- The management VRF is for management purposes only.
- Only the mgmt 0 interface can be in the management VRF.
- The mgmt 0 interface cannot be assigned to another VRF.
- No routing protocols can run in the management VRF (static only).

Default VRF

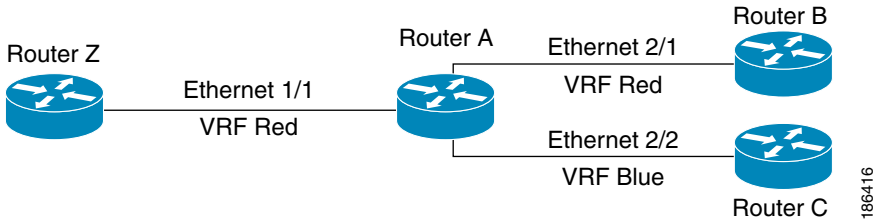
- All Layer 3 interfaces exist in the default VRF until they are assigned to another VRF.
- Routing protocols run in the default VRF context unless another VRF context is specified.
- The default VRF uses the default routing context for all **show** commands.
- The default VRF is similar to the global routing table concept in Cisco IOS.

VRF and Routing

All unicast and multicast routing protocols support VRFs. When you configure a routing protocol in a VRF, you set routing parameters for the VRF that are independent of routing parameters in another VRF for the same routing protocol instance.

You can assign interfaces and route protocols to a VRF to create virtual Layer 3 networks. An interface exists in only one VRF. Figure 13-1 shows one physical network split into two virtual networks with two VRFs. Routers Z, A, and B exist in VRF Red and form one address domain. These routers share route updates that do not include Router C because Router C is configured in a different VRF.

Figure 13-1 VRFs in a Network



By default, Cisco NX-OS uses the VRF of the incoming interface to select which routing table to use for a route lookup. You can configure a route policy to modify this behavior and set the VRF that Cisco NX-OS uses for incoming packets.

VRF Route Leaking

Cisco NX-OS supports route leaking between VRFs.

You can import IP prefixes from the global routing table (the default VRF) into any other VRF by using an import policy or export IP prefixes from a non-default VRF into the default VRF using an export policy. The VRF import and export policies uses a route map to specify the prefixes to be imported or exported into a VRF. The policies can import or export IPv4 and IPv6 unicast prefixes.

Note Routes in the BGP default VRF can be imported directly. Any other routes in the default VRF should be redistributed into BGP first.

IP prefixes are defined as match criteria for the import or export route map through standard route policy filtering mechanisms. For example, you can create an IP prefix list or an as-path filter to define an IP prefix or IP prefix range and use that prefix list or as-path filter in a match clause for the route map. Prefixes that pass through the route map are imported or exported into the specified VRF using the import or export policy. Any route or path that is imported from another VRF cannot be imported or exported again.

For more information, see the "Guidelines and Limitations for VRF Route Leaking" section on page 13-5.

VRF-Aware Services

A fundamental feature of the Cisco NX-OS architecture is that every IP-based feature is VRF aware.

The following VRF-aware services can select a particular VRF to reach a remote server or to filter information based on the selected VRF:

- AAA—See the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide* for more information.
- Call Home—See the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide* for more information.
- DNS—See Chapter 4, "Configuring DNS" for more information.
- HTTP—See the *Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide* for more information.
- HSRP—See Chapter 17, "Configuring HSRP," for more information.
- NTP—See the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide* for more information.
- RADIUS—See the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide* for more information.
- Ping and Traceroute —See the *Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide* for more information.
- SSH—See the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide* for more information.
- SNMP—See the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide* for more information.
- Syslog—See the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide* for more information.
- TACACS+—See the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide* for more information.
- TFTP—See the *Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide* for more information.
- VRRP—See Chapter 18, "Configuring VRRP," for more information.
- XML—See the *Cisco NX-OS XML Management Interface User Guide* for more information.

