

# Configuration de la fuite de route VRF sur les commutateurs Nexus

## Table des matières

---

[Introduction](#)

[Conditions préalables](#)

[Exigences](#)

[Composants utilisés](#)

[Limites](#)

[Configurer](#)

[VRF par défaut vers VRF](#)

[VRF vers VRF](#)

[VRF vers VRF par défaut](#)

[Vérifier](#)

---

## Introduction

Ce document décrit comment configurer la fuite de route sur les commutateurs basés sur Cisco Nexus NX-OS.

## Conditions préalables

### Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Logiciel Nexus NX-OS.
- Protocoles de routage tels que EIGRP (Enhanced Interior Gateway Routing Protocol), OSPF (Open Shortest Path First), BGP (Border Gateway Protocol), etc.

### Composants utilisés

Les informations contenues dans ce document sont basées sur Cisco Nexus 7000 avec 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. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

### Limites

Vous devez faire passer une route directement du VRF source dans le VRF cible. Vous ne pouvez

pas laisser fuir une route qui est actuellement divulguée à partir d'un autre VRF.

Considérez qu'une session BGP du Nexus ne peut pas être établie vers une adresse IP homologue lorsqu'elle est routée via un VRF différent sur le Nexus.

## Configurer

La fuite entre les VRF est effectuée au niveau du processus BGP. Pour cette raison, il est nécessaire d'ajouter d'abord les routes au processus BGP, en particulier dans la table BGP.

---

 Remarque : les termes VRF par défaut et Table de routage globale sont utilisés de manière interchangeable dans ce document.

---

### VRF par défaut vers VRF

Dans ce cas, Nexus a reçu deux routes dans son VRF par défaut via EIGRP. La configuration laisse passer les routes en VRF BLUE.

Dans le cadre de cet exemple, seule la route 192.168.2.0/24 est divulguée.

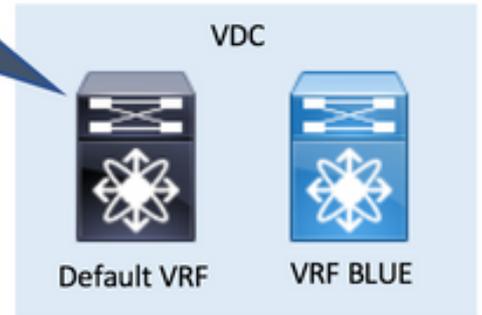
#### Sortie de la table de routage globale

```
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
```



 Remarque : dans NX-OS, vous devez activer les fonctionnalités en mode de configuration globale. Afin d'activer BGP, la commande est `feature bgp`.

- Étape 1. Redistribuez dans BGP.

Redistribuez les routes qui existent dans la table de routage VRF par défaut dans BGP.

Puisque les routes sont dans le VRF par défaut, la commande redistribuée dans BGP passe sous la section globale `address-family ipv4 unicast`.

Utilisez le paramètre correct pour la commande redistribuée, cela dépend de la façon dont les routes sont dans le VRF par défaut (directement connecté, `eigrp`, `ospf`,...).

 Remarque : vous pouvez ignorer l'étape 1 dans tous les scénarios si les routes à fuiter sont installées en tant que routes BGP dans le VRF d'origine. Dans cet exemple, le VRF d'origine est le VRF par défaut (table de routage globale).

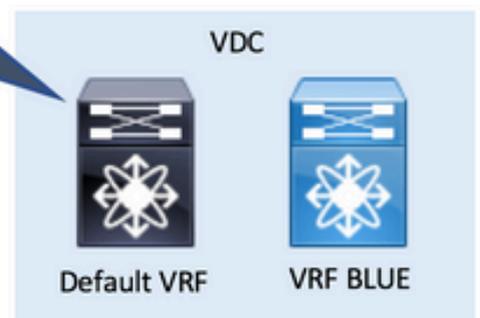
### Redistribuer dans BGP

```
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
```



 Remarque : dans NX-OS, une route-map est toujours requise comme paramètre afin de redistribuer sélectivement les routes.

Une instruction route-map permit vide créée est valide afin de correspondre à toutes les routes.

- Étape 2. Configurez Import VRF par défaut dans le VRF de destination.

La commande import vrf default est configurée dans le VRF de destination. La ligne de commande requiert une route-map comme paramètre afin de définir explicitement les routes à importer dans le VRF de destination qui dans ce cas est le VRF nommé BLUE.

