Orbit VRF Feature Guide

05-7230A01, Version C

Table of Contents

Table of Figures

INTRODUCTION

SCOPE AND PURPOSE

This manual describes the typical use cases and configuration examples for VRF feature on the Orbit platform.

ABBREVIATIONS, ACRONYMS AND SYMBOLS

VRF Virtual Routing and Forwarding

APPLICABLE DOCUMENTS

The following documents form a part of this specification to the extent specified herein. Referenced documents are available from GE MDS or as an industry standard.

NORMATIVE

Ref	Title	Document Number	Location
RFC-KEYORDS	RFC 2119		http://datatracker.ietf.org/doc/rfc2119/

INFORMATIVE

Ref	Title	Document Number	Location

OVERVIEW

The network interfaces present on a router are typically part of the main/global routing instance, which allows traffic to be routed between them based on the routes in the main/global routing table and firewall rules governing flow of traffic between the interfaces.

The VRF feature enables the network interfaces to be segregated from each other by moving them into a separate routing instance with its own routing table. This enables layer-3 segmentation of the network by creating one or more isolated routing instances.

Orbit has had support for policy routing with multiple routing tables where the user can configure one or more routing tables along with rules to classify traffic on any IPv4 header fields and direct the matching traffic through a particular routing table. This enables routing of IPv4 traffic based on any header field rather than just destination address. The VRF feature enhances this support by introducing a new virtual network interface of type 'vrf' that is associated with a routing instance and allows other routable network interfaces like Ethernet, Bridge, Cellular etc. to be added to the VRF interface as members, thereby, confining them within the routing domain of that VRF.

FEATURES

Following table describes the features supported for the VRF functionality.

S.No.	Feature	Description
1.	Creating VRF interface and adding following routable interface types as members: - Ethernet - Cellular - Cellular-PDN (second APN) - Bridge - VLAN - GRE	Each VRF interface is associated with its own routing instance/table and the member interfaces exist within this routing table segregated from default/main and other VRFs in the system.
2.	VRF aware DMVPN	DMVPN instance can operate within a VRF, where the GRE interface uses a transport interface that is within a VRF. IPsec SA also operates within that VRF. The GRE tunnel interface itself is outside the VRF. It could be in the main/default or another VRF.
3.	VRF aware BGPv4, RIPv2 and OSPFv2	BGPv4 (IPv4 only), RIPv2 (IPv4 only) and OSPFv2 protocols can operate within a VRF (that is they import/export routes from the routing table associated with that VRF).
4.	VRF aware CLI commands	Following CLI commands can be passed VRF within which they need to execute: ping,ping6,ssh, telnet, traceroute,traceroute6

USE CASES

DUAL APN WITH EACH APN IN RESPECTIVE VRF

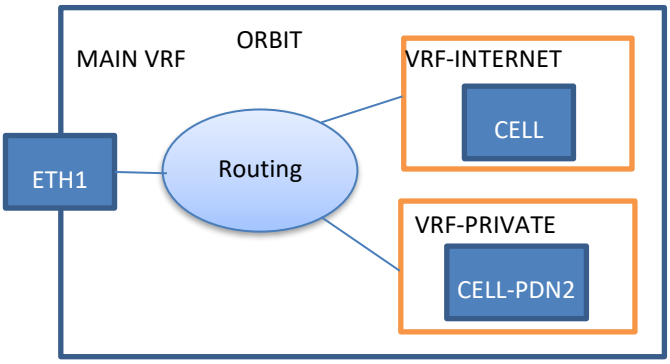

Figure 1 Dual APN with each APN within respective VRF

DESCRIPTION

In this use case, we have a single cellular modem in the Orbit with DUAL APN configuration where the PDN interface associated with each APN is in its respective VRF. In general, IP traffic cannot be routed between interfaces that are isolated in separate VRFs. There are two solutions to allow traffic received or originating in a VRF (for example, received on the ETH interface in main/global VRF in figure above) to be routed through another VRF:

1. Static route leaking – In this case, one configures a static route in the VRF where the traffic is received (or originates) with destination VRF as the outgoing-interface.
2. Policy based routing – In this case, one configures rules to match the given traffic and direct route lookup for this traffic in the destination VRF.

Static route leaking setup is easier, intuitive, and, therefore, preferred.

CONFIGURATION

The master cellular interface named ‘Cell’ contains all cellular modem related configuration including configuration for the second APN under extra-pdn configuration. This interface is also the network interface associated with the first APN.

1. Create master ‘Cell’ interface (if not already present on the device):

set interfaces interface Cell type cellular

set interfaces interface Cell enabled true

set interfaces interface Cell cell-config

2. Configure the first APN along with desired bearer protocol type. In this example, we configure the first PDN connection to be an IPv4 and IPv6 connection.

