

# Configure VRF Leaks on IOS XE

## Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Scenario 1 - VRF Route Leak between BGP and IGP \( EIGRP \)](#)

[Network Diagram](#)

[Configure](#)

[Verify](#)

[Scenario 2 - VRF Leaking between VRF A and VRF B](#)

[Network Diagram](#)

[Configure](#)

[Verify](#)

[Scenario 3 - VRF Leaking between OSPF \( VRF \) and EIGRP \( Global \) with BGP \( Optional \)](#)

[Network Diagram](#)

[Configure](#)

[Verify](#)

[Additional Resources](#)

## Introduction

This document describes and provides example configurations for common methods of Virtual Routing and Forwarding (VRF) route leaking.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of these topics:

- Border Gateway Protocol (BGP)
- Routing Protocol Redistribution
- VRF
- Cisco IOS® XE Software

For more information on these topics, see:

[Redistribute Routing Protocols](#)

[Mutual Redistribution Between EIGRP and BGP Configuration Example](#)

[Understand Redistribution of OSPF Routes into BGP](#)

## Components Used

The information in this document is based on Routers with Cisco IOS® XE versions 16.12.X and 17.X

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.

## Background Information

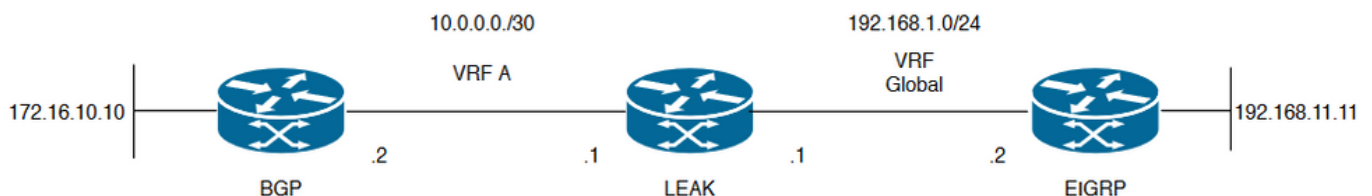
VRF allows a router to maintain separate routing tables for different virtual networks. When exceptions are needed, VRF route leaking allows some traffic to be routed between the VRFs without the use of static routes.

## Scenario 1 - VRF Route Leak between BGP and IGP ( EIGRP )

Scenario 1 provides an example of VRF route leaking between BGP and EIGRP. This method can be used for other IGPs.

### Network Diagram

The Network Diagram as seen in Image 1 shows the layer 3 topology where route leaking is needed.



**Image 1.** Route Leaking Topology for Scenario 1

Router "LEAK" has a BGP neighborhood to a neighbor in VRF A, and an EIGRP neighbor in the global VRF. Device 192.168.11.11 needs to be able to connect to device 172.16.10.10 across the network.

Router LEAK cannot route between the two since the routes are in different VRFs. These routing tables show the current routes per VRF, and indicate which routes need to be leaked between the global VRF and VRF A.

LEAK routing tables:

## EIGRP Routing Table ( Global Routing )

LEAK#show ip route

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, m - OMP  
n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA  
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  
H - NHRP, G - NHRP registered, g - NHRP registration summary  
o - ODR, P - periodic downloaded static route, 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, GigabitEthernet2  
L 192.168.1.1/32 is directly connected, GigabitEthernet2  
192.168.11.0/32 is subnetted, 1 subnets  
**D 192.168.11.11 [90/130816] via 192.168.1.2, 02:30:29, GigabitEthernet2** >> Route to be exchange to the VRF A routing table.

### VRF A Routing Table

LEAK#show ip route vrf A

Routing Table: A

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, m - OMP  
n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA  
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  
H - NHRP, G - NHRP registered, g - NHRP registration summary  
o - ODR, P - periodic downloaded static route, l - LISP  
a - application route  
+ - replicated route, % - next hop override, p - overrides from PFR

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks  
C 10.0.0.0/30 is directly connected, GigabitEthernet1  
L 10.0.0.1/32 is directly connected, GigabitEthernet1  
172.16.0.0/32 is subnetted, 1 subnets  
**B 172.16.10.10 [200/0] via 10.0.0.2, 01:47:58** >> Route to be exchange to the global routing table.

