



Configuring Easy Virtual Network Shared Services

This chapter describes how to use route replication and redistribution to share services in an Easy Virtual Network (EVN).

- [Finding Feature Information, on page 1](#)
- [Prerequisites for Virtual IP Network Shared Services, on page 1](#)
- [Restrictions for Virtual IP Network Shared Services, on page 2](#)
- [Information About Easy Virtual Network Shared Services, on page 2](#)
- [How to Share Services Using Easy Virtual Network , on page 5](#)
- [Configuration Example for Easy Virtual Network Shared Services, on page 14](#)
- [Additional References, on page 20](#)
- [Feature Information for Easy Virtual Network Shared Services, on page 21](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Virtual IP Network Shared Services

- Read the “Overview of Easy Virtual Networks” module.
- Implement EVN based on the “Configuring Easy Virtual Networks” module.

Restrictions for Virtual IP Network Shared Services

Route replication is supported for Static, Enhanced Interior Gateway Routing Protocol (EIGRP), and Open Shortest Path First (OSPF) routes. It is not possible to replicate routes to and from Border Gateway Protocol (BGP), but that is not an issue because the BGP import and export method of copying routes between Virtual Routing and Forwarding (VRF) is available in a virtual network.

Information About Easy Virtual Network Shared Services

Shared Services in an Easy Virtual Network

There are some common services (such as database and application servers) that multiple virtual networks need to access. Sharing these services are beneficial because:

- They are usually not duplicated for each group.
- It is economical, efficient, and manageable.
- Policies can be centrally deployed.

To achieve route separation, you could replicate the service, either physically or virtually, one service for each virtual network. However, that solution might not be cost effective or feasible. For a router that supports EVN, the solution is to perform route replication and route redistribution.

Route replication allows shared services because routes are replicated between virtual networks and clients who reside in one virtual network can reach prefixes that exist in another virtual network.

A shared services approach works best for Dynamic Name Systems (DNS), Dynamic Host Configuration Protocol (DHCP), and corporate communications. It is not a solution for sharing access to an Internet gateway.

Easy Virtual Network Shared Services Easier than VRF-Lite

Sharing servers in VRF-Lite requires route distinguishers (RDs), route targets with importing and exporting, and configuring BGP.

In an EVN environment, shared services are achieved with route replication, which is a simple deployment. Route replication requires no BGP, no RD, no route targets, and no import or export.

In summary, the BGP import and export method of copying routes between VRFs works with both VRF-Lite and EVN. However, route replication is the simpler alternative to enable sharing of common services across multiple virtual networks.

Route Replication Process in Easy Virtual Network

With shared services, clients and servers are located in different virtual networks. To achieve connectivity between clients and servers, routes must be exchanged among virtual networks. Depending on whether VRF-Lite or EVN is implemented, route exchanges among VRFs are accomplished in one of the following ways:

- If VRF-Lite is implemented, route leaking is achieved via BGP by using the route import/export feature.

- If EVN is implemented, route replication is supported directly by the Routing Information Base (RIB); there is no dependency on BGP. After routes are replicated from a different virtual network, those routes are propagated across each virtual network through existing redistribution into the Interior Gateway Protocol (IGP).

In the following route replication scenario, a router has two VRFs named Services and User-A. OSPF is configured:

```
router ospf 99 vrf services
 network 126.1.0.0 0.0.255.255 area 0
!
router ospf 98 vrf user-a
 network 126.1.0.0 0.0.255.255 area 0
```

Furthermore, route replication is configured for VRF User-A:

```
vrf definition user-a
!
 address-family ipv4
 route-replicate from vrf services unicast ospf 99
 exit-address-family
```

In the scenario, the following RIB for the VRF Services contains four routes, three of which are replicated to the RIB for VRF User-A. Route replication creates a link to the source RIB, as shown in the figure below.

RIB – VRF Services

Route	Type	Destination Interface	Next Hop
126.1.17.0/24	Connected	Gi0/1	
126.1.9.0/24	OSPF	Gi0/1	126.1.17.13
126.1.12.0/24	OSPF	Gi0/1	126.1.17.13
126.1.14.0/24	OSPF	Gi0/1	126.1.17.13

RIB – VRF User-A

Route	Type	Destination Interface	Next Hop
126.1.9.0/24	OSPF	Gi0/1	126.1.17.13
126.1.12.0/24	OSPF	Gi0/1	126.1.17.13
126.1.14.0/24	OSPF	Gi0/1	126.1.17.13



Configuring route replication allows mutual redistribution between virtual IP networks. In the case of shared services, you configure route replication within the VRF that needs access to shared services. Within each **route-replicate** command, you can optionally filter out routes with a route map to prevent a routing loop. That is, you do not want to redistribute routes back into the original routing protocol. You do not want a native route to show up as a replicated route.