See the appropriate configuration guide for each service for more information on configuring VRF support in that service.

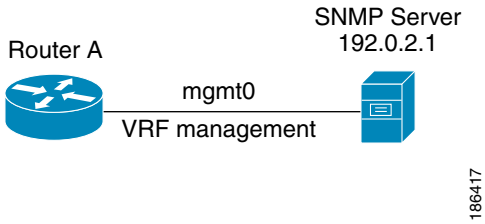
This section contains the following topics:

Reachability

Reachability indicates which VRF contains the routing information needed to get to the server providing the service. For example, you can configure an SNMP server that is reachable on the management VRF. When you configure that server address on the router, you also configure which VRF that Cisco NX-OS must use to reach the server.

Figure 13-2 shows an SNMP server that is reachable over the management VRF. You configure Router A to use the management VRF for the SNMP server host 192.0.2.1.

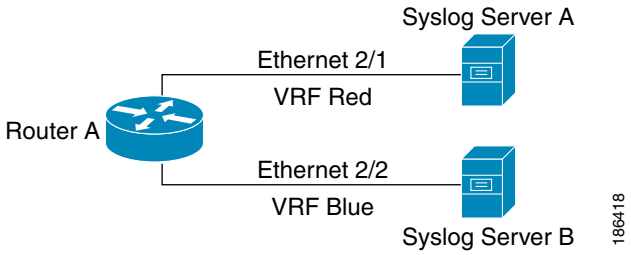
Figure 13-2 Service VRF Reachability



Filtering

Filtering allows you to limit the type of information that goes to a VRF-aware service based on the VRF. For example, you can configure a syslog server to support a particular VRF. Figure 13-3 shows two syslog servers with each server supporting one VRF. syslog server A is configured in VRF Red, so Cisco NX-OS sends only system messages generated in VRF Red to syslog server A.

Figure 13-3 Service VRF Filtering

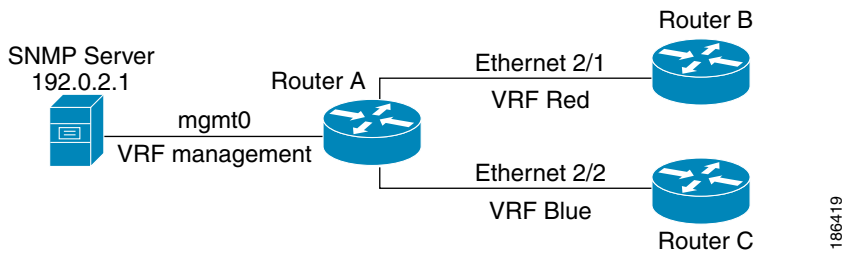


Combining Reachability and Filtering

You can combine reachability and filtering for VRF-aware services. You can configure the VRF that Cisco NX-OS uses to connect to that service as well as the VRF that the service supports. If you configure a service in the default VRF, you can optionally configure the service to support all VRFs.

Figure 13-4 shows an SNMP server that is reachable on the management VRF. You can configure the SNMP server to support only the SNMP notifications from VRF Red, for example.

Figure 13-4 Service VRF Reachability Filtering



Licensing Requirements for VRFs

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	VRFs require no license. Any feature not included in a license package is bundled with the nx-os image and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*. VRF route leaking requires an Enterprise Services license. For a complete explanation of the Cisco NX-OS licensing scheme and how to obtain and apply licenses, see the *Cisco NX-OS Licensing Guide.*

Guidelines and Limitations for VRFs

VRFs have the following configuration guidelines and limitations:

- When you make an interface a member of an existing VRF, Cisco NX-OS removes all Layer 3 configurations. You should configure all Layer 3 parameters after adding an interface to a VRF.
- You should add the mgmt0 interface to the management VRF and configure the mgmt0 IP address and other parameters after you add it to the management VRF.
- If you configure an interface for a VRF before the VRF exists, the interface is operationally down until you create the VRF.
- Cisco NX-OS creates the default and management VRFs by default. You should make the mgmt0 interface a member of the management VRF.
- The **write erase boot** command does not remove the management VRF configurations. You must use the **write erase** command and then the **write erase boot** command.

