

Configure VRF Route Leak on Nexus Switches

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Limitations](#)

[Configure](#)

[Default VRF to VRF](#)

[VRF to VRF](#)

[VRF to Default VRF](#)

[Verify](#)

Introduction

This document describes how to configure Route Leak on Cisco Nexus NX-OS based Switches.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Nexus NX-OS Software.
- Routing Protocols like Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and the like.

Components Used

The information in this document is based on Cisco Nexus 7000 with NXOS version 7.3(0)D1(1)

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Limitations

You must leak a route directly from the source VRF into the target VRF. You cannot leak a route that is currently leaked from another VRF.

Consider that a BGP session from the Nexus cannot be established to a peer IP when routed through a different VRF on the Nexus.

Configure

The leak between VRFs is performed at BGP process level. Because of this, it is necessary to add the routes to the BGP process first, specifically in the BGP table.

Note: The terms Default VRF and Global Routing Table are used interchangeably in this document.

Default VRF to VRF

In this case, Nexus has received two routes in its Default VRF via EIGRP. The configuration leaks the routes in VRF BLUE.

For the purpose of this example, only the route 192.168.2.0/24 is leaked.

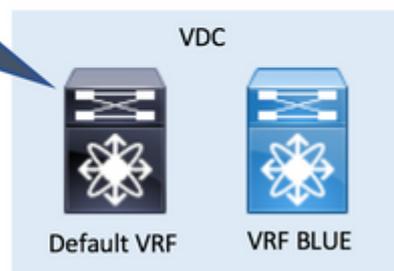
Global Routing Table output

```
Nexus# show ip route eigrp
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>

172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
Nexus#
```

Default VRF Routing Table

```
Nexus# show ip route eigrp
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
```



Note: In NX-OS, enable the features in Global Configuration Mode is required. In order to enable BGP, the command is **feature bgp**.

- Step 1. Redistribute into BGP.

Redistribute the routes which exist in the Default VRF Routing Table in BGP.

Since the routes are in the Default VRF, the **redistribute** command in BGP goes under the global **address-family ipv4 unicast** section.

Use the correct parameter for the **redistribute** command, this depends on how the routes are in the Default VRF (directly connected, eigrp, ospf,...).

Note: You can skip Step 1 in all scenarios if the routes to be leaked are installed as BGP routes in the origin VRF. In this example the origin VRF is the Default VRF (Global Routing Table).

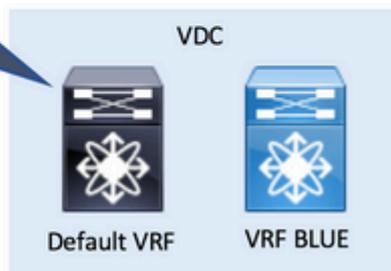
Redistribute into BGP

```
route-map ALL permit 10
!
router bgp 65535
  address-family ipv4 unicast
    redistribute eigrp 1 route-map ALL
```

Default VRF BGP Table

```
Nexus# show bgp ipv4 unicast
BGP routing table information for VRF default, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100         32768 ?
*>r192.168.2.0/24 0.0.0.0      130816      100         32768 ?
```

```
Nexus# show ip route eigrp
172.16.2.2/32, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
```



Note: In NX-OS a **route-map** is always required as a parameter in order to selectively redistribute routes.

An empty **route-map permit** statement created is valid in order to match any and all the routes.

- Step 2. Configure Import VRF default in the destination VRF.

The **import vrf default** command is configured in the destination VRF. The command line requires a **route-map** as a parameter in order to explicitly define the routes to be imported in the destination VRF which in this case is the VRF named BLUE.

Configure Import VRF default in the destination VRF

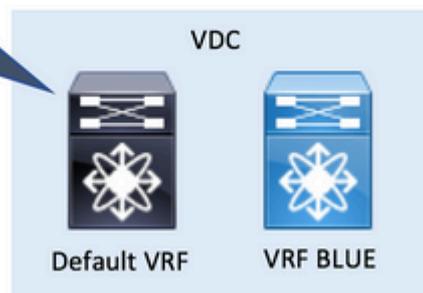
```
ip prefix-list NETWORK seq 5 permit 192.168.2.0/24
!
route-map GLOBAL-TO-VRF permit 10
  match ip address prefix-list NETWORK
!
vrf context BLUE
  address-family ipv4 unicast
    import vrf default map GLOBAL-TO-VRF
```

```
Nexus# show ip bgp
BGP routing table information for VRF default, address family IPv4 Unicast
Network          Next Hop        Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0         130816      100         32768 ?
*>r192.168.2.0/24 0.0.0.0         130816      100         32768 ?
```



```
VRF BLUE
Nexus# show bgp ipv4 unicast vr
BGP routing table information f
Network          Next Hop
*>r192.168.2.0/24 0.0.0.0
```

```
Nexus# show ip route eigrp
172.16.2.2/32, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
```



- Step 3. Check the destination VRF Routing Table.