Where to Implement Route Replication

We recommend implementing route replication on the router as close to the shared service as possible. Ideally, the router that is directly connected to the server subnet should be used, to eliminate the need to redistribute the host prefixes on the server VRF, and, thereby, avoid a potential routing loop.

Route Replication Behavior for Easy Virtual Network

This section describes the behavior of route replication for EVN, which differs from the behavior for Multi-Topology Routing. In an EVN environment:

- The **route-replicate** command is accepted only under the **address-family ipv4** command, which is configured under the **vrf definition** command.
- The **route-replicate** command replicates routes into the base topology within the specified address family.
- If **all** is specified as a source protocol, only one **route-replicate** command is allowed per VRF for a given destination topology.
- The **no route-replicate** command is allowed to exclude a source protocol.
- If **all** is specified as a source protocol, then connected routes are replicated (unlike in the Multi-Topology Routing version of the **route-replicate** command).
- A replicated route inherits the administrative distance and source protocol of the source route.

Route Preference Rules After Route Replication in Easy Virtual Network

If a route is replicated, the following rule determines route preference:

- If two routes are owned by the same protocol and have the same source VRF, and if one of the routes is NOT replicated, then the nonreplicated route is preferred.

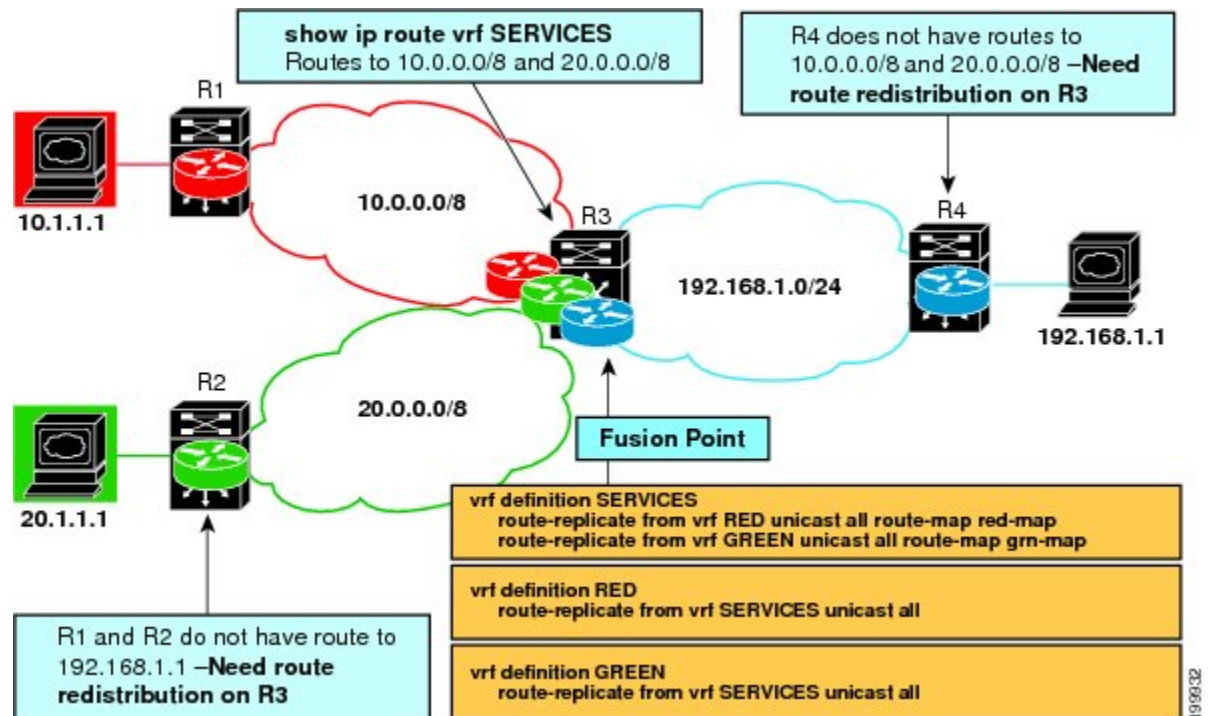
If the above rule does not apply, the following rules determine route preference, in this order:

1. Prefer the route with smaller administrative distance.
2. Prefer the route with smaller default administrative distance.
3. Prefer a non-replicated route over a replicated route.
4. Compare original vrf-names. Prefer the route with the lexicographically smaller vrf-name.
5. Compare original sub-address-families: Prefer unicast over multicast.
6. Prefer the oldest route.