NOTE: When an explicit APN is not configured, the Orbit device will send LTE attach request without an APN, which allows the network to setup the PDN data session designated as ‘default’ APN. This is often desirable as it obviates

the need to configure APN explicitly. However, this is dependent on network policy as some LTE networks may require the device to send an explicit APN in the LTE attach request, in which case, it must be configured explicitly.

set interfaces interface Cell cell-config connection-profile PROFILE-1 bearer-config apn VZWINTERNET

set interfaces interface Cell cell-config connection-profile PROFILE-1 bearer-config protocol ipv4v6

3. Setup firewall filters, NAT rulesets and IP related configuration. Below we enable both IpV4 and IPv6 configuration.

set interfaces interface Cell filter input IN_UNTRUSTED

set interfaces interface Cell filter output OUT_UNTRUSTED

set interfaces interface Cell nat source MASQ

set interfaces interface Cell ipv4

set interfaces interface Cell ipv4 dhcp

set interfaces interface Cell ipv4 dhcp point-to-point-connection true

set interfaces interface Cell ipv6

set interfaces interface Cell ipv6 autoconf create-global-addresses true

set interfaces interface Cell ipv6 autoconf enabled true

4. Create second PDN network interface of type ‘cellular-pdn’ to associate with the second APN under master Cell interface configuration:

set interfaces interface Cell-PDN2 type cellular-pdn

set interfaces interface Cell-PDN2 enabled true

set interfaces interface Cell-PDN2 filter input IN_UNTRUSTED

set interfaces interface Cell-PDN2 filter output OUT_UNTRUSTED

set interfaces interface Cell-PDN2 ipv4

set interfaces interface Cell-PDN2 ipv4 dhcp

set interfaces interface Cell-PDN2 ipv4 dhcp request-routers true

set interfaces interface Cell-PDN2 ipv4 dhcp point-to-point-connection true

5. Configure second APN as an entry under ‘extra-pdn’ list and associate it with the Cell-PDN2 network interface:

set interfaces interface Cell cell-config connection-profile PROFILE-1 extra-pdn GEDEROC.GW6.VZWENTP interface Cell-PDN2

6. Create VRF interface named VRF-INTERNET

set interfaces interface VRF-INTERNET type vrf

set interfaces interface VRF-INTERNET vrf-config

set interfaces interface VRF-INTERNET filter input IN_UNTRUSTED

set interfaces interface VRF-INTERNET filter output OUT_UNTRUSTED

7. Create a routing instance and associate with the VRF interface. It is recommended that same name be used for VRF and routing instance for convenience.

set routing routing-instance VRF-INTERNET interface VRF-INTERNET

8. Add 'Cell' interface, which is the PDN network interface associated with first APN, to this VRF. This will move the Cell interface from main routing table to the routing table associated with this VRF:

   set interfaces interface VRF-INTERNET vrf-config member Cell

9. Create VRF interface named VRF-PRIVATE

set interfaces interface VRF-PRIVATE type vrf

set interfaces interface VRF-PRIVATE vrf-config

set interfaces interface VRF-PRIVATE filter input IN_UNTRUSTED

set interfaces interface VRF-PRIVATE filter output OUT_UNTRUSTED

10. Create a routing instance and associate with the VRF interface. It is recommended that same name be used for VRF and routing instance for convenience.

set routing routing-instance VRF-PRIVATE interface VRF-PRIVATE

11. Add 'Cell-PDN2' interface, which is the PDN network interface associated with second APN, to this VRF. This will move the Cell interface from main routing table to the routing table associated with this VRF:

set interfaces interface VRF-PRIVATE vrf-config member Cell-PDN2

12. Commit above configuration

    Commit

13. At this point, the first APN/PDN interface 'Cell' is in VRF-INTERNET and second APN/PDN interface named 'Cell-PDN2' is in VRF-PRIVATE. This can be seen using following CLI status commands.

    a. Following command show routes from main/global VRF

```
admin@(none) 10:04:28> show routing-state routes
                    NEXT  OUTGOING
DEST PREFIX     HOP   INTERFACE  SOURCE
-----------------------------------------
192.168.1.0/24  -     Bridge     kernel
fe80::/64       -               kernel
```

b. Following command show routes from VRF-INTERNET

```
admin@(none) 10:08:24> show routing-state routing-instance VRF-INTERNET routes
DEST       NEXT  OUTGOING
PREFIX     HOP   INTERFACE  SOURCE
-------------------------------------
0.0.0.0/0  -     Cell       dhcp
```

c. Following command show routes from VRF-PRIVATE

```
admin@(none) 10:08:26> show routing-state routing-instance VRF-PRIVATE  routes
DEST       NEXT  OUTGOING
PREFIX     HOP   INTERFACE  SOURCE
-------------------------------------
0.0.0.0/0  -     Cell-PDN2  dhcp
```

d. Following command shows that an external internet host (8.8.8.8) is NOT reachable from the main/global VRF:

```
admin@(none) 10:09:03> ping 8.8.8.8
connect: Network is unreachable
[error][2021-03-02 10:10:28]
```

e. Following command shows that an external internet host (8.8.8.8) is reachable from the VRF-INTERNET:

```
admin@(none) 11:14:44> ping 8.8.8.8 vrf VRF-INTERNET
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=54 time=191 ms
^C
--- 8.8.8.8 ping statistics ---
```

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 191.445/191.445/191.445/0.000 ms

[ok][2021-03-02 11:14:59]

f. Following command shows that an external internet host (8.8.8.8) is NOT reachable from the VRF-INTERNET:

admin@(none) 11:14:59> ping 8.8.8.8 vrf VRF-PRIVATE

PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.