## Configure

Please do the procedures to create the leak between the two routing tables:

### Step 1.

Create route-maps to filter the routes to be injected in both routing tables.

```
LEAK(config)#Route-map VRF_TO_EIGRP
LEAK(config-route-map)#match ip address prefix-list VRF_TO_EIGRP
LEAK(config-route-map)#exit
!
```

Prefix-list created to match the host that is attached to the previous route-map configured.

```
!  
ip prefix-list VRF_TO_EIGRP permit 172.16.10.10/32
```

or

```
LEAK(config)#Route-map VRF_TO_EIGRP  
LEAK(config-route-map)# match ip address 10  
LEAK(config-route-map)#exit
```

ACL created to match the host that is attached to the previous route-map.

```
!  
LEAK#show ip access-lists 10  
10 permit 172.16.10.10
```

```
LEAK(config)#Route-map EIGRP_TO_VRF  
LEAK(config-route-map)#match ip address prefix-list EIGRP_TO_VRF  
LEAK(config-route-map)#exit  
LEAK(config)#
```

Prefix-list created to match the host that is attached to the previous route-map configured.

```
!  
ip prefix-list EIGRP_TO_VRF permit 192.168.11.11/32
```

or

```
LEAK(config)#Route-map EIGRP_TO_VRF  
LEAK(config-route-map)#match ip address 20  
LEAK(config-route-map)#exit  
LEAK(config)#
```

ACL created to match the host that is attached to the previous route-map.

```
!  
LEAK#show ip access-list 20  
10 permit 192.168.11.11
```

## Step 2.

Define the import/export maps and add the route-map names.

```
LEAK(config)#vrf definition A  
LEAK(config-vrf)#address-family ipv4  
LEAK(config-vrf-af)#import ipv4 unicast map EIGRP_TO_VRF >> Import the global routing table  
routes at the VRF routing table.  
LEAK(config-vrf-af)#export ipv4 unicast map VRF_TO_EIGRP >> Export the VRF routes to the Global  
Routing Table.  
LEAK(config-vrf-af)#end
```

## Step 3.

Proceed with the dual redistribution.

Redistribute EIGRP

```
LEAK(config)#router bgp 1  
LEAK(config-router)#redistribute eigrp 1  
LEAK(config-router)#end
```

Redistribution BGP

```
LEAK(config)#router eigrp 1  
LEAK(config-router)#redistribute bgp 1 metric 100 1 255 1 1500  
LEAK(config-router)#end
```

# Verify

## Routing table from VRF A

LEAK#show ip route vrf A

Routing Table: A

< Snip for resume >

```
10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 10.0.0.0/30 is directly connected, GigabitEthernet1
L 10.0.0.1/32 is directly connected, GigabitEthernet1
172.16.0.0/32 is subnetted, 1 subnets
B 172.16.10.10 [200/0] via 10.0.0.2, 00:58:53
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
B 192.168.1.0/24 is directly connected, 00:01:00, GigabitEthernet2
L 192.168.1.1/32 is directly connected, GigabitEthernet2
192.168.11.0/32 is subnetted, 1 subnets
B 192.168.11.11 [20/130816] via 192.168.1.2, 00:01:00, GigabitEthernet2 >> Route from global routing table at VRF A routing table.
```

