Configuring VRF Route Sharing

The following chapter describes how you can configure VRF Route Sharing on a CSR 1000v instance. VRF Route Sharing is required when you need to forward traffic between an On-Premise Site and a Public Cloud Site. Configure VRF Route Sharing across VxLAN peers to deploy shared services across the cloud.

- Information About VRF Route Sharing, on page 1
- Limitations of VRF Route Sharing, on page 1
- Prerequisites of VRF Route Sharing, on page 2
- How to Configure VRF Route Sharing, on page 2
- Verifying VRF Route Sharing, on page 6

Information About VRF Route Sharing

In a hybrid cloud solution where there is an APIC layer (On-Premise) and a Public Cloud Site, the CSR 1000v instance connects the Data Centers through Layer-3 boundaries. The CSR 1000v instance has a VRF instance configured with two sets of import and export route-targets. One set of the import/export route target is associated with the BGP EVPN session with VXLAN encapsulation and L3 routing information in the On-Premise router. The other set of import/export route-target is associated with the L3VPN BGP neighbour in the service provider network. The CSR 1000v instance enables the L3 traffic movement across the EVPN by stitching the route between the On-Premise site and the service provider network.

The CSR 1000v instance forwards traffic across the EVPN even if the VRFs have the same VTEP IP (VxLAN tunnel endpoint) and RMAC (router MAC address). With this feature, the CSR1000v instance uses a binding label to setup the routing and forwaring chain.

Using the VRF Route Sharing functionality, you can deploy shared services across hybrid clouds. The shared services that run on the public cloud can be consumed by the endpoints on the On-Premise Site. The CSR 1000v instance shares the L3 prefix to multiple VRFs on the On-Premise Site, and vice versa. The APIC layer imports the addresses and the services are thus consumed in the APIC side.

Platforms Supported

The VRF Route Sharing functionality is currently supported only on Cisco CSR 1000v.

Limitations of VRF Route Sharing

- The VRF Sharing functionality supports up to 32 common VRFs, and 1000 customer VRF combination.

- This functionality does not support RT filters.
- VRF Route Sharing is supported only for IPv4 addresses and not IPv6 addresses.

Prerequisites of VRF Route Sharing

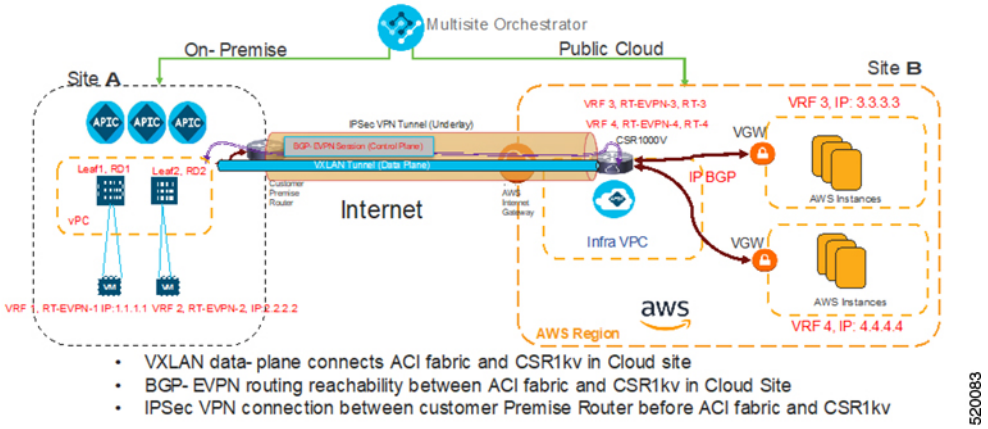
Before you configure the VRF Route Sharing functionality to enable the traffic between the ACI and the public cloud, ensure that:

- You configure VRF1 and VRF2 on the vPC pair of ACI.
- VRF3 are VRF4 on the CSR 1000v instance which peers with VGW have two RTs for each VRF.
- The CSR 1000v instance imports EVPN routes of VRF1&2 from ACI into VRF3&4.
- The IP BGP on the CSR 1000v side redistributes the routes to the gateway in the public cloud.
- The next-hop of routes from ACI are the spine of the border leaf of the ACI.
- There are no overlaps of prefix across the Route Sharing VRF.
- Advertise the L3 VPN routing and to forward the VRF prefixes to the EVPN neighbours. Run the advertise l2vpn evpn command and and export stitching RTs to push the native routes towards the EVPN.

How to Configure VRF Route Sharing

Sample Topology and Use Cases

The following is a sample topology that is used as an example to explain the VRF Route Sharing in a hybrid cloud:



In the above-mentioned topology, the CSR 1000v instance is deployed on the VM of the public cloud. Site A is an ACI deployment site, while Site B is the public cloud. Leaf 1 and Leaf 2 are the Virtual Port Channel (vPC) pair for ACI. These two vPCs are configured with different Route Distinguishers (RD). Here, VRF 1 and VRF 2 are configured on the vPC pair for ACI. For example,

VRF1 - RT:RT-EVPN-1, prefix:1.1.1.1

VRF2 - RT:RT-EVPN-2, prefix:2.2.2.2

VRF3 and VRF4 are configured on the CSR 1000v instance. These two VRFs pair with the Voice Gateway (VGW), and these two VRFs have two different Route Targets (RT). For example,

VRF3 – RT for EVPN: RT-EVPN-3, RT for IP BGP: RT-3, prefix:3.3.3.3

VRF4 – RT for EVPN: RT-EVPN-4, RT for IP BGP: RT-4, prefix:4.4.4.4

In the topology, the BGP-EVPN fabric is present between the ACI and the CSR 1000v instance in the public cloud and the IP BGP protocol is used between the CSR 1000v instance and the Cloud Service Provider such as Azure. The BGP-EVPN fabric redistributes the stitching routes between the EVPN and the IP BGP.