^C

--- 8.8.8.8 ping statistics ---

1 packets transmitted, 0 received, 100% packet loss, time 0ms

[error][2021-03-02 11:15:03]

14. Following example shows how to use static route leaking to allow traffic received or originating in one VRF from reaching destinations that are reachable through another VRF

a. Following routing setup enables traffic received or originating in main/global VRF intended for external host 8.8.8.8 to be sent over the 'Cell' interface which is in VRF-INTERNET:

set routing static-routes ipv4 route 1 dest-prefix 8.8.8.8/32

set routing static-routes ipv4 route 1 outgoing-interface VRF-INTERNET

commit

b. Following command shows that external host 8.8.8.8 is now reachable from main/global VRF:

admin@(none) 11:57:30> ping 8.8.8.8

PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.

64 bytes from 8.8.8.8: icmp_seq=1 ttl=54 time=198 ms

64 bytes from 8.8.8.8: icmp_seq=2 ttl=54 time=54.6 ms

^C

--- 8.8.8.8 ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1000ms

rtt min/avg/max/mdev = 54.630/126.715/198.801/72.086 ms

[ok][2021-03-02 11:58:22]

c. Following routing setup enables traffic received or originating in main/global VRF intended for any host in 172.18.175.0/24 network to be sent over the 'Cell-PDN2' interface which is in VRF-PRIVATE:

```
set routing static-routes ipv4 route 2 outgoing-interface VRF-PRIVATE
set routing static-routes ipv4 route 2 dest-prefix 172.18.175.0/24
commit
```

d. Following command shows that an external host 172.18.175.40 is now reachable from main/global VRF:

```
admin@(none) 11:58:24> ping 172.18.175.40
PING 172.18.175.40 (172.18.175.40) 56(84) bytes of data.
64 bytes from 172.18.175.40: icmp_seq=1 ttl=61 time=202 ms
64 bytes from 172.18.175.40: icmp_seq=2 ttl=61 time=59.1 ms
^C
--- 172.18.175.40 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 59.109/130.725/202.342/71.617 ms
[ok][2021-03-02 12:01:19]
```

15. As an alternative to static route leaking setup described in the previous step, following shows an example of how to use policy-based routing to allow traffic received or originating in one VRF from reaching destinations that are reachable through another VRF.
    a. Following PBR setup enables traffic received or originating in main/global VRF intended for external host 8.8.8.8 to be sent over the 'Cell' interface which is in VRF-INTERNET:

```
set services qos classifier C1 match M1 ipv4 dst-address address 8.8.8.8/32
set routing rule 1 classifier [ C1 ]
set routing rule 1 action lookup VRF-INTERNET
commit
```

b. Following command shows that external host 8.8.8.8 is now reachable from main/global VRF:

```
admin@(none) 13:28:34> ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=214 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=57.5 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 57.558/136.060/214.562/78.502 ms
```

[ok][2021-03-02 13:28:50]

c. Following PBR setup enables traffic received or originating in main/global VRF intended for any host in 172.18.175.0/24 network to be sent over the 'Cell-PDN2' interface which is in VRF-PRIVATE:

```
set services qos classifier C2 match M1 ipv4 dst-address address 172.18.175.0/24
set routing rule 2 classifier [ C2 ]
set routing rule 2 action lookup VRF-PRIVATE
commit
```

d. Following command shows that an external host 172.18.175.40 is now reachable from main/global VRF:

```
admin@(none) 13:28:50> ping 172.18.175.40
PING 172.18.175.40 (172.18.175.40) 56(84) bytes of data.
64 bytes from 172.18.175.40: icmp_seq=1 ttl=61 time=204 ms
64 bytes from 172.18.175.40: icmp_seq=2 ttl=61 time=65.5 ms
^C
--- 172.18.175.40 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 65.513/135.178/204.844/69.666 ms
```

WAN REDUNDANCY ON DMVPN SPOKE

Dual WAN links can be used to provide transport redundancy for DMVPN spoke. For example, Ethernet WAN connection can be the primary link and the cellular WAN connection can be the backup link. From security standpoint these are often untrusted interfaces and user traffic traversing these transport networks needs to be protected using DMVPN.

The VRF feature allows the redundant WAN links to be isolated within their respective VRFs, each running a DMVPN instance, such that the traffic generated by the DMVPN GRE tunnel interfaces post encapsulation uses the respective VRF routing tables. The DMVPN GRE tunnel interfaces themselves can be in the global/main VRF.

Dynamic routing protocol like BGP can be used on the DMVPN tunnel interfaces to provide path redundancy for the user application traffic.

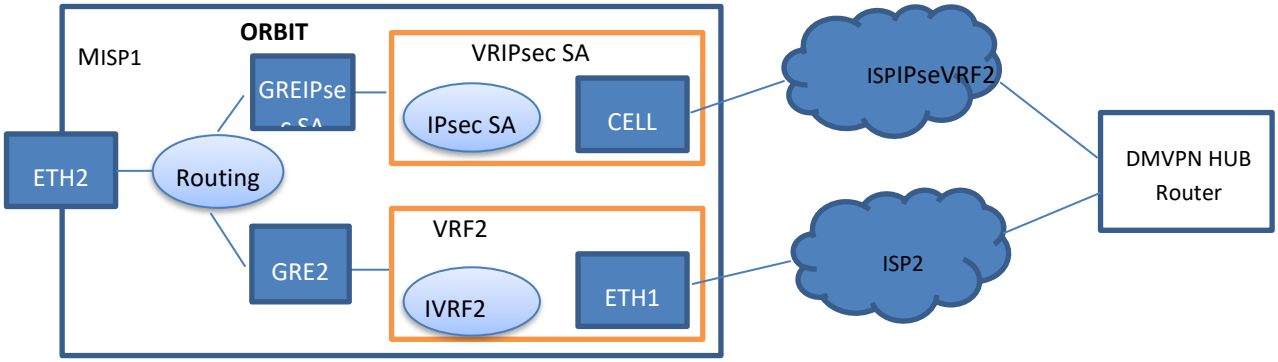

Figure 2 WAN Redundancy on DMVPN Spoke

Following sections describe configuration of this use case with following IP addressing scheme and other details:

DMVPN SPOKE:

- Cell-192.168.3.22/32 - The WAN transport/underlay interface simulating link to ISP1. The IP address is assigned by the cellular network.
- ETH1-172.23.16.22/24 - The WAN transport/underlay interface simulating link to ISP2.
- VRF1- The VRF interface tied to VRF1 routing instance that contains Cell interface as its member.
- VRF2- The VRF interface tied to VRF2 routing instance that contains ETH1 interface as its member.
- GRE1-172.16.1.22/24 - The DMVPN GRE overlay tunnel interface that routes encapsulated packets through VRF1 over the cellular network.
- GRE2-172.16.2.22/24 - The DMVPN GRE overlay tunnel interface that routes encapsulated packets through VRF2 over the LAN network.
- BGP HUB-CELL – The BGP session established with the HUB router over GRE1 tunnel interface. The route to LO1 interface is exported with AS-PATH attribute prepended with local-as (65001) to make this path less preferable than one over the LAN.
- BGP HUB-LAN – The BGP session established with the HUB router over GRE2 tunnel interface.
- LO1-10.1.1.22/32 – The loopback interface that is advertised by BGP.

DMVPN HUB:

Note: In this example, the configuration on HUB does not use VRFs. This is just to show that VRF is a local construct.

- Cell-192.168.3.24/32 - The WAN transport/underlay interface simulating link to ISP1. The IP address is assigned by the cellular network.
- ETH1-172.23.16.24/24 - The WAN transport/underlay interface simulating link to ISP2.
- GRE1-172.16.1.24/24 - The DMVPN GRE overlay tunnel interface that routes encapsulated packets through default/main VRF over Cellular network.
- GRE2-172.16.2.24/24 - The DMVPN GRE overlay tunnel interface that routes encapsulated packets through default/main VRF over LAN network.
- BGP HUB-CELL – The BGP session established with the SPOKE router over GRE1 tunnel interface. The default is exported with AS-PATH attribute prepended with local-as (65000) to make this path less preferable than one over the LAN.
- BGP HUB-LAN – The BGP session established with the SPOKE router over GRE2 tunnel interface.
- LO1-10.1.1.24/32 – The loopback interface.

DMVPN SPOKE CONFIGURATION

1. Configure Cell interface

```
set interfaces interface Cell type cellular
set interfaces interface Cell cell-config
set interfaces interface Cell cell-config connection-profile PROFILE-1 bearer-config protocol ip
set interfaces interface Cell filter input IN_UNTRUSTED
set interfaces interface Cell filter output OUT_UNTRUSTED
set interfaces interface Cell nat source MASQ
set interfaces interface Cell ipv4
set interfaces interface Cell ipv4 dhcp
set interfaces interface Cell ipv4 dhcp request-routers true
set interfaces interface Cell ipv4 dhcp point-to-point-connection true
```

2. Configure ETH1 interface.

NOTE: Be sure to remove it from Bridge interface.

```
delete interfaces interface Bridge bridge-settings members port ETH1
set interfaces interface ETH1 type ethernet
set interfaces interface ETH1 filter input IN_UNTRUSTED
set interfaces interface ETH1 filter output OUT_UNTRUSTED
set interfaces interface ETH1 ipv4
set interfaces interface ETH1 ipv4 address 172.16.23.22 prefix-length 24
```

3. Create VRF1 interface with Cell interface as a member.

```
set interfaces interface VRF1 type vrf
set interfaces interface VRF1 vrf-config
set interfaces interface VRF1 vrf-config member Cell
set interfaces interface VRF1 filter input IN_UNTRUSTED
```

set interfaces interface VRF1 filter output OUT_UNTRUSTED

4. Create VRF2 interface with ETH1 interface as a member.

set interfaces interface VRF2 type vrf
set interfaces interface VRF2 vrf-config
set interfaces interface VRF2 vrf-config member ETH1
set interfaces interface VRF2 filter input IN_UNTRUSTED
set interfaces interface VRF2 filter output OUT_UNTRUSTED