## Global Routing Table (EIGRP)

LEAK#show ip route

< snip for resume >

Gateway of last resort is not set

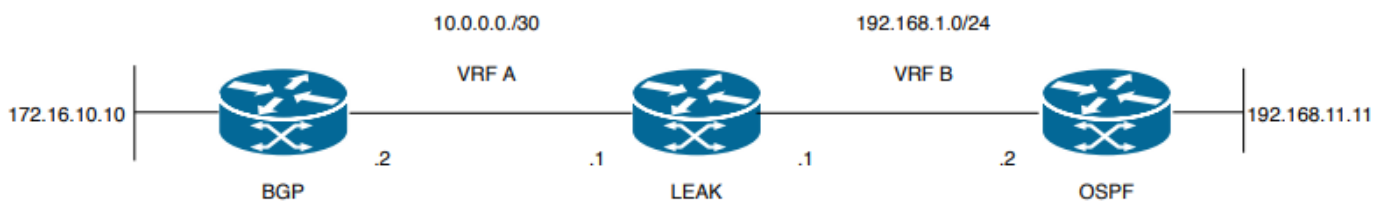
```
172.16.0.0/32 is subnetted, 1 subnets
B 172.16.10.10 [200/0] via 10.0.0.2 (A), 00:04:47 >> Route from VRF A at global routing table.
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.1.0/24 is directly connected, GigabitEthernet2
L 192.168.1.1/32 is directly connected, GigabitEthernet2
192.168.11.0/32 is subnetted, 1 subnets
D 192.168.11.11 [90/130816] via 192.168.1.2, 01:03:35, GigabitEthernet2
LEAK#
```

# Scenario 2 - VRF Leaking between VRF A and VRF B

Scenario 2 describes the leak between two different VRFs.

## Network Diagram

This documents uses this network setup:



## Image 2. Route leaking Topology for Scenario 2

Router "LEAK" has a BGP neighborhood to a neighbor in VRF A, and an OSPF neighbor in the VRF B. Device 192.168.11.11 needs to connect to device 172.16.10.10 across the network.

Router LEAK cannot route between the two since the routes are in different VRFs. These routing tables show the current routes per VRF, and indicate which routes need to be leaked between the VRF A and VRF B.

LEAK Routing Table:

### VRF A Routing Table

```
LEAK#show ip route vrf A
```

Routing Table: A

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, m - OMP

n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA

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

H - NHRP, G - NHRP registered, g - NHRP registration summary

o - ODR, P - periodic downloaded static route, l - LISP

a - application route

+ - replicated route, % - next hop override, p - overrides from PFR

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

C 10.0.0.0/30 is directly connected, Ethernet0/0

L 10.0.0.2/32 is directly connected, Ethernet0/0

172.16.0.0/32 is subnetted, 1 subnets

**B 172.16.10.10 [200/0] via 10.0.0.1, 00:03:08** >> Route to be exchanged to routing table VRF B.

### VRF B Routing Table

```
LEAK#show ip route vrf B
```

Routing Table: B

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, m - OMP

n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA

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

H - NHRP, G - NHRP registered, g - NHRP registration summary

o - ODR, P - periodic downloaded static route, 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, Ethernet0/1

L 192.168.1.2/32 is directly connected, Ethernet0/1

192.168.11.0/32 is subnetted, 1 subnets

O 192.168.11.11 [110/11] via 192.168.1.1, 00:58:45, Ethernet0/1 >> Route to be exchange to routing table VRF A.

## Configure

Do these procedures to create the leak between the two routing tables:

### Step 1.

Create route-maps to filter the routes to be injected in both routing tables.

```
LEAK(config)#Route-map VRFA_TO_VRFB
LEAK(config-route-map)#match ip address prefix-list VRFA_TO_VRFB
LEAK(config-route-map)#exit
```

!  
Prefix-list created to match the host and IP segment that is attached to the previous route-map configured.

```
!  
ip prefix-list VRFA_TO_VRFB permit 172.16.10.10/32
ip prefix-list VRFA_TO_VRFB permit 10.0.0.0/30
```

or

```
LEAK(config)#Route-map VRFA_TO_VRFB
LEAK(config-route-map)#match ip address 10
LEAK(config-route-map)#exit
```

!  
ACL created to match the host and IP segment that is attached to the previous route-map.

```
!  
LEAK#show ip access-lists 10
10 permit 172.16.10.10
20 permit 10.0.0.0
```

```
LEAK(config)#Route-map VRFB_TO_VRFA
LEAK(config-route-map)#match ip address prefix-list VRFB_TO_VRFA
LEAK(config-route-map)#exit
```