### Configurer le VRF d'importation par défaut dans le VRF de destination

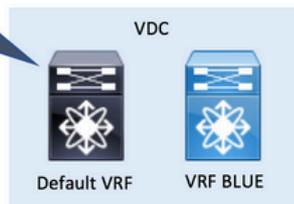
```
ip prefix-list NETWORK seq 5 permit 192.168.2.0/24
!
route-map GLOBAL-TO-VRF permit 10
  match ip address prefix-list NETWORK
!
vrf context BLUE
  address-family ipv4 unicast
    import vrf default map GLOBAL-TO-VRF
```

```
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 BGP Table

```
Nexus# show bgp ipv4 unicast vrf BLUE
BGP routing table information for VRF BLUE, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>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
```



- Étape 3. Vérifiez la table de routage VRF de destination.

Vous pouvez confirmer dans le VRF de destination que les routes sont maintenant visibles via BGP.

Ces routes BGP dans le VRF peuvent maintenant être redistribuées dans tout autre protocole de routage qui s'exécute dans le même VRF.

### Vérifier la table de routage VRF de destination

```
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#
```

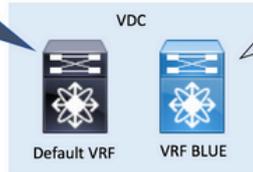
```
Nexus# show ip bgp
BGP routing table information for VRF default, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100        32768 ?
*>r192.168.2.0/24  0.0.0.0      130816      100        32768 ?
```

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

VRF BLUE 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
```

```
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], 00:15:00, bgp-65535, external, tag 65535,
```



## VRF vers VRF

Dans ce cas, Nexus a reçu deux routes dans son VRF appelé RED via EIGRP. La configuration laisse passer les routes en VRF BLUE.

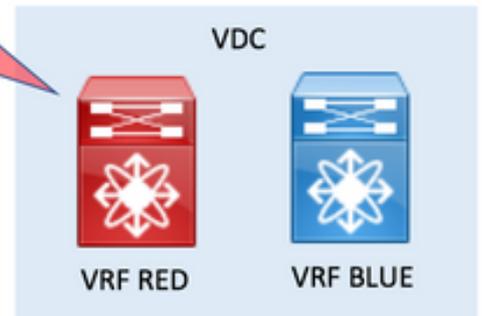
### Sortie de la table de routage ROUGE VRF

```
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
```



- Étape 1. Redistribuez dans BGP.

Redistribuez les routes qui existent dans la table de routage VRF RED dans BGP.

Puisque les routes sont dans le VRF RED, la commande redistribuée dans BGP passe sous la section vrf RED address-family ipv4 unicast.

### Redistribuer dans BGP

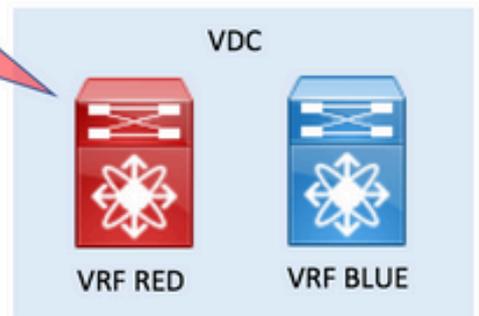
```
route-map ALL permit 10
!
router bgp 65535
  vrf RED
    address-family ipv4 unicast
      redistribute eigrp 1 route-map ALL
```

## 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
```



- Étape 2. Créez des cibles de routage d'exportation et d'importation.

Afin d'éviter les fuites entre les VRF, l'utilisation de Route-Targets est requise.

Le VRF d'origine exporte une valeur Route-Target.

Le VRF de destination importe la même valeur Route-Target.

### Créer des cibles de routage d'exportation et d'importation

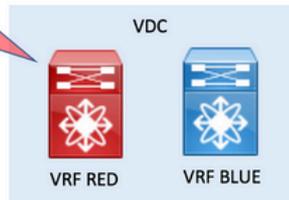
```
vrf context RED
  address-family ipv4 unicast
    route-target export 1:1
!
vrf context BLUE
  address-family ipv4 unicast
    route-target import 1:1
```

```
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 ?
```

#### VRF BLUE BGP Table

```
Nexus# show bgp ipv4 unicast vrf BLUE
BGP routing table information for VRF BLUE, 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
```



- Étape 3. Vérifiez la table de routage VRF de destination.

Vous pouvez confirmer dans le VRF de destination que les routes sont maintenant visibles via BGP.

Ces routes BGP dans le VRF peuvent maintenant être redistribuées dans tout autre protocole de routage qui s'exécute dans le même VRF.

#### Vérifier la table de routage VRF de destination

```
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#
```

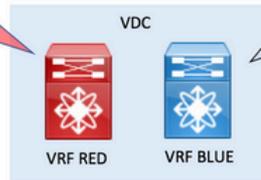
```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network      Next Hop      Metric    LocPrf    Weight Path
*>r172.16.2.2/32  0.0.0.0      130816    100      32768 ?
*>r192.168.2.0/24  0.0.0.0      130816    100      32768 ?
```

```
Nexus# show bgp ipv4 unicast vrf BLUE
BGP routing table information for VRF BLUE, address family IPv4 Unicast
Network      Next Hop      Metric    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 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
```