How to Share Services Using Easy Virtual Network

Configuring Route Replication to Share Services in Easy Virtual Network

Perform this task to replicate routes from one VRF to another. The examples in the task table are based on the figure below.



In this particular task, routes from VRF SERVICES are replicated to both VRF RED and VRF GREEN, and VRF RED and VRF GREEN are not allowed to share routes between them. In order to allow bidirectional traffic, routes from VRF RED and VRF GREEN are also replicated to VRF SERVICES.



Note In a real EVN environment, there would also be route replication between VRF SERVICES and a third VRF, and maybe more VRFs. Such replication is left out of the following configuration task for the sake of brevity.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **vnet tag** *number*
5. **description** *string*
6. **address-family ipv4**
7. **exit**

8. **exit**
9. **vrf definition** *vrf-name*
10. **vnet tag** *number*
11. **description** *string*
12. **address-family ipv4**
13. **exit**
14. **exit**
15. **interface** *type number*
16. **vrf forwarding** *vrf-name*
17. **ip address** *ip-address mask*
18. **no shutdown**
19. **exit**
20. **router ospf** *process-id vrf vrf-name*
21. **network** *ip-address wildcard-mask area area-id*
22. **exit**
23. **router ospf** *process-id [vrf vrf-name]*
24. **network** *ip-address wildcard-mask area area-id*
25. **exit**
26. **vrf definition** *vrf-name*
27. **address-family ipv4**
28. **route-replicate from** [*vrf vrf-name*] {**multicast**|**unicast**} {**all**|*protocol-name*} [**route-map** *map-tag*]
29. **exit**
30. **exit**
31. **vrf definition** *vrf-name*
32. **address-family ipv4**
33. **route-replicate from** [*vrf vrf-name*] {**multicast**|**unicast**} {**all**|*protocol-name*} [**route-map** *map-tag*]
34. **end**
35. **show ip route vrf** *vrf-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	vrf definition <i>vrf-name</i> Example: Router(config)# vrf definition SERVICES	Defines a VRF and enters VRF configuration mode.

	Command or Action	Purpose
Step 4	vnet tag <i>number</i> Example: <pre>Router(config-vrf)# vnet tag 100</pre>	Specifies the global, numeric tag for the VRF. <ul style="list-style-type: none"> The same tag number must be configured for the same VRF on each edge and trunk interface.
Step 5	description <i>string</i> Example: <pre>Router(config-vrf)# description shared services</pre>	(Optional) Describes a VRF to help the network administrator looking at the configuration file.
Step 6	address-family ipv4 Example: <pre>Router(config-vrf)# address-family ipv4</pre>	Enters address family configuration mode to configure a routing session using standard IPv4 address prefixes.
Step 7	exit Example: <pre>Router(config-vrf-af)# exit</pre>	Exits to VRF configuration mode.
Step 8	exit Example: <pre>Router(config-vrf)# exit</pre>	Exits to global configuration mode.
Step 9	vrf definition <i>vrf-name</i> Example: <pre>Router(config)# vrf definition RED</pre>	Defines a VRF and enters VRF configuration mode.
Step 10	vnet tag <i>number</i> Example: <pre>Router(config-vrf)# vnet tag 200</pre>	Specifies the global, numeric tag for the VRF. <ul style="list-style-type: none"> The same tag number must be configured for the same VRF on each edge and trunk interface.
Step 11	description <i>string</i> Example: <pre>Router(config-vrf)# description user of services</pre>	(Optional) Describes a VRF to help the network administrator looking at the configuration file.
Step 12	address-family ipv4 Example: <pre>Router(config-vrf)# address-family ipv4</pre>	Enters address family configuration mode to configure a routing session using standard IP Version 4 address prefixes.
Step 13	exit Example:	Exits to VRF configuration mode.