Guidelines and Limitations for VRF Route Leaking

VRF route leaking has the following configuration guidelines and limitations:

- Route leaking is supported between any two non-default VRFs and from the default VRF to a non-default VRF. Beginning with Cisco NX-OS Release 7.0(3)I2(1), route leaking is also supported from a non-default VRF to the default VRF.
- You can restrict route leaking to specific routes using route map filters to match designated IP addresses.
- By default, the maximum number of IP prefixes that can be imported from the default VRF into a non-default VRF is 1000 routes.
- There is no limit on the number of routes that can be leaked between two non-default VRFs.
- VRF route leaking requires an Enterprise license, and BGP must be enabled.

Default Settings

Table 13-1 lists the default settings for VRF parameters.

Table 13-1 Default VRF Parameters

Parameters	Default
Configured VRFs	Default, management
Routing context	Default VRF
Prefix limit for VRF route leaking	1000

Configuring VRFs

This section contains the following topics:

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Creating a VRF

You can create a VRF.

SUMMARY STEPS

1. **configure terminal**
2. **vrf context** *vrf-name*
3. (Optional) **ip route** {*ip-prefix* | *ip-addr ip-mask*} {[*next-hop* | *nh-prefix*] | [*interface next-hop* | *nh-prefix*]} [**tag** *tag-value* [*pref*]
4. (Optional) **show vrf** [*vrf-name*]
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vrf context** *name*
	
	Example:
	`switch(config)# vrf context Enterprise`
	`switch(config-vrf)#`
		Creates a new VRF and enters VRF configuration mode. The *name* can be any case-sensitive, alphanumeric string up to 32 characters.

Step 3	**ip route** {*ip-prefix* | *ip-addr ip-mask*} {[*next-hop* | *nh-prefix*] | [*interface next-hop* | *nh-prefix*]} [**tag** *tag-value* [*pref*] **Example**: `switch(config-vrf)# ip route 192.0.2.0/8 ethernet 1/2 192.0.2.4`	(Optional) Configures a static route and the interface for this static route. You can optionally configure the next-hop address. The *preference* value sets the administrative distance. The range is from 1 to 255. The default is 1.
Step 4	**show vrf** [*vrf-name*] **Example**: `switch(config-vrf)# show vrf Enterprise`	(Optional) Displays VRF information.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves this configuration change.

To delete the VRF and the associated configuration, use the following command in global configuration mode:

Command	Purpose
no vrf context *name* **Example:** `switch(config)# no vrf context Enterprise`	Deletes the VRF and all associated configurations.

Any commands available in global configuration mode are also available in VRF configuration mode.

This example shows how to create a VRF and add a static route to the VRF:

```
switch# configure terminal
switch(config)# vrf context Enterprise
switch(config-vrf)# ip route 192.0.2.0/8 ethernet 1/2
switch(config-vrf)# exit
switch(config)# copy running-config startup-config
```

Assigning VRF Membership to an Interface

You can make an interface a member of a VRF.

BEFORE YOU BEGIN

Assign the IP address for an interface after you have configured the interface for a VRF.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-type slot/port*
3. **vrf member** *vrf-name*
4. **ip-address** *ip-prefix/length*
5. (Optional) **show vrf** *vrf-name* **interface** *interface-type number*
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** *interface-type slot/port*
	
	Example:
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		Enters interface configuration mode.

Step 3	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member RemoteOfficeVRF`	Adds this interface to a VRF.
Step 4	**ip address** *ip-prefix/length* **Example:** `switch(config-if)# ip address 192.0.2.1/16`	Configures an IP address for this interface. You must do this step after you assign this interface to a VRF.
Step 5	**show vrf** *vrf-name* **interface** *interface-type number* **Example:** `switch(config-vrf)# show vrf Enterprise interface ethernet 1/2`	(Optional) Displays VRF information.
Step 6	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves this configuration change.