You can confirm in the destination VRF that the routes are now seen via BGP.

These BGP routes in the VRF can now be redistributed in any other Routing Protocol that runs in the same VRF.

Check the destination VRF Routing Table

```
Nexus# show ip route vrf BLUE
```

```

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

192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2%default, Eth2/1, [20/130816], 00:15:00, bgp-65535, external, tag 65535,
Nexus#

```

```

Nexus# show ip bgp
BGP routing table information for VRF default, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100         32768 ?
*>r192.168.2.0/24  0.0.0.0      130816      100         32768 ?

```

```

Nexus# show bgp ipv4 unicast vrf BLUE
BGP routing table information for VRF BLUE, address
Network      Next Hop      Metric      LocPrf      Weight Path
*>r192.168.2.0/24  0.0.0.0      130816

```

```

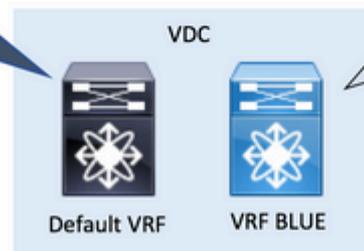
Nexus# show ip route eigrp
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:21, eigrp-1, internal

```

```

Nexus# show ip route vrf BLUE
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2%default, Eth2/1, [20/130816]

```



VRF to VRF

In this case, Nexus has received two routes in its VRF called RED via EIGRP. The configuration leaks the routes in VRF BLUE.

VRF RED Routing Table output

```

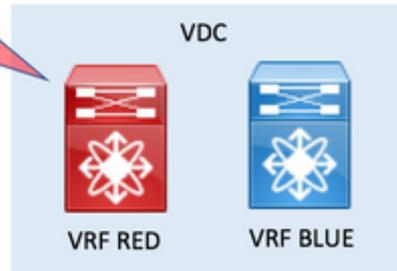
Nexus# show ip route eigrp vrf RED
IP Route Table for VRF "RED"
'*' denotes best ucast next-hop
 '**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
Nexus#

```

VRF RED Routing Table

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```



- Step 1. Redistribute into BGP.

Redistribute the routes which exist in VRF RED Routing Table in BGP.

Since the routes are in the VRF RED, the **redistribute** command in BGP goes under the vrf RED **address-family ipv4 unicast** section.

Redistribute into BGP

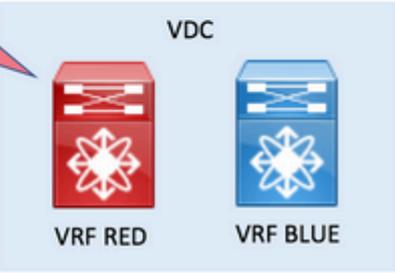
```
route-map ALL permit 10
!
router bgp 65535
  vrf RED
    address-family ipv4 unicast
      redistribute eigrp 1 route-map ALL
```

VRF RED BGP Table

```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network          Next Hop        Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0         130816      100         32768 ?
*>r192.168.2.0/24 0.0.0.0         130816      100         32768 ?
```



```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```



- Step 2. Create export & import Route-Targets.

In order to leak between VRFs, the use of Route-Targets is required.

The origin VRF **exports** a Route-Target value.

The destination VRF **imports** the same Route-Target value.

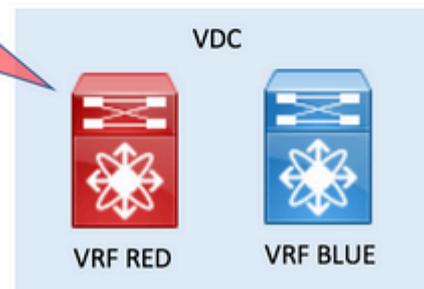
Create export & import Route-Targets