!  
Prefix-list created to match the host and IP segment that is attached to the previous route-map configured.

```
!  
ip prefix-list VRFB_TO_VRFA permit 192.168.11.11/32
ip prefix-list VRFB_TO_VRFA permit 192.168.1.0/24
```

or

```
LEAK(config)#Route-map VRFB_TO_VRFA
LEAK(config-route-map)#match ip address 20
LEAK(config-route-map)#exit
```

!  
ACL created to match the host and IP segment that is attached to the previous route-map configured.

```
!  
LEAK#show ip access-lists 20
10 permit 192.168.11.11
20 permit 192.168.1.0
```

### Step 2.

At the VRFs configure the import/export map, use the route-map names to leak the routes.

```
LEAK(config)#vrf definition A
```

```
LEAK(config-vrf)#address-family ipv4
LEAK(config-vrf-af)#export map VRFA_TO_VRFB
LEAK(config-vrf-af)#import map VRFB_TO_VRFA
```

```
LEAK(config)#vrf definition B
LEAK(config-vrf)#address-family ipv4
LEAK(config-vrf-af)#export map VRFB_TO_VRFA
LEAK(config-vrf-af)#import map VRFA_TO_VRFB
```

### Step 3.

Add the route-target to import and export the route distinguisher from both VRFs.

```
! --- Current configuration for VRF A

vrf definition A
rd 1:2
!
address-family ipv4
route-target export 1:2
route-target import 1:1
exit-address-family

! --- Current configuration from VRF B

vrf definition B
rd 2:2
!
address-family ipv4
exit-address-family

! --- Import the routes from VRF B into VRF A

LEAK(config)#vrf definition A
LEAK(config-vrf)#address-family ipv4
LEAK(config-vrf-af)#route-target import 2:2

! --- Import routes from VRF A to VRF B and export routes from VRF B

LEAK(config-vrf-af)#vrf definition B
LEAK(config-vrf)#address-family ipv4
LEAK(config-vrf-af)#route-target import 1:2
LEAK(config-vrf-af)#route-target export 2:2
```

## Verify

Check the Routing Tables

### VRF A Routing Table

```
LEAK#show ip route vrf A
```

Routing Table: A

<Snip for resume >

```
10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 10.0.0.0/30 is directly connected, Ethernet0/0
L 10.0.0.2/32 is directly connected, Ethernet0/0
172.16.0.0/32 is subnetted, 1 subnets
```



```

B 172.16.10.10 [200/0] via 10.0.0.1, 00:07:20
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
B 192.168.1.0/24 is directly connected, 00:00:10, Ethernet0/1
L 192.168.1.2/32 is directly connected, Ethernet0/1
192.168.11.0/32 is subnetted, 1 subnets
B 192.168.11.11 [20/11] via 192.168.1.1 (B), 00:00:10, Ethernet0/1 >> Route from VRF B routing table at VRF A.

```

#### VRF B Routing Table

```
LEAK#show ip route vrf B
```

```
Routing Table: B
```

```
< Snip for resume >
```

```

10.0.0.0/30 is subnetted, 1 subnets
B 10.0.0.0 [200/0] via 10.0.0.1 (A), 00:00:15
172.16.0.0/32 is subnetted, 1 subnets
B 172.16.10.10 [200/0] via 10.0.0.1 (A), 00:00:15 >> Route from VRF A routing table at VRF B.
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.1.0/24 is directly connected, Ethernet0/1
L 192.168.1.2/32 is directly connected, Ethernet0/1
192.168.11.0/32 is subnetted, 1 subnets
O 192.168.11.11 [110/11] via 192.168.1.1, 01:05:12, Ethernet0/1

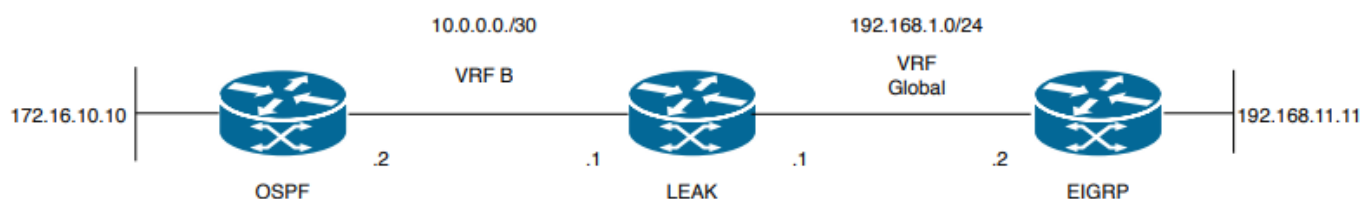
```