This example shows how to add an interface to the VRF:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# vrf member RemoteOfficeVRF
switch(config-if)# ip address 192.0.2.1/16
switch(config-if)# copy running-config startup-config
```

Configuring VRF Parameters for a Routing Protocol

You can associate a routing protocol with one or more VRFs. See the appropriate chapter for information on how to configure VRFs for the routing protocol. This section uses OSPFv2 as an example protocol for the detailed configuration steps.

SUMMARY STEPS

1. **configure terminal**
2. **router ospf** *instance-tag*
3. **vrf** *vrf-name*
4. (Optional) **maximum-paths** *paths*
5. **interface** *interface-type slot/port*
6. **vrf member** *vrf-name*
7. **ip address** *ip-prefix/length*
8. **ip router ospf** *instance-tag* **area** *area-id*
9. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router ospf** *instance-tag*
	
	Example:
	`switch(config-vrf)# router ospf 201`
	`switch(config-router)#`
		Creates a new OSPFv2 instance with the configured instance tag.

Step 3	**vrf** *vrf-name*
	
	Example:
	`switch(config-router)# vrf RemoteOfficeVRF`
	`switch(config-router-vrf)#`
		Enters VRF configuration mode.

Step 4	**maximum-paths** *paths* Example: `switch(config-router-vrf)# maximum-paths 4`	(Optional) Configures the maximum number of equal OSPFv2 paths to a destination in the route table for this VRF. Used for load balancing.

	Command	Purpose

Step 5	**interface** *interface-type slot/port*
	
	Example:
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		Enters interface configuration mode.

Step 6	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member RemoteOfficeVRF`	Adds this interface to a VRF.
Step 7	**ip address** *ip-prefix/length* **Example:** `switch(config-if)# ip address 192.0.2.1/16`	Configures an IP address for this interface. You must do this step after you assign this interface to a VRF.
Step 8	**ip router ospf** *instance-tag* **area** *area-id* **Example:** `switch(config-if)# ip router ospf 201 area 0`	Assigns this interface to the OSPFv2 instance and area configured.
Step 9	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves this configuration change.

This example shows how to create a VRF and add an interface to the VRF:

```
switch# configure terminal
switch(config)# vrf context RemoteOfficeVRF
switch(config-vrf)# exit
switch(config)# router ospf 201
switch(config-router)# vrf RemoteOfficeVRF
switch(config-router-vrf)# maximum-paths 4
switch(config-router-vrf)# interface ethernet 1/2
switch(config-if)# vrf member RemoteOfficeVRF
switch(config-if)# ip address 192.0.2.1/16
switch(config-if)# ip router ospf 201 area 0
switch(config-if)# exit
switch(config)# copy running-config startup-config
```

Configuring Global VRF Route Leaking

You can configure route leaking from the default VRF to a non-default VRF or from a non-default VRF to the default VRF.

Route leaking between non-default VRFs is enabled automatically (with route target matching).

BEFORE YOU BEGIN

Make sure that the Enterprise license is installed and BGP is enabled.

SUMMARY STEPS

1. **configure terminal**
2. **vrf context** [*vrf-name*]
3. **address-family** {**ipv4** | **ipv6**} **unicast**
4. {**import** | **export**} **vrf default** [*prefix-limit*] **map** *route-map*
5. (Optional) **show bgp process vrf** [*vrf-name*]
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	**vrf context** *vrf-name* **Example:** switch(config)# vrf context vpn1 switch(config-vrf)#	Creates a new VRF.
Step 3	**address-family** {**ipv4** | **ipv6**} **unicast** **Example:** switch(config-vrf)# address-family ipv4 unicast switch(config-vrf-af-ipv4)#	Enters global address family configuration mode for the IPv4 or IPv6 address family.
Step 4	{**import** | **export**} **vrf default** [*prefix-limit*] **map** *route-map* Example: switch(config-vrf-af-ipv4)# import vrf default map importmap Example: switch(config-vrf-af-ipv4)# export vrf default map exportmap	Configures VRF route leaking. The following options are available: • **import**—Copies a route containing IPv4 or IPv6 unicast prefixes from the global routing table (the default VRF) into any other VRF. • **export**—Copies a route containing IPv4 or IPv6 unicast prefixes from a non-default VRF into the global routing table (the default VRF). • *prefix-limit*—Specifies the maximum number of routes that can be imported or exported. The range is from 1 to 2147483647, and the default value is 1000.
Step 5	**show bgp process vrf** [*vrf-name*] **Example:** switch(config-vrf-af-ipv4)# show bgp process vrf vpn1	(Optional) Displays the BGP process information for the specified VRF.
Step 6	**copy running-config startup-config** **Example:** switch(config)# copy running-config startup-config	(Optional) Saves this configuration change.

Configuring a VRF-Aware Service

You can configure a VRF-aware service for reachability and filtering. See the "VRF-Aware Services" section on page 13-3 for links to the appropriate chapter or configuration guide for information on how to configure the service for VRFs. This section uses SNMP and IP domain lists as example services for the detailed configuration steps.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *ip-address* [**filter-vrf** *vrf-name*] [**use-vrf** *vrf-name*]
3. **vrf context** [*vrf-name*]
4. **ip domain-list** *domain-name* [**all-vrfs**][**use-vrf** *vrf-name*]
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**snmp-server host** *ip-address* [**filter-vrf** *vrf-name*] [**use-vrf** *vrf-name*]
	
	Example:
	`switch(config)# snmp-server host 192.0.2.1 use-vrf Red`
	`switch(config-vrf)#`
		Configures a global SNMP server and configures the VRF that Cisco NX-OS uses to reach the service. Use the **filter-vrf** keyword to filter information from the selected VRF to this server.

Step 3	**vrf context** *vrf-name*
	
	Example:
	`switch(config)# vrf context Blue`
	`switch(config-vrf)#`
		Creates a new VRF.

Step 4	**ip domain-list** *domain-name* [**all-vrfs**][**use-vrf** *vrf-name*]
	
	Example:
	`switch(config-vrf)# ip domain-list List all-vrfs use-vrf Blue`
	`switch(config-vrf)#`
		Configures the domain list in the VRF and optionally configures the VRF that Cisco NX-OS uses to reach the domain name listed.

Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves this configuration change.

This example shows how to send SNMP information for all VRFs to SNMP host 192.0.2.1, reachable on VRF Red:

```
switch# configure terminal
switch(config)# snmp-server host 192.0.2.1 for-all-vrfs use-vrf Red
switch(config)# copy running-config startup-config
```

This example shows how to filter SNMP information for VRF Blue to SNMP host 192.0.2.12, reachable on VRF Red:

```
switch# configure terminal
switch(config)# vrf context Blue
switch(config-vrf)# snmp-server host 192.0.2.12 use-vrf Red
switch(config)# copy running-config startup-config
```

Setting the VRF Scope

You can set the VRF scope for all EXEC commands (for example, **show** commands). This automatically restricts the scope of the output of EXEC commands to the configured VRF. You can override this scope by using the VRF keywords available for some EXEC commands.

To set the VRF scope, use the following command in EXEC mode:

Command	Purpose

routing-context vrf *vrf-name*
 	
Example:
`switch# routing-context vrf red`
`switch%red#`
	Sets the routing context for all EXEC commands. Default routing context is the default VRF.

To return to the default VRF scope, use the following command in EXEC mode:

Command	Purpose

routing-context vrf default
 	
Example:
`switch%red# routing-context vrf default`
`switch#`
	Sets the default routing context.

Verifying the VRF Configuration

To display VRF configuration information, perform one of the following tasks:

Command	Purpose
show bgp process vrf [*vrf-name*]	Displays the BGP process information for the specified VRF.
show vrf [*vrf-name*]	Displays the information for all or one VRF.

Command	Purpose
show vrf [*vrf-name*] **detail**	Displays detailed information for all or one VRF.
show vrf [*vrf-name*] [**interface** *interface-type slot/port*]	Displays the VRF status for an interface.

Configuration Examples for VRF

This example shows how to configure VRF Red, add an SNMP server to that VRF, and add an instance of OSPF to VRF Red:

```
configure terminal
vrf context Red
 snmp-server host 192.0.2.12 use-vrf Red
router ospf 201
   vrf Red
interface ethernet 1/2
 vrf member Red
 ip address 192.0.2.1/16
 ip router ospf 201 area 0
```

This example shows how to configure VRF Red and Blue, add an instance of OSPF to each VRF, and create an SNMP context for each OSPF instance in each VRF:

```
configure terminal
!Create the VRFs
vrf context Red
vrf context Blue
vrf context Green
!Create the OSPF instances and associate them with a single VRF or multiple VRFs
(recommended)
feature ospf
router ospf Lab
 vrf Red
!
router ospf Production
 vrf Blue
   router-id 1.1.1.1
vrf Green
   router-id 2.2.2.2
!Configure one interface to use ospf Lab on VRF Red
interface ethernet 1/2
 vrf member Red
 ip address 192.0.2.1/16
 ip router ospf Lab area 0
 no shutdown
!Configure another interface to use ospf Production on VRF Blue
interface ethernet 10/2
 vrf member Blue
 ip address 192.0.2.1/16
 ip router ospf Production area 0
 no shutdown
!
interface ethernet 10/3
 vrf member Green
 ip address 192.0.2.1/16
 ip router ospf Production area 0
 no shutdown

!configure the SNMP server
```

```
snmp-server user admin network-admin auth md5 nbv-12345
snmp-server community public ro
!Create the SNMP contexts for each VRF
snmp-server context lab instance Lab vrf Red
snmp-server context production instance Production vrf Blue
!Use the SNMP context lab to access the OSPF-MIB values for the OSPF instance Lab in VRF
Red in this example.
```

Use the SNMP context **lab** to access the OSPF-MIB values for the OSPF instance Lab in VRF Red in this example.

This example shows how to configure route leaking between two non-default VRFs and from the default VRF to a non-default VRF:

```
feature bgp
vrf context Green
ip route 33.33.33.33/32 35.35.1.254
address-family ipv4 unicast
route-target import 3:3
route-target export 2:2
export map test
import map test
import vrf default map test

interface Ethernet1/7
vrf member Green
ip address 35.35.1.2/24

vrf context Shared
ip route 44.44.44.44/32 45.45.1.254
address-family ipv4 unicast
route-target import 1:1
route-target import 2:2
route-target export 3:3
export map test
import map test
import vrf default map test

interface Ethernet1/11
vrf member Shared
ip address 45.45.1.2/24


router bgp 100
address-family ipv4 unicast
redistribute static route-map test
vrf Green
address-family ipv4 unicast
redistribute static route-map test
vrf Shared
address-family ipv4 unicast
redistribute static route-map test

ip prefix-list test seq 5 permit 0.0.0.0/0 le 32

route-map test permit 10
match ip address prefix-list test

ip route 100.100.100.100/32 55.55.55.1

switch# show ip route vrf all
IP Route Table for VRF "default"
```

```
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

55.55.55.0/24, ubest/mbest: 1/0, attached
*via 55.55.55.5, Lo0, [0/0], 00:07:59, direct
55.55.55.5/32, ubest/mbest: 1/0, attached
*via 55.55.55.5, Lo0, [0/0], 00:07:59, local
100.100.100.100/32, ubest/mbest: 1/0
*via 55.55.55.1, [1/0], 00:07:42, static

IP Route Table for VRF "management"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