5. Create IPsec connection to secure GRE_DMVPN_t4 tunnel traffic

set services vpn enabled true
set services vpn ike
set services vpn ike policy DMVPN_t4_ike_policy version ikev2
set services vpn ike policy DMVPN_t4_ike_policy auth-method pre-shared-key
set services vpn ike policy DMVPN_t4_ike_policy pre-shared-key
8R4ZXPPWoJIKz9lEvQeTgy6uTRqZFLyO0uJj3VdLQ06Q=
set services vpn ike policy DMVPN_t4_ike_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ike policy DMVPN_t4_ike_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ike policy DMVPN_t4_ike_policy ciphersuite CS1 dh-group dh14
set services vpn ike policy DMVPN_t4_ike_policy life-time 180
set services vpn ike policy DMVPN_t4_ike_policy reauth false
set services vpn ike policy DMVPN_t4_ike_policy mobike true
set services vpn ike peer DMVPN_t4_ike_peer ike-policy DMVPN_t4_ike_policy
set services vpn ike peer DMVPN_t4_ike_peer local-endpoint any
set services vpn ike peer DMVPN_t4_ike_peer local-identity default
set services vpn ike peer DMVPN_t4_ike_peer peer-endpoint address 192.168.3.24
set services vpn ike peer DMVPN_t4_ike_peer peer-identity default
set services vpn ike peer DMVPN_t4_ike_peer role responder
set services vpn ike peer DMVPN_t4_ike_peer dpd-enabled true
set services vpn ike peer DMVPN_t4_ike_peer dpd-interval 300
set services vpn ipsec connection DMVPN_t4 ike-peer DMVPN_t4_ike_peer
set services vpn ipsec connection DMVPN_t4 ipsec-policy DMVPN_t4_ipsec_policy
set services vpn ipsec connection DMVPN_t4 host-to-host
set services vpn ipsec connection DMVPN_t4 failure-retry-interval 5
set services vpn ipsec connection DMVPN_t4 filter input IN_TRUSTED
set services vpn ipsec connection DMVPN_t4 filter output OUT_TRUSTED
set services vpn ipsec connection DMVPN_t4 underlay-vrf VRF1

6. Create IPsec connection to secure GRE_DMVPN-2_t4 tunnel traffic

set services vpn ike policy DMVPN-2_t4_ike_policy version ikev2
set services vpn ike policy DMVPN-2_t4_ike_policy auth-method pre-shared-key
set services vpn ike policy DMVPN-2_t4_ike_policy pre-shared-key
8TO+F2V2Ol/bGtDgOZH01Bibto3Koz9hG332xaxL+wws=
set services vpn ike policy DMVPN-2_t4_ike_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ike policy DMVPN-2_t4_ike_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ike policy DMVPN-2_t4_ike_policy ciphersuite CS1 dh-group dh14
set services vpn ike policy DMVPN-2_t4_ike_policy life-time 180
set services vpn ike policy DMVPN-2_t4_ike_policy reauth false
set services vpn ike policy DMVPN-2_t4_ike_policy mobike true

set services vpn ike peer DMVPN-2_t4_ike_peer ike-policy DMVPN-2_t4_ike_policy
set services vpn ike peer DMVPN-2_t4_ike_peer local-endpoint any
set services vpn ike peer DMVPN-2_t4_ike_peer local-identity default
set services vpn ike peer DMVPN-2_t4_ike_peer peer-endpoint address 172.16.23.24
set services vpn ike peer DMVPN-2_t4_ike_peer peer-identity default
set services vpn ike peer DMVPN-2_t4_ike_peer role responder
set services vpn ike peer DMVPN-2_t4_ike_peer dpd-enabled true
set services vpn ike peer DMVPN-2_t4_ike_peer dpd-interval 300
set services vpn ipsec policy DMVPN-2_t4_ipsec_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ipsec policy DMVPN-2_t4_ipsec_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ipsec policy DMVPN-2_t4_ipsec_policy life-time 60
set services vpn ipsec policy DMVPN_t4_ipsec_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ipsec policy DMVPN_t4_ipsec_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ipsec policy DMVPN_t4_ipsec_policy life-time 60
set services vpn ipsec connection DMVPN-2_t4 ike-peer DMVPN-2_t4_ike_peer
set services vpn ipsec connection DMVPN-2_t4 ipsec-policy DMVPN-2_t4_ipsec_policy
set services vpn ipsec connection DMVPN-2_t4 host-to-host
set services vpn ipsec connection DMVPN-2_t4 failure-retry-interval 5
set services vpn ipsec connection DMVPN-2_t4 filter input IN_TRUSTED
set services vpn ipsec connection DMVPN-2_t4 filter output OUT_TRUSTED
set services vpn ipsec connection DMVPN-2_t4 underlay-vrf VRF2

7. Create GRE_DMVPN_t4 tunnel interface to use VRF1 for route lookup for post-GRE encapsulated packets

set interfaces interface GRE_DMVPN_t4 type gre
set interfaces interface GRE_DMVPN_t4 enabled true
set interfaces interface GRE_DMVPN_t4 gre-config
set interfaces interface GRE_DMVPN_t4 gre-config mode ip-over-gre
set interfaces interface GRE_DMVPN_t4 gre-config src-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4 gre-config dst-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4 gre-config key 1000
set interfaces interface GRE_DMVPN_t4 gre-config underlay-vrf VRF1
set interfaces interface GRE_DMVPN_t4 gre-config ipsec-connection DMVPN_t4
set interfaces interface GRE_DMVPN_t4 filter input IN_TRUSTED
set interfaces interface GRE_DMVPN_t4 filter output OUT_TRUSTED
set interfaces interface GRE_DMVPN_t4 ipv4
set interfaces interface GRE_DMVPN_t4 ipv4 mtu 1300
set interfaces interface GRE_DMVPN_t4 ipv4 address 172.16.1.22 prefix-length 24