## Scenario 3 - VRF Leaking between OSPF ( VRF ) and EIGRP ( Global ) with BGP ( Optional )

Scenario 3 describes the route leak between two IGPs (VRF B and Global VRF).

### Network Diagram

This documents uses this network setup:



### Imagen 3. Route leaking Topology for Scenario 3

Router "LEAK" has an OSPF neighborhood to a neighbor in VRF B, and an EIGRP neighbor in the global VRF. Device 172.16.10.10 needs to be able to connect to device 192.168.11.11 across the network.

Router LEAK is not able to connect these two host. These routing tables show the current routes per VRF, and indicate which routes need to be leaked between the VRF B and Global VRF.

**Note:** This configuration is presented as an example for perform a leak when one of the IGPs is on a VRF, The use of redistribution between VRF and global VRF is not allowed in the devices.

LEAK Routing Table:

### EIGRP Routing Table ( EIGRP )

LEAK#show ip route

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, m - OMP  
n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA  
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  
H - NHRP, G - NHRP registered, g - NHRP registration summary  
o - ODR, P - periodic downloaded static route, 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, Ethernet0/1

L 192.168.1.1/32 is directly connected, Ethernet0/1

192.168.11.0/32 is subnetted, 1 subnets

**D 192.168.11.11 [90/1024640] via 192.168.1.2, 01:08:38, Ethernet0/1 >> Route to be exchange from global routing table at VRF B routing table.**

### VRF B Routing Table ( OSPF )

LEAK#show ip route vrf B

Routing Table: B

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, m - OMP  
n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA  
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  
H - NHRP, G - NHRP registered, g - NHRP registration summary  
o - ODR, P - periodic downloaded static route, l - LISP  
a - application route  
+ - replicated route, % - next hop override, p - overrides from PFR

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

C 10.0.0.0/30 is directly connected, Ethernet0/0

L 10.0.0.2/32 is directly connected, Ethernet0/0

172.16.0.0/32 is subnetted, 1 subnets

**O 172.16.10.10 [110/11] via 10.0.0.1, 01:43:45, Ethernet0/0 >> Route to be exchange from routing table VRF B at global routing table.**

## Configure

Do this procedures to create the leak between the two routing tables:

**Step 1.**

Create route-maps for import and export to be injected in both routing tables.

```
LEAK(config)#Route-map OSPF_TO_EIGRP
LEAK(config-route-map)#match ip address prefix-list OSPF_TO_EIGRP
LEAK(config-route-map)#exit
```

!  
Prefix-list created to match the host that is attached to the previous route-map configured.

```
!  
ip prefix-list OSPF_TO_EIGRP permit 172.16.10.10/32
ip prefix-list OSPF_TO_EIGRP permit 10.0.0.0/30
```

or

```
LEAK(config)#Route-map OSPF_TO_EIGRP
LEAK(config-route-map)#match ip address 10
LEAK(config-route-map)#exit
```

!  
ACL created to match the host that is attached to the previous route-map.

```
!  
LEAK#show ip access-lists 10
10 permit 172.16.10.10
20 permit 10.0.0.0
```

```
LEAK(config)#Route-map EIGRP_TO_OSPF
LEAK(config-route-map)#match ip address prefix-list EIGRP_TO_OSPF
LEAK(config-route-map)#exit
```