	Command or Action	Purpose
	<code>Router(config-vrf-af)# exit</code>	
Step 14	exit Example: <code>Router(config-vrf)# exit</code>	Exits to global configuration mode.
Step 15	interface <i>type number</i> Example: <code>Router(config)# interface gigabitethernet 0/0/0</code>	Configures an interface type and number and enters interface configuration mode.
Step 16	vrf forwarding <i>vrf-name</i> Example: <code>Router(config-if)# vrf forwarding SERVICES</code>	Associates a VRF instance with an interface.
Step 17	ip address <i>ip-address mask</i> Example: <code>Router(config-if)# ip address 192.168.1.3 255.255.255.0</code>	Sets a primary IP address for an interface.
Step 18	no shutdown Example: <code>Router(config-if)# no shutdown</code>	Restarts an interface.
Step 19	exit Example: <code>Router(config-if)# exit</code>	Exits to global configuration mode.
Step 20	router ospf <i>process-id vrf vrf-name</i> Example: <code>Router(config)# router ospf 99 vrf SERVICES</code>	Configures an OSPF routing process and enters router configuration mode. <ul style="list-style-type: none">• This example uses OSPF; EIGRP is also available.
Step 21	network <i>ip-address wildcard-mask area area-id</i> Example: <code>Router(config-router)# network 192.168.1.0 0.0.0.255 area 0</code>	Defines the interfaces on which OSPF runs and the area ID for those interfaces.
Step 22	exit Example:	Exits to global configuration mode.

	Command or Action	Purpose
	<code>Router(config-router)# exit</code>	
Step 23	router ospf <i>process-id</i> [vrf <i>vrf-name</i>] Example: <code>Router(config)# router ospf 98 vrf RED</code>	Configures an OSPF routing process and enters router configuration mode.
Step 24	network <i>ip-address wildcard-mask area area-id</i> Example: <code>Router(config-router)# network 192.168.1.0 0.0.0.255 area 0</code>	Defines the interfaces on which OSPF runs and the area ID for those interfaces.
Step 25	exit Example: <code>Router(config-router)# exit</code>	Exits to the global configuration mode.
Step 26	vrf definition <i>vrf-name</i> Example: <code>Router(config)# vrf definition RED</code>	Defines a VRF and enters VRF configuration mode.
Step 27	address-family ipv4 Example: <code>Router(config-vrf)# address-family ipv4</code>	Enters address family configuration mode to configure a routing session using standard IPv4 address prefixes.
Step 28	route-replicate from [vrf <i>vrf-name</i>] { multicast unicast } { all <i>protocol-name</i> } [route-map <i>map-tag</i>] Example: <code>Router(config-vrf-af)# route replicate from vrf SERVICES unicast all</code>	Replicates routes into the base topology within the specified address family. <ul style="list-style-type: none"> • If the all keyword is specified as a source protocol, only one route-replicate command is allowed per VRF for a given destination topology. • Use the connected keyword as a source <i>protocol-name</i> in order to replicate only connected routes.
Step 29	exit Example: <code>Router(config-vrf-af)# exit</code>	Exits to VRF configuration mode.
Step 30	exit Example: <code>Router(config-vrf)# exit</code>	Exits to global configuration mode.

Example

	Command or Action	Purpose
Step 31	vrf definition <i>vrf-name</i> Example: Router(config)# vrf definition SERVICES	Defines a VRF and enters VRF configuration mode.
Step 32	address-family ipv4 Example: Router(config-vrf)# address-family ipv4	Enters address family configuration mode to configure a routing session using standard IPv4 address prefixes.
Step 33	route-replicate from [<i>vrf vrf-name</i>] { multicast unicast } { all <i>protocol-name</i> } [route-map <i>map-tag</i>] Example: Router(config-vrf-af)# route replicate from vrf RED unicast all	Replicates routes into the base topology within the specified address family. • This is the reciprocal replication to Step 28 to allow bidirectional traffic.
Step 34	end Example: Router(config-vrf-af)# end	Exits configuration mode.
Step 35	show ip route vrf <i>vrf-name</i> Example: Router# show ip route vrf RED	(Optional) Displays routes, including those replicated, which are indicated by a plus sign (+).

Example

The following is sample output from the **show ip route vrf** command based on the task in the preceding task table:

```
Router# show ip route vrf RED

Routing Table: RED
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override
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 (SERVICES), GigabitEthernet0/0/0
L   +   192.168.1.3/32 is directly connected (SERVICES), GigabitEthernet0/0/0
Router#
```