0.0.0.0/0, ubest/mbest: 1/0
*via 10.29.176.1, [1/0], 12:53:54, static
10.29.176.0/24, ubest/mbest: 1/0, attached
*via 10.29.176.233, mgmt0, [0/0], 13:11:57, direct
10.29.176.233/32, ubest/mbest: 1/0, attached
*via 10.29.176.233, mgmt0, [0/0], 13:11:57, local

IP Route Table for VRF "Green"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

33.33.33.33/32, ubest/mbest: 1/0
*via 35.35.1.254, [1/0], 00:23:44, static
35.35.1.0/24, ubest/mbest: 1/0, attached
*via 35.35.1.2, Eth1/7, [0/0], 00:26:46, direct
35.35.1.2/32, ubest/mbest: 1/0, attached
*via 35.35.1.2, Eth1/7, [0/0], 00:26:46, local
44.44.44.44/32, ubest/mbest: 1/0
*via 45.45.1.254%Shared, [20/0], 00:12:08, bgp-100, external, tag 100
100.100.100.100/32, ubest/mbest: 1/0
*via 55.55.55.1%default, [20/0], 00:07:41, bgp-100, external, tag 100

IP Route Table for VRF "Shared"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

33.33.33.33/32, ubest/mbest: 1/0
*via 35.35.1.254%Green, [20/0], 00:12:34, bgp-100, external, tag 100
44.44.44.44/32, ubest/mbest: 1/0
*via 45.45.1.254, [1/0], 00:23:16, static
45.45.1.0/24, ubest/mbest: 1/0, attached
*via 45.45.1.2, Eth1/11, [0/0], 00:25:53, direct
45.45.1.2/32, ubest/mbest: 1/0, attached
*via 45.45.1.2, Eth1/11, [0/0], 00:25:53, local
100.100.100.100/32, ubest/mbest: 1/0
*via 55.55.55.1%default, [20/0], 00:07:41, bgp-100, external, tag 100
```

This example shows how to configure route leaking from a non-default VRF to the default VRF:

```
feature bgp
vrf context vpn1
    address-family ipv4 unicast
        import vrf default map importmap
        export vrf default map exportmap

show bgp ipv4 unicast 123.123.123.123/32
BGP routing table information for VRF default, address family IPv4 Unicast
BGP routing table entry for 123.123.123.123/32, version 6
Paths: (1 available, best #1)
Flags: (0x8008001a) on xmit-list, is in urib, is best urib route

  Advertised path-id 1
  Path type: redist, path is valid, is best path
             Imported from 100:1:123.123.123.123/32 (VRF vpn1)
  AS-Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (1.1.1.1)
      Origin incomplete, MED 0, localpref 100, weight 32768
Extcommunity:  RT:100:1

  Path-id 1 not advertised to any peer
  Path-id 1 scheduled to be advertised to peers:
    2.2.2.2

show bgp process vrf vpn1
Information regarding configured VRFs:

BGP Information for VRF vpn1
VRF Id                          : 3
VRF state                       : UP
Router-ID                       : 20.0.0.1
Configured Router-ID            : 0.0.0.0
Confed-ID                       : 0
Cluster-ID                      : 0.0.0.0
No. of configured peers         : 2
No. of pending config peers     : 0
No. of established peers        : 2
VRF RD                          : 100:1

    Information for address family IPv4 Unicast in VRF vpn1
    Table Id                    : 3
    Table state                 : UP
Peers      Active-peers    Routes     Paths      Networks   Aggregates
    1          1               6          6          0          0

    Redistribution
        None

    Export RT list:
        100:1
        1000:1
    Import RT list:
        100:1
    Label mode: per-prefix
    Aggregate label: 492287
    Import default limit       : 1000
    Import default prefix count : 2
    Import default map         : importmap
    Export default limit       : 1000
    Export default prefix count : 3
    Export default map         : exportmap
```

Additional References

For additional information related to implementing virtualization, see the following sections:

Related Documents

Related Topic	Document Title
BGP	*Chapter 9, "Configuring Basic BGP"*
VRFs	*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide* *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*

Standards

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