```
vrf context RED
  address-family ipv4 unicast
  route-target export 1:1
!
vrf context BLUE
  address-family ipv4 unicast
  route-target import 1:1
```

```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network          Next Hop          Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0           130816      100         32768 ?
*>r192.168.2.0/24 0.0.0.0           130816      100         32768 ?
```

```
Nexus# show bgp ipv4 unicast v
BGP routing table information f
Network          Next Hop
*>r172.16.2.2/32  0.0.0.0
*>r192.168.2.0/24 0.0.0.0
```

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```



- Step 3. Check the destination VRF Routing Table.

You can confirm in the destination VRF that the routes are now seen via BGP.

These BGP routes in the VRF can now be redistributed in any other Routing Protocol that runs in the same VRF.

Check the destination VRF Routing Table

```
Nexus# show ip route vrf BLUE
IP Route Table for VRF "BLUE"
 '*' denotes best ucast next-hop
 '**' denotes best mcast next-hop
 '[x/y]' denotes [preference/metric]
 '%<string>' in via output denotes VRF <string>

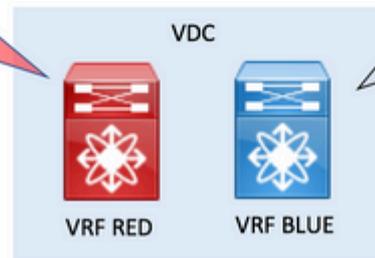
172.16.2.2/32, ubest/mbest: 1/0
 *via 10.1.2.2%RED, Eth2/1, [20/130816], 00:01:58, bgp-65535, external, tag 65535,
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2%RED, Eth2/1, [20/130816], 00:01:58, bgp-65535, external, tag 65535,
Nexus#
```

```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100        32768 ?
*>r192.168.2.0/24  0.0.0.0      130816      100        32768 ?
```

```
Nexus# show bgp ipv4 unicast vrf BLUE
BGP routing table information for VRF BLUE, address family IPv4 Unicast
Network      Next Hop      Metric
*>r172.16.2.2/32  0.0.0.0      130816
*>r192.168.2.0/24  0.0.0.0      130816
```

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```

```
Nexus# show ip route vrf BLUE
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2%RED, Eth2/1, [20/130816]
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2%RED, Eth2/1, [20/130816]
```



- Step 4 (Optional). Assign Route-Target to specific routes.

You can optionally use the **export map** command under the origin VRF in order to assign Route-Targets to specific routes to be exported.

Use the **set extcommunity rt** parameter in the route-map in order to assign the Route-Target.

In this example, only network 192.168.2.0/24 is exported with Route-Target 1:1 which is later imported in VRF BLUE.

The result is that only the specified network is leaked.

Assign Route-Target to specific routes

```
ip prefix-list NETWORK seq 5 permit 192.168.2.0/24
!
route-map ADD-RT permit 10
  match ip address prefix-list NETWORK
  set extcommunity rt 1:1
!
vrf context RED
  address-family ipv4 unicast
    export map ADD-RT
!
vrf context BLUE
  address-family ipv4 unicast
    route-target import 1:1
```

VRF to Default VRF

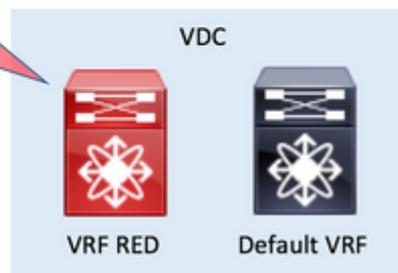
Note: On Nexus 7000 and Nexus 7700 family of Switches, this feature that supports the export of IP prefixes to the global routing table (the default VRF) from any other VRF with the use of the **export vrf default map** command was introduced in NX-OS release 7.3(0)D1(1)

Nexus has received two routes in its VRF called RED via EIGRP. The configuration leaks the routes in the Default VRF.

For the purpose of this example, only the route 192.168.2.0/24 is leaked.

VRF RED Routing Table

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```



- Step 1. Redistribute into BGP.

Redistribute the routes which exist in VRF RED Routing Table in BGP.

Since the routes are in the VRF RED, the **redistribute** command in BGP goes under the vrf RED **address-family ipv4 unicast** section.

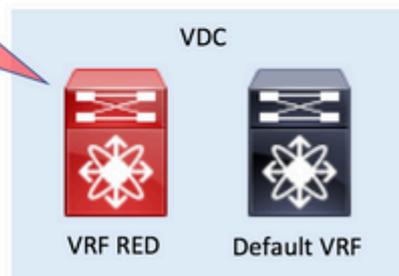
Redistribute into BGP