What to Do Next

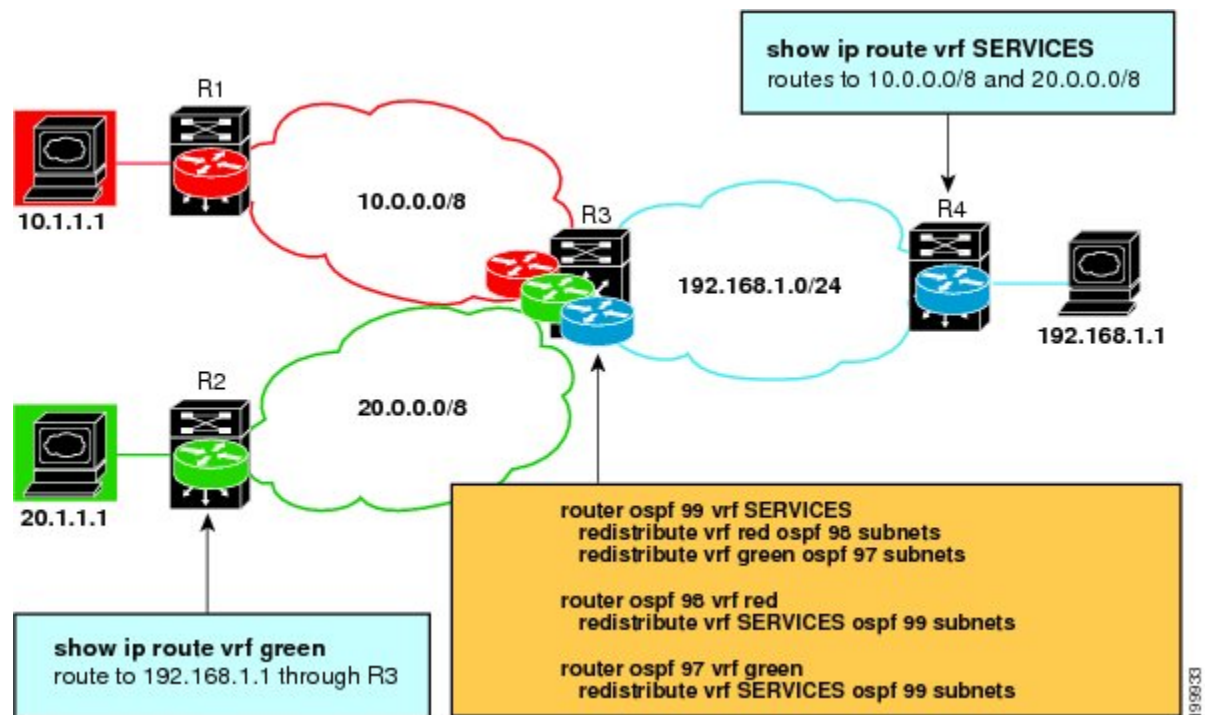
After you perform the “Configuring Route Replication to Share Services in Easy Virtual Network” task, you must configure VRF GREEN as per the figure above, noting that Router 3 has routes to 10.0.0.0/8 and 20.0.0.0/8 and Router 1 and Router 2 have a route to 192.168.1.0/24.

After the configuration is complete, Router 1 and Router 2 still do not have a route to the shared service residing on 192.168.1.1 and Router 4 does not have routes to 10.0.0.0/8 and 20.0.0.0/8. Such access requires the route redistribution performed in the next task, "Configuring Redistribution to Share Services in EVN".

Configuring Redistribution to Share Services in Easy Virtual Network

This task is based on the assumption that you also performed the task, Configuring Route Replication to Share Services in EVN.

The figure below shows the same networks we used in the figure above. In this task, we perform redistribution on Router 3 so that Router 1 and Router 2 have a route to the shared service residing on 192.168.1.1.



SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router ospf** *process-id* **vrf** *vrf-name*
4. **redistribute vrf** *vrf-name* **ospf** *process-id* **subnets**
5. **redistribute vrf** *vrf-name* **ospf** *process-id* **subnets**
6. **exit**
7. **router ospf** *process-id* **vrf** *vrf-name*
8. **redistribute vrf** *vrf-name* **ospf** *process-id* **subnets**

9. **exit**
10. **router ospf *process-id* vrf *vrf-name***
11. **redistribute vrf *vrf-name* ospf *process-id* subnets**
12. **end**
13. **show ip route vrf *vrf-name***