!  
Prefix-list created to match the host that is attached to the previous route-map configured.

```
!  
ip prefix-list EIGRP_TO_OSPF permit 192.168.11.11/32
ip prefix-list EIGRP_TO_OSPF permit 192.168.1.0/24
```

or

```
LEAK(config)#Route-map EIGRP_TO_OSPF
LEAK(config-route-map)#match ip address 20
LEAK(config-route-map)#exit
```

!  
ACL created to match the host that is attached to the previous route-map.

```
!  
LEAK#show ip access-lists 20
10 permit 192.168.11.11
20 permit 192.168.1.0/24
```

**Step 2.**

Add the import/export maps in order to match the route-map names.

Current configuration

```
!  
vrf definition B
rd 1:2
!  
address-family ipv4
exit-address-family
```

```
!  
!  
LEAK(config-vrf)#vrf definition B
LEAK(config-vrf)#address-family ipv4
LEAK(config-vrf-af)#import ipv4 unicast map EIGRP_TO_OSPF
```

```
LEAK(config-vrf-af)#export ipv4 unicast map OSPF_TO_EIGRP
```

### Step 3.

To perform the leak is necessary to create a BGP process, in order to redistribute the IGPs protocols.

```
router bgp 1
bgp log-neighbor-changes
!
address-family ipv4 vrf B >> Include the address-family to inject VRF B routing table (OSPF)
!
exit-address-family
```

**Note:** Ensure that the VRF has a Route Distinguisher configured in order to avoid the error:  
"%vrf B does not have "rd" configured, please configure "rd" before configuring import route-map"

### Step 4.

Create a Dual Redistribution.

IGPs redistribution.

```
LEAK(config-router)#router bgp 1
LEAK(config-router)#redistribute eigrp 1
!
LEAK(config-router)#address-family ipv4 vrf B
LEAK(config-router-af)#redistribute ospf 1 match internal external 1 external 2
LEAK(config-router-af)#end
```

BGP Redistribution

```
LEAK(config)#router ospf 1 vrf B
LEAK(config-router)#redistribute bgp 1
!
LEAK(config-router)#router eigrp TAC
LEAK(config-router)#
LEAK(config-router)# address-family ipv4 unicast autonomous-system 1
LEAK(config-router-af)#
LEAK(config-router-af)# topology base
LEAK(config-router-af-topology)#redistribute bgp 1 metric 100 1 255 1 1500
```

## Verify

Check the Routing Tables

### Global Routing Table

```
LEAK#show ip route
```

<Snip for resume >

```
172.16.0.0/32 is subnetted, 1 subnets
B 172.16.10.10 [20/11] via 10.0.0.1, 00:14:48, Ethernet0/0 >> Route from VRF B routing table at
global routing table ( EIGRP ).
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.1.0/24 is directly connected, Ethernet0/1
L 192.168.1.1/32 is directly connected, Ethernet0/1
192.168.11.0/32 is subnetted, 1 subnets
D 192.168.11.11 [90/1024640] via 192.168.1.2, 02:16:51, Ethernet0/1
```

### VRF B Routing Table

```
LEAK#show ip route vrf B
```

```
Routing Table: B
```

```
<Snip for resume >
```

```
10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
```

```
C 10.0.0.0/30 is directly connected, Ethernet0/0
```

```
L 10.0.0.2/32 is directly connected, Ethernet0/0
```

```
172.16.0.0/32 is subnetted, 1 subnets
```

```
O 172.16.10.10 [110/11] via 10.0.0.1, 00:34:25, Ethernet0/0
```

```
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
```

```
B 192.168.1.0/24 is directly connected, 00:08:51, Ethernet0/1
```

```
L 192.168.1.1/32 is directly connected, Ethernet0/1
```

```
192.168.11.0/32 is subnetted, 1 subnets
```

```
B 192.168.11.11 [20/1024640] via 192.168.1.2, 00:08:51, Ethernet0/1 >> Route from global routing table ( EIGRP ) at VRF B routing table.
```

## Additional Resources