8. Create GRE_DMVPN_t4-2 tunnel interface to use VRF2 for route lookup for post-GRE encapsulated packets.

set interfaces interface GRE_DMVPN_t4-2 type gre
set interfaces interface GRE_DMVPN_t4-2 enabled true
set interfaces interface GRE_DMVPN_t4-2 gre-config
set interfaces interface GRE_DMVPN_t4-2 gre-config mode ip-over-gre
set interfaces interface GRE_DMVPN_t4-2 gre-config src-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4-2 gre-config dst-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4-2 gre-config key 1001
set interfaces interface GRE_DMVPN_t4-2 gre-config underlay-vrf VRF2
set interfaces interface GRE_DMVPN_t4-2 gre-config ipsec-connection DMVPN-2_t4
set interfaces interface GRE_DMVPN_t4-2 filter input IN_TRUSTED
set interfaces interface GRE_DMVPN_t4-2 filter output OUT_TRUSTED

set interfaces interface GRE_DMVPN_t4-2 ipv4
set interfaces interface GRE_DMVPN_t4-2 ipv4 mtu 1300
set interfaces interface GRE_DMVPN_t4-2 ipv4 address 172.16.2.22 prefix-length 24

9. Configure NHRP service to register with DMVPN HUB router over the GRE tunnel interfaces.

set services nhrp enabled true
set services nhrp interface GRE_DMVPN_t4 map HUB protocol-address 172.16.1.24
set services nhrp interface GRE_DMVPN_t4 map HUB protocol-netmask 255.255.255.0
set services nhrp interface GRE_DMVPN_t4 map HUB nbma-address 192.168.3.24
set services nhrp interface GRE_DMVPN_t4 map HUB register true
set services nhrp interface GRE_DMVPN_t4 map HUB cisco false
set services nhrp interface GRE_DMVPN_t4 authentication
8yTnMCAUz/L+KJnzC0m+s5jeLhZ0q1W+pTGluOaNIO9g=
set services nhrp interface GRE_DMVPN_t4 holding-time 300

set services nhrp interface GRE_DMVPN_t4-2 map HUB protocol-address 172.16.2.24
set services nhrp interface GRE_DMVPN_t4-2 map HUB protocol-netmask 255.255.255.0
set services nhrp interface GRE_DMVPN_t4-2 map HUB nbma-address 172.16.23.24
set services nhrp interface GRE_DMVPN_t4-2 map HUB register true
set services nhrp interface GRE_DMVPN_t4-2 map HUB cisco false
set services nhrp interface GRE_DMVPN_t4-2 authentication
8oNyW5Nj2614TYsFQmmOheZa4QEXs7yC6zKFZprwTHhk=
set services nhrp interface GRE_DMVPN_t4-2 holding-time 300

10. Create loopback interface to allow remote hosts to access this device through the tunnel

set interfaces interface LO1 type loopback
set interfaces interface LO1 enabled true
set interfaces interface LO1 ipv4
set interfaces interface LO1 ipv4 address 10.1.1.22 prefix-length 32

11. Configure router ID

set routing router-id 10.1.1.22

12. Create route filter to export loopback address to DMVPN HUB router via BGP

set routing route-filter LO1 rule 1 match outgoing-interface LO1
set routing route-filter LO1 rule 1 actions action accept

13. Create routing instances and associate to respective VRF interfaces

set routing routing-instance VRF1 interface VRF1
set routing routing-instance VRF2 interface VRF2

14. Configure BGP session over GRE_DMVPN_t4 tunnel interface and export loopback address.

set routing route-filter LO1-OVER-CELL rule 1 match outgoing-interface LO1
set routing route-filter LO1-OVER-CELL rule 1 actions action accept
set routing route-filter LO1-OVER-CELL rule 1 actions set bgp-as-path 65001

set routing bgp neighbor HUB-CELL peer-address 172.16.1.24

set routing bgp neighbor HUB-CELL local-address 172.16.1.22
set routing bgp neighbor HUB-CELL enabled true
set routing bgp neighbor HUB-CELL export-filter LO1-OVER-CELL
set routing bgp neighbor HUB-CELL passive false
set routing bgp neighbor HUB-CELL local-as 65001
set routing bgp neighbor HUB-CELL peer-as 65000
set routing bgp neighbor HUB-CELL next-hop-self false
set routing bgp neighbor HUB-CELL hold-time 240
set routing bgp neighbor HUB-CELL keepalive-time 80
set routing bgp neighbor HUB-CELL connect-retry-time 120

15. Configure BGP session over GRE_DMVPN_t4-2 tunnel interface and export loopback address.

set routing route-filter LO1-OVER-LAN rule 1 match outgoing-interface LO1
set routing route-filter LO1-OVER-LAN rule 1 actions action accept

set routing bgp neighbor HUB-LAN peer-address 172.16.2.24
set routing bgp neighbor HUB-LAN local-address 172.16.2.22
set routing bgp neighbor HUB-LAN enabled true
set routing bgp neighbor HUB-LAN export-filter LO1-OVER-LAN
set routing bgp neighbor HUB-LAN passive false
set routing bgp neighbor HUB-LAN local-as 65001
set routing bgp neighbor HUB-LAN peer-as 65000
set routing bgp neighbor HUB-LAN next-hop-self false
set routing bgp neighbor HUB-LAN hold-time 30
set routing bgp neighbor HUB-LAN keepalive-time 10
set routing bgp neighbor HUB-LAN connect-retry-time 30