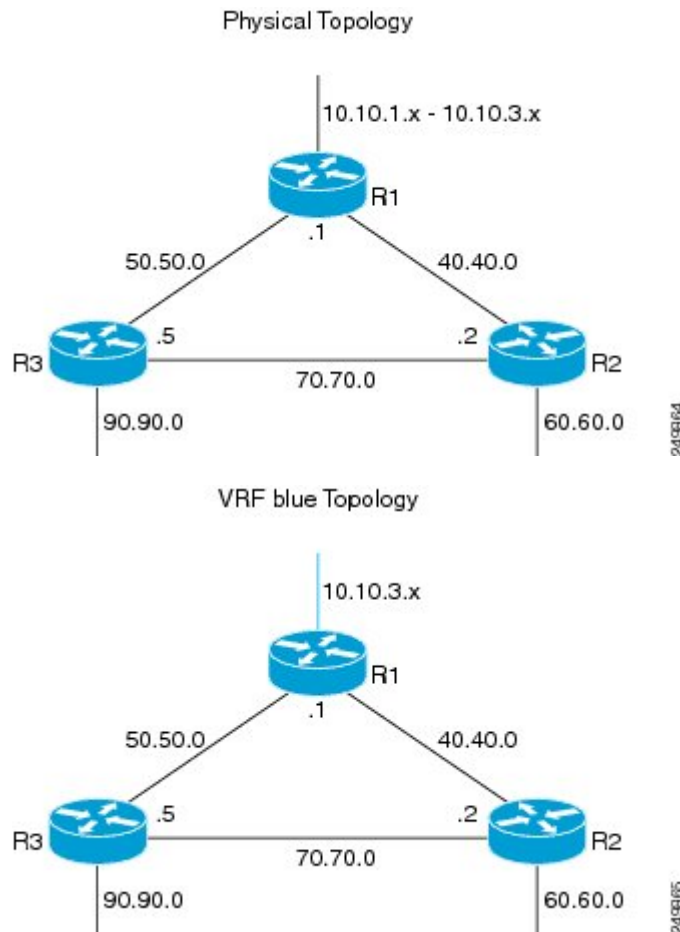
DETAILED STEPS

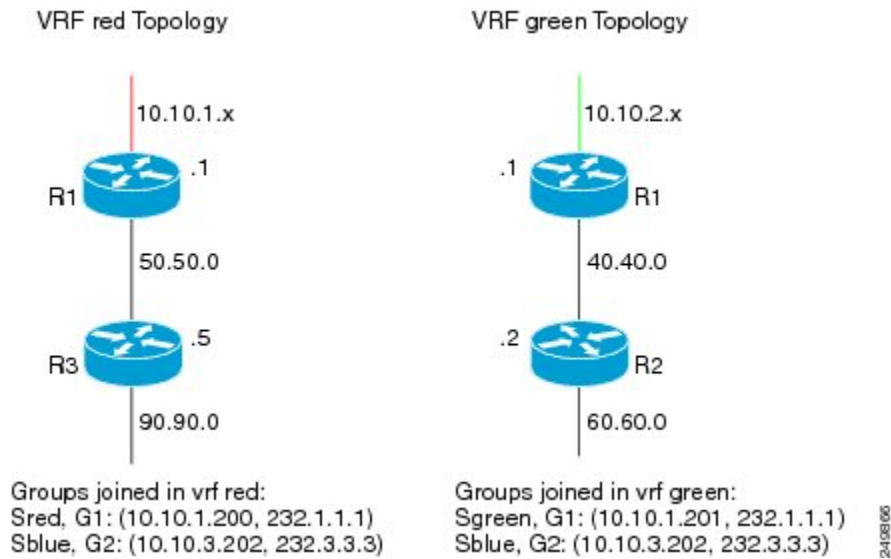
	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	router ospf <i>process-id</i> vrf <i>vrf-name</i> Example: Router(config)# router ospf 99 vrf SERVICES	Configures an OSPF routing process and enters router configuration mode.
Step 4	redistribute vrf <i>vrf-name</i> ospf <i>process-id</i> subnets Example: Router(config-router)# redistribute vrf RED ospf 98 subnets	Redistributes routes from one routing domain into another routing domain.
Step 5	redistribute vrf <i>vrf-name</i> ospf <i>process-id</i> subnets Example: Router(config-router)# redistribute vrf GREEN ospf 97 subnets	Redistributes routes from one routing domain into another routing domain.
Step 6	exit Example: Router(config-router)# exit	Exits to global configuration mode.
Step 7	router ospf <i>process-id</i> vrf <i>vrf-name</i> Example: Router(config)# router ospf 98 vrf RED	Configures an OSPF routing process and enters router configuration mode.
Step 8	redistribute vrf <i>vrf-name</i> ospf <i>process-id</i> subnets Example:	Redistributes routes from one routing domain into another routing domain.

	Command or Action	Purpose
	Router(config-router)# redistribute vrf SERVICES ospf 99 subnets	
Step 9	exit Example: Router(config-router)# exit	Exits to global configuration mode.
Step 10	router ospf process-id vrf vrf-name Example: Router(config)# router ospf 97 vrf GREEN	Configures an OSPF routing process and enters router configuration mode.
Step 11	redistribute vrf vrf-name ospf process-id subnets Example: Router(config-router)# redistribute vrf SERVICES ospf 99 subnets	Redistributes routes from one routing domain into another routing domain.
Step 12	end Example: Router(config-router)# end	Exits configuration mode.
Step 13	show ip route vrf vrf-name Example: Router# show ip route vrf RED	(Optional) Displays routes, including those replicated, which are indicated by a plus sign (+).