```
route-map ALL permit 10
!
router bgp 65535
  vrf RED
    address-family ipv4 unicast
      redistribute eigrp 1 route-map ALL
```

VRF RED BGP Table

```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network          Next Hop        Metric    LocPrf    Weight Path
*>r172.16.2.2/32  0.0.0.0         130816   100      32768 ?
*>r192.168.2.0/24 0.0.0.0         130816   100      32768 ?
```

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```



- Step 2. Configure Export VRF default in the origin VRF.

The **export vrf default** command is configured in the origin VRF. The command line requires a **route-map** as a parameter in order to explicitly define the routes to be exported in the Default VRF.

Configure Export VRF default in the origin VRF

```
ip prefix-list NETWORK seq 5 permit 192.168.2.0/24
!
route-map GLOBAL-TO-VRF permit 10
 match ip address prefix-list NETWORK
!
vrf context RED
 address-family ipv4 unicast
  export vrf default map GLOBAL-TO-VRF
```

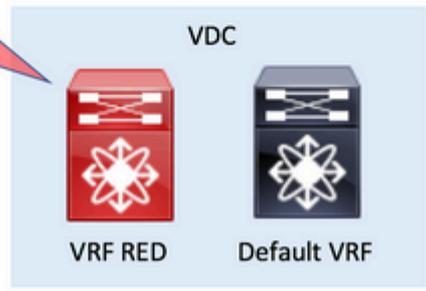
```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100         32768 ?
*>r192.168.2.0/24 0.0.0.0      130816      100         32768 ?
```



Default VRF

```
Nexus# show bgp ipv4 unicast
BGP routing table information for
Network      Next Hop
*>r192.168.2.0/24 0.0.0.0
```

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```



- Step 3. Check the Default VRF Routing Table.

You can confirm in the Default VRF that the routes are now seen via BGP.

These BGP routes in the Default VRF can now be redistributed in any other Routing Protocol that also runs in the Default VRF.

Check the Default VRF Routing Table

```
Nexus# show ip route
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>

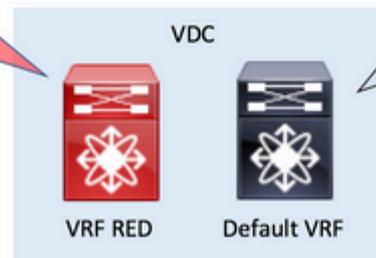
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2%RED, Eth2/1, [20/130816], 00:08:19, bgp-65535, external, tag 65535,
Nexus#
```

```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100        32768 ?
*>r192.168.2.0/24 0.0.0.0      130816      100        32768 ?
```

```
Nexus# show bgp ipv4 unicast
BGP routing table information for VRF default,
Network      Next Hop      Metric
*>r192.168.2.0/24 0.0.0.0      130816
```

```
Nexus# show ip route eigrp vrf RED
172.16.2.2/32, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2, Eth2/1, [90/130816], 00:00:08, eigrp-1, internal
```

```
Nexus# show ip route
192.168.2.0/24, ubest/mbest: 1/0
  *via 10.1.2.2%RED, Eth2/1, [20/130816]
```



Verify

There are 4 phases in the vrf route leak process. Verification can be done in order:



In order to check the routes are correctly in the Routing Table, the command is:

```
show ip route [vrf <vrf name>]
```

In order to check the routes are correctly in the BGP table, the commands are:

Notice that the second command can be used interchangeably in order to show IPv4 Unicast addresses in the BGP table.

```
show bgp ipv4 unicast [vrf <vrf name>]
```

```
show ip bgp [vrf <vrf name>]
```

Lastly, the **show forwarding route A.B.C.D/LEN [VRF <vrf name>]** can be used in order to confirm the Layer 3 - route programmed at Line Card level (Hardware Programming)

Nexus# show forwarding route 10.1.2.2

slot 1
=====

IPv4 routes for table default/base

'*' denotes recursive route

Prefix	Next-hop	Interface	Labels
10.1.2.0/24	Attached	Ethernet2/1	

Nexus#