To enable the traffic flow between the ACI Site and the Public Cloud, both ACI and the CSR 1000v instance need to support VRF Route Sharing.

The CSR 1000v instance must be able to import the EVPN routes of VRF1 and VRF2 from ACI into VRF3 and VRF4. The IP BGP on the CSR side then redistributes the routes to the VGW in the public cloud.

Note When the VTEP (VxLAN Tunnel Endpoint) IP and the RMAC (Route MAC addrress) are the same for two leafs, and the VNIC alone differs, the CSR1000v instance can forward the traffic across the tunnel.

Use Cases

Using the same sample topology, here are the use cases for configuring VRF Route Sharing in a CSR 1000v instance:

- When VRF1 and VRF2 can talk to VRF3, but VRF3 and VRF4 cannot talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
vrf definition VRF4
rd 400:1
address-family ipv4
```

- When VRF1 and VRF2 can talk to VRF3&4, but VRF3 and VRF4 cannot talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
vrf definition VRF4
rd 400:1
address-family ipv4
route-target export RT-EVPN-4 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
```

- When VRF1 and VRF2 can talk to VRF3, but VRF3 and VRF4 can talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
route-target export RT-3
route-target import RT-4
vrf definition VRF4
rd 400:1
address-family ipv4
route-target import RT-3
route-target export RT-4
```

- When VRF1 and VRF2 can talk to VRF3&4, but VRF3 and VRF4 can talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
route-target export RT-3
route-target import RT-4
vrf definition VRF4
rd 400:1
address-family ipv4
route-target export RT-EVPN-4 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
route-target import RT-3
route-target export RT-4
```

Note For the above-mentioned use case, the CSR 1000v instance must configure EVPN on both VRF3 and VRF4.

Even IP BGP already imports all the routes from VRF3 and VRF4, BGP does not advertise the imported routes of the VRF to the EVPN peer.

You need to use the **Stitching** keyword in the configuration only when the sharing happens across the EVPN.

Configuring VRF Route Sharing

Perform the following configuration to configure VRF Route Sharing in a hybrid cloud where VRF 1 and VRF 2 (On-Premise) can talk to to VRF 3 and VRF 4 (in the public cloud). In this sample solution, VRF3 and VRF4 cannot talk to each other.

Example:

```
vrf definition vrf3
rd 3:3
address-family ipv4
Route-target export 100:3
```

```
Route-target import 100:4
route-target export 3:3 stitching
route-target import 1:1 stitching
route-target import 2:2 stitching
exit-address-family
!
!
vrf definition vrf4
rd 4:4
address-family ipv4
Route-target import 100:3
Route-target export 100:4
route-target export 4:4 stitching
route-target import 1:1 stitching
route-target import 2:2 stitching
exit-address-family
!
!
interface BDI100
no shutdown
vrf forwarding vrf3
ip address 10.1.1.1 255.255.255.224
!
interface GigabitEthernet4.2
encapsulation dot1Q 2
vrf forwarding vrf3
ip address 4.4.4.1 255.255.255.224
bridge-domain 100
member vni 10100
!
interface nve1
source-interface loopback0
host-reachability protocol bgp
member vni 10100 vrf vrf3
!
router bgp 100
bgp router-id 11.11.11.11
no bgp default ipv4-unicast
neighbor 22.22.22.22 remote-as 200
neighbor 22.22.22.22 update-source loopback0
neighbor 22.22.22.22 ebgp-multihop 255
address-family ipv4 vrf vrf3
redistribute connected
neighbor 4.4.4.2 remote-as 300
neighbor 4.4.4.2 activate
neighbor 4.4.4.2 send-community both
advertise l2vpn evpn
exit-address-family
!
address-family l2vpn evpn
neighbor 22.22.22.22 activate
neighbor 22.22.22.22 send-community both
exit-address-family
end
```

Verifying VRF Route Sharing

Step 1 **show ip bgp l2vpn evpn summary**.

Provides the BGP summary information for the VRF default address family (L2VPN EVPN).

Example:

```
show ip bgp l2vpn evpn summary
BGP router identifier 11.11.11.11, local AS number 100
BGP table version is 8, main routing table version 8
7 network entries using 2408 bytes of memory
......
BGP activity 14/0 prefixes, 16/0 paths, scan interval 60 secs
7 networks peaked at 17:34:38 Aug 14 2019 CST (00:00:26.895 ago)
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
22.22.22.22     4          200       6       5        4    0    0 00:01:23        4
Device#
```

Step 2 **show ip route vrf vrf3 bgp | in binding**.

Displays the IP routing table information associated with the VRF. When you see the output with the binding label, it indicates that the configuration is successful and BGP uses the binding label as the next hop.

Example:

```
+++ 17:35:05 Minuet(default) exec +++
show ip route vrf vrf3 bgp | in binding
B        10.2.1.0/24 [20/0] via binding label: 0x3000001, 00:00:26
B        10.2.2.0/24 [20/0] via binding label: 0x3000002, 00:00:26
B     192.168.1.0/24 [20/0] via binding label: 0x3000001, 00:00:26
B     192.168.2.0/24 [20/0] via binding label: 0x3000002, 00:00:26
Device#
```