Configuration Example for Easy Virtual Network Shared Services

Example: Easy Virtual Network Route Replication and Route Redistribution in a Multicast Environment





In the figures above there are three multicast streams:

- Sred, G1: (10.10.1.200, 232.1.1.1)--Source and receivers in VRF red
- Sgreen, G1: (10.10.2.201, 232.1.1.1)--Source and receivers in VRF green
- Sblue, G2: (10.10.3.202, 232.3.3.3)--Source in blue and receivers in VRFs red and green.

The server-prefix in VRF blue (10.10.3.0/24) is replicated and distributed into VRFs red and green on R3 and R2.

Multicast group 232.3.3.3 with its source in VRF blue has receivers in both VRF red and VRF green. The stream is transmitted over the shared VRF (blue), and then replicated into VRF red on R3 and into VRF green on R2.

R1 Configuration

```
vrf definition blue
vnet tag 4
!
address-family ipv4
exit-address-family
!
vrf definition green
vnet tag 3
!
address-family ipv4
exit-address-family
!
vrf definition red
vnet tag 2
!
address-family ipv4
exit-address-family
!
vrf list vnet-list1
member blue
member red
!
```

```

vrf list vnet-list2
  member blue
  member green
!
vrf list vnet-list3
  member blue
!
ip multicast-routing distributed
ip multicast-routing vrf red distributed
ip multicast-routing vrf green distributed
ip multicast-routing vrf blue distributed
!
interface FastEthernet0/0/2
  vnet trunk list vnet-list1                                [vnet trunk for red and blue]
  ip address 50.50.0.1 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/1
  vnet trunk list vnet-list2                                [vnet trunk for green and blue]

  ip address 40.40.0.1 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3
  ip address 10.10.0.1 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3.2
  vrf forwarding red
  encapsulation dot1Q 2
  ip address 10.10.1.1 255.255.255.0
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3.3
  vrf forwarding green
  encapsulation dot1Q 3
  ip address 10.10.2.1 255.255.255.0
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3.4
  vrf forwarding blue
  encapsulation dot1Q 4
  ip address 10.10.3.1 255.255.255.0
  ip pim sparse-dense-mode
!
router ospf 201 vrf red
  nsf
  redistribute connected subnets
  network 10.10.1.0 0.0.0.255 area 0
  network 50.50.0.0 0.0.0.255 area 0
!
router ospf 202 vrf green
  nsf
  network 10.10.2.0 0.0.0.255 area 0
  network 40.40.0.0 0.0.0.255 area 0
!
router ospf 203 vrf blue
  router-id 11.11.11.11

```



```

nsf
network 10.10.3.0 0.0.0.255 area 0
network 40.40.0.0 0.0.0.255 area 0
network 50.50.0.0 0.0.0.255 area 0
!
router ospf 200
nsf
redistribute connected subnets
network 10.10.0.0 0.0.0.255 area 0
network 40.40.0.0 0.0.0.255 area 0
network 50.50.0.0 0.0.0.255 area 0
!
ip pim ssm default
ip pim vrf red ssm default
ip pim vrf green ssm default
ip pim vrf blue ssm default
!

```

R2 Configuration

```

vrf definition blue
vnet tag 4
!
address-family ipv4
exit-address-family
!
vrf definition green
vnet tag 3
!
address-family ipv4
route-replicate from vrf blue unicast all route-map blue-map
[replicate routes from blue to green]
exit-address-family
!
vrf definition red
vnet tag 2
!
address-family ipv4
exit-address-family
!
vrf list vnet-list1
member blue
member green
!
vrf list vnet-list2
member blue
!
ip multicast-routing distributed
ip multicast-routing vrf red distributed
ip multicast-routing vrf green distributed
ip multicast-routing vrf blue distributed
!
interface FastEthernet0/0/6
vnet trunk list vnet-list2 [vnet trunk for blue]
ip address 70.70.0.2 255.255.255.0
no ip redirects
no ip proxy-arp
ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/2
vnet trunk list vnet-list1 [vnet trunk for green and blue]
ip address 40.40.0.2 255.255.255.0
no ip redirects