16. Configure firewall

set services firewall enabled true
set services firewall address-set CELL-IP
set services firewall filter IN_TRUSTED rule 10 match protocol all
set services firewall filter IN_TRUSTED rule 10 actions
set services firewall filter IN_TRUSTED rule 10 actions action accept
set services firewall filter IN_UNTRUSTED rule 1 match protocol icmp
set services firewall filter IN_UNTRUSTED rule 1 actions
set services firewall filter IN_UNTRUSTED rule 1 actions action accept
set services firewall filter IN_UNTRUSTED rule 2 match protocol udp
set services firewall filter IN_UNTRUSTED rule 2 match src-port
set services firewall filter IN_UNTRUSTED rule 2 match src-port services [dns]
set services firewall filter IN_UNTRUSTED rule 3 match protocol tcp
set services firewall filter IN_UNTRUSTED rule 3 match dst-port
set services firewall filter IN_UNTRUSTED rule 3 match dst-port services [http https netconf ssh]
set services firewall filter IN_UNTRUSTED rule 4 match protocol udp
set services firewall filter IN_UNTRUSTED rule 4 match dst-port
set services firewall filter IN_UNTRUSTED rule 4 match dst-port services [ike]
set services firewall filter IN_UNTRUSTED rule 4 actions
set services firewall filter IN_UNTRUSTED rule 4 actions action accept
set services firewall filter IN_UNTRUSTED rule 5 match protocol esp
set services firewall filter IN_UNTRUSTED rule 5 actions
set services firewall filter IN_UNTRUSTED rule 5 actions action accept
set services firewall filter IN_UNTRUSTED rule 10 match protocol all

```
set services firewall filter IN_UNTRUSTED rule 10 actions
set services firewall filter IN_UNTRUSTED rule 10 actions action drop
set services firewall filter OUT_TRUSTED rule 10 match protocol all
set services firewall filter OUT_TRUSTED rule 10 actions
set services firewall filter OUT_TRUSTED rule 10 actions action accept
set services firewall filter OUT_UNTRUSTED rule 1 match protocol gre
set services firewall filter OUT_UNTRUSTED rule 1 actions
set services firewall filter OUT_UNTRUSTED rule 1 actions action drop
set services firewall filter OUT_UNTRUSTED rule 2 match src-address
set services firewall filter OUT_UNTRUSTED rule 2 match src-address address-set CELL-IP
set services firewall filter OUT_UNTRUSTED rule 2 match src-address add-interface-address true
set services firewall filter OUT_UNTRUSTED rule 2 actions
set services firewall filter OUT_UNTRUSTED rule 2 actions action accept
set services firewall filter OUT_UNTRUSTED rule 10 match protocol all
set services firewall filter OUT_UNTRUSTED rule 10 actions
set services firewall filter OUT_UNTRUSTED rule 10 actions action drop
set services firewall nat source rule-set MASQ rule 1 source-nat interface
```

DMVPN SPOKE STATUS

1. Following command show routes within VRF1

```
admin@MCR-MC7455 11:41:36> show routing-state routing-instance VRF1 routes
DEST      NEXT  OUTGOING
PREFIX    HOP   INTERFACE  SOURCE
-------------------------------------
0.0.0.0/0 -     Cell       dhcp

[ok][2021-03-04 11:41:39]
```

2. Following command show routes within VRF2

```
admin@MCR-MC7455 11:41:39> show routing-state routing-instance VRF2 routes
            NEXT  OUTGOING
DEST PREFIX     HOP   INTERFACE  SOURCE
-----------------------------------------
172.16.23.0/24  -     ETH1       kernel

[ok][2021-03-04 11:41:42]
```

3. Following command shows IPsec SA status

```
admin@MCR-MC7455 11:53:32> show services vpn
services vpn ike security-associations security-association DMVPN-2_t4
 state          ESTABLISHED
 local-host     172.16.23.22
 local-id       172.16.23.22
 remote-host    172.16.23.24
 remote-id      172.16.23.24
 initiator      true
 initiator-spi  20bf5b4a7e0ca24b
 responder-spi  babfd7d125541ec7
```

```
 ciphersuite     AES_CBC-128/HMAC_SHA2_256_128/PRF_HMAC_SHA2_256/MODP_2048
 established-time 43
 rekey-time      10092
 reauth-time     14919016
services vpn ike security-associations security-association DMVPN_t4
 state           ESTABLISHED
 local-host      192.168.3.22
 local-id        192.168.3.22
 remote-host     192.168.3.24
 remote-id       192.168.3.24
 initiator       true
 initiator-spi   9524e8fdf7daa1a2
 responder-spi   baa6d8987fdeddf8
 ciphersuite     AES_CBC-128/HMAC_SHA2_256_128/PRF_HMAC_SHA2_256/MODP_2048
 established-time 16
 rekey-time      10094
 reauth-time     6777953
services vpn ipsec security-associations security-association DMVPN-2_t4-1
 state       INSTALLED
 mode        TRANSPORT
 udp-encap   false
 in-spi      cfb1ab5c
 out-spi     cf1479ab
 ciphersuite AES_CBC-128/HMAC_SHA2_256_128
 in-bytes    524
 in-packets  6
 in-last-use 0
 out-bytes   196
 out-packets 2
 out-last-use 40
 rekey-time  9795
 life-time   10755
 install-time 46
 local-ts    172.16.23.22/32[gre]
 remote-ts   172.16.23.24/32[gre]
services vpn ipsec security-associations security-association DMVPN_t4-1
 state       INSTALLED
 mode        TRANSPORT
 udp-encap   false
 in-spi      cd3afd49
 out-spi     c8c391f2
 ciphersuite AES_CBC-128/HMAC_SHA2_256_128
 in-bytes    990
 in-packets  10
 in-last-use 5
 out-bytes   1001
 out-packets 12
 out-last-use 5
 rekey-time  9877
 life-time   10782
 install-time 18
 local-ts    192.168.3.22/32[gre]
 remote-ts   192.168.3.24/32[gre]
```

```
[ok][2021-03-04 11:53:38]
admin@MCR-MC7455 11:53:38>
```