```
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], 00:04:39, bgp-65535, external, tag 65535,
192.168.2.0/24, ubest/mbest: 1/0
 *via 10.1.2.2%RED, Eth2/1, [20/130816], 00:04:39, bgp-65535, external, tag 65535,
```



- Étape 4 (facultatif). Attribuez la cible de routage à des routes spécifiques.

Vous pouvez éventuellement utiliser la commande export map sous le VRF d'origine afin d'attribuer des cibles de route à des routes spécifiques à exporter.

Utilisez le paramètre set extcommunity rt dans la route-map afin d'attribuer la route-Target.

Dans cet exemple, seul le réseau 192.168.2.0/24 est exporté avec Route-Target 1:1 qui est importé ultérieurement en VRF BLUE.

Il en résulte que seul le réseau spécifié est sujet à des fuites.

#### Attribuer une route-cible à des routes spécifiques

```
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 vers VRF par défaut

 Remarque : sur les gammes de commutateurs Nexus 7000 et Nexus 7700, cette fonctionnalité qui prend en charge l'exportation des préfixes IP vers la table de routage

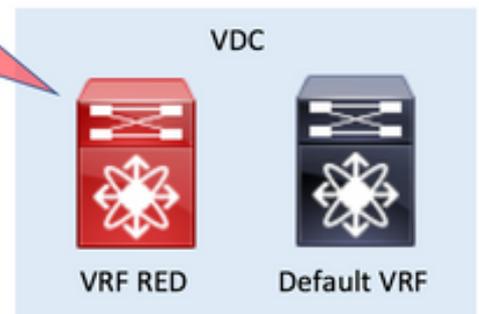
 globale (VRF par défaut) à partir de tout autre VRF à l'aide de la commande `export vrf default map` a été introduite dans NX-OS version 7.3(0)D1(1)

Nexus a reçu deux routes dans son VRF appelé RED via EIGRP. La configuration laisse échapper les routes dans le VRF par défaut.

Dans le cadre de cet exemple, seule la route 192.168.2.0/24 est divulguée.

### 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
```



- Étape 1. Redistribuez dans BGP.

Redistribuez les routes qui existent dans la table de routage VRF RED dans BGP.

Puisque les routes sont dans le VRF RED, la commande redistribuée dans BGP passe sous la section `vrf RED address-family ipv4 unicast`.

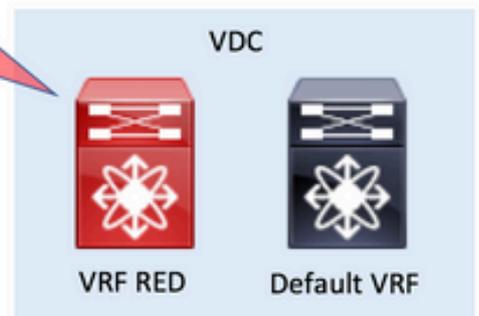
### Redistribuer dans BGP

```
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
```



- Étape 2. Configurez Export VRF par défaut dans le VRF d'origine.

La commande `export vrf default` est configurée dans le VRF d'origine. La ligne de commande requiert une route-map comme paramètre afin de définir explicitement les routes à exporter dans le VRF par défaut.

### Configurez Export VRF par défaut dans le VRF d'origine

```
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
```

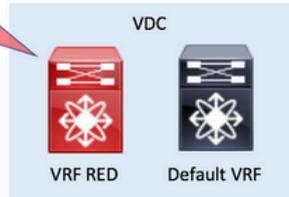
```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100        32768 ?
*>r192.168.2.0/24  0.0.0.0      130816      100        32768 ?
```



#### Default VRF 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
*>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
```



- Étape 3. Vérifiez la table de routage VRF par défaut.

Vous pouvez confirmer dans le VRF par défaut que les routes sont maintenant vues via BGP.

Ces routes BGP dans le VRF par défaut peuvent maintenant être redistribuées dans tout autre protocole de routage qui s'exécute également dans le VRF par défaut.

#### Vérification de la table de routage VRF par défaut

```
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#
```

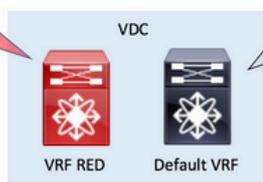
```
Nexus# show bgp ipv4 unicast vrf RED
BGP routing table information for VRF RED, address family IPv4 Unicast
Network      Next Hop      Metric      LocPrf      Weight Path
*>r172.16.2.2/32  0.0.0.0      130816      100        32768 ?
*>r192.168.2.0/24  0.0.0.0      130816      100        32768 ?
```

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

Default VRF 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
```

```
Nexus# show ip route
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,
```



## Vérifier

Il y a 4 phases dans le processus de fuite de route VRF. La vérification peut être effectuée dans l'ordre suivant :



Afin de vérifier que les routes sont correctement dans la table de routage, la commande est :

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

Afin de vérifier que les routes sont correctement dans la table BGP, les commandes sont :

Notez que la deuxième commande peut être utilisée de manière interchangeable afin d'afficher les adresses de monodiffusion IPv4 dans la table BGP.

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

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

Enfin, la commande `show forwarding route A.B.C.D/LEN [VRF <vrf name>]` peut être utilisée afin de confirmer la couche 3 - route programmée au niveau de la carte de ligne (programmation matérielle)

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#

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.