```

```

no ip proxy-arp
ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/4
vnet trunk list vnet-list1 [vnet trunk for green and blue]

ip address 60.60.0.2 255.255.255.0
no ip redirects
no ip proxy-arp
ip pim sparse-dense-mode
!
router ospf 202 vrf green
redistribute connected subnets
redistribute vrf blue ospf 203 subnets route-map blue-map [redistribute routes replicated
from blue in red]
network 40.40.0.0 0.0.0.255 area 0
network 60.60.0.0 0.0.0.255 area 0
!
router ospf 203 vrf blue
router-id 22.22.22.22
network 40.40.0.0 0.0.0.255 area 0
network 60.60.0.0 0.0.0.255 area 0
network 70.70.0.0 0.0.0.255 area 0
!
router ospf 200
redistribute connected subnets
network 40.40.0.0 0.0.0.255 area 0
network 60.60.0.0 0.0.0.255 area 0
network 70.70.0.0 0.0.0.255 area 0
!
ip pim ssm default
ip pim vrf red ssm default
ip pim vrf green ssm default
ip pim vrf blue ssm default
!
ip prefix-list server-prefix seq 5 permit 10.10.3.0/24
!
route-map blue-map permit 10
match ip address prefix-list server-prefix
!

```

R3 Configuration

```

vrf definition blue
vnet tag 4
!
address-family ipv4
exit-address-family
!
vrf definition green
vnet tag 3
!
address-family ipv4
exit-address-family
!
vrf definition red
vnet tag 2
!
address-family ipv4
route-replicate from vrf blue unicast all route-map blue-map [replicate routes from
blue to red]
exit-address-family
!

```

```

vrf list vnet-list1
  member blue
  member red
!
vrf list vnet-list2
  member blue
!
ip multicast-routing distributed
ip multicast-routing vrf red distributed
ip multicast-routing vrf green distributed
ip multicast-routing vrf blue distributed
!
interface GigabitEthernet0/2/0
  vnet trunk list vnet-list1                                [vnet trunk for red and blue]
  ip address 90.90.0.5 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet1/2/0
  vnet trunk list vnet-list1                                [vnet trunk for red and blue]
  ip address 50.50.0.5 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface FastEthernet2/0/0
  vnet trunk list vnet-list2                                [vnet trunk for blue]
  ip address 70.70.0.5 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
router ospf 201 vrf red
  redistribute connected subnets
  redistribute vrf blue ospf 203 subnets route-map blue-map    [redistribute routes
replicated from blue in red]
  network 50.50.0.0 0.0.0.255 area 0
  network 90.90.0.0 0.0.0.255 area 0
!
router ospf 203 vrf blue
  router-id 55.55.55.55
  network 50.50.0.0 0.0.0.255 area 0
  network 70.70.0.0 0.0.0.255 area 0
  network 90.90.0.0 0.0.0.255 area 0
!
router ospf 200
  redistribute connected subnets
  network 50.50.0.0 0.0.0.255 area 0
  network 70.70.0.0 0.0.0.255 area 0
  network 90.90.0.0 0.0.0.255 area 0
!
ip pim ssm default
ip pim vrf red ssm default
ip pim vrf green ssm default
ip pim vrf blue ssm default
!
ip prefix-list server-prefix seq 5 permit 10.10.3.0/24
!
route-map blue-map permit 10
  match ip address prefix-list server-prefix
!

```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Easy Virtual Network commands	Easy Virtual Network Command Reference
Overview of Easy Virtual Network	“Overview of Easy Virtual Network” module in the <i>Easy Virtual Network Configuration Guide</i>
Configuring Easy Virtual Network	“Configuring Easy Virtual Network” module in the <i>Easy Virtual Network Configuration Guide</i>
Easy Virtual Network management and troubleshooting	“Easy Virtual Network Management and Troubleshooting” module in the <i>Easy Virtual Network Configuration Guide</i>

MIBs

MIB	MIBs Link
<p>Any MIB that gives VRF information will continue to work with Easy Virtual Network. VRF-independent MIBs report information on every VRF in a system:</p> <ul style="list-style-type: none"> • CISCO-MVPN-MIB • MPLS-VPN-MIB • CISCO-VRF-MIB 	<p>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

Technical Assistance

Description	Link
<p>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for Easy Virtual Network Shared Services

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Easy Virtual Network Shared Services

Feature Name	Releases	Feature Information
EVN Route Replication	Cisco IOS XE Release 3.2S 15.0(1)SY 15.1(1)SG Cisco IOS XE Release 3.3SG 15.3(2)T	This module describes how to use route replication and redistribution to share services in an EVN environment. This feature modifies the following command: redistribute (IP) This feature introduces the following command: route-replicate (VRF address family)