4. Following command shows NHRP status

```
admin@MCR-MC7455 11:53:38> show services nhrp
UNDERLAY              PROTOCOL                    EXPIRES
VRF      NBMA ADDRESS  ADDRESS       STATE    TYPE    IN
-------------------------------------------------------------------
-        0.0.0.0      172.16.1.22/32  up      local
-        0.0.0.0      172.16.2.22/32  up      local
-        0.0.0.0      172.16.1.22/32  up      local
-        0.0.0.0      172.16.2.22/32  up      local
-        0.0.0.0      10.1.1.24/32    up      local   2:41
-        192.168.3.24  172.16.1.24/24  used up  static
-        172.16.23.24  172.16.2.24/24  up      static

[ok][2021-03-04 11:53:50]
```

5. Following command shows main/global routing table, which includes default route obtained from the HUB router over the GRE_DMVPN_t4-2 tunnel.

```
admin@MCR-MC7455 12:04:02> show routing-state routes
                    OUTGOING
DEST PREFIX    NEXT HOP     INTERFACE      SOURCE
-----------------------------------------------------
0.0.0.0/0      172.16.2.24  GRE_DMVPN_t4-2  dynamic
172.16.1.0/24  -            GRE_DMVPN_t4    kernel
172.16.2.0/24  -            GRE_DMVPN_t4-2  kernel
fe80::/64      -                    kernel

[ok][2021-03-04 12:04:13]
```

DMVPN HUB CONFIGURATION

This section describes corresponding configuration on the Orbit setup as a DMVPN HUB router. The default route is advertised with longer AS-PATH over Cell than on ETH1 interface, so when both tunnels are up, the tunnel over ETH1 will be preferred. **Note**: This configuration does not use VRFs.

```
set interfaces interface Bridge type bridge
set interfaces interface Bridge bridge-settings members port ETH1 port-path-cost 1500
set interfaces interface Bridge bridge-settings members port ETH2 port-path-cost 1500
set interfaces interface Bridge bridge-settings members port ETH3 port-path-cost 1500
set interfaces interface Bridge bridge-settings members port ETH4 port-path-cost 1500
set interfaces interface Bridge filter input IN_TRUSTED
set interfaces interface Bridge filter output OUT_TRUSTED
set interfaces interface Bridge ipv4
set interfaces interface Bridge ipv4 address 172.16.23.24 prefix-length 24
set interfaces interface Cell type cellular
set interfaces interface Cell cell-config
set interfaces interface Cell cell-config connection-profile PROFILE-1 bearer-config protocol ip
set interfaces interface Cell filter input IN_UNTRUSTED
```

set interfaces interface Cell filter output OUT_UNTRUSTED
set interfaces interface Cell nat source MASQ
set interfaces interface Cell ipv4
set interfaces interface Cell ipv4 dhcp
set interfaces interface Cell ipv4 dhcp request-routers true
set interfaces interface Cell ipv4 dhcp point-to-point-connection true
set interfaces interface ETH1 type ethernet
set interfaces interface ETH2 type ethernet
set interfaces interface ETH3 type ethernet
set interfaces interface ETH4 type ethernet
set interfaces interface GRE_DMVPN_t4 type gre
set interfaces interface GRE_DMVPN_t4 enabled true
set interfaces interface GRE_DMVPN_t4 gre-config
set interfaces interface GRE_DMVPN_t4 gre-config mode ip-over-gre
set interfaces interface GRE_DMVPN_t4 gre-config src-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4 gre-config dst-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4 gre-config key 1000
set interfaces interface GRE_DMVPN_t4 gre-config ipsec-connection DMVPN_t4
set interfaces interface GRE_DMVPN_t4 filter input IN_TRUSTED
set interfaces interface GRE_DMVPN_t4 filter output OUT_TRUSTED
set interfaces interface GRE_DMVPN_t4 ipv4
set interfaces interface GRE_DMVPN_t4 ipv4 mtu 1300
set interfaces interface GRE_DMVPN_t4 ipv4 address 172.16.1.24 prefix-length 24
set interfaces interface GRE_DMVPN_t4-2 type gre
set interfaces interface GRE_DMVPN_t4-2 enabled true
set interfaces interface GRE_DMVPN_t4-2 gre-config
set interfaces interface GRE_DMVPN_t4-2 gre-config mode ip-over-gre
set interfaces interface GRE_DMVPN_t4-2 gre-config src-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4-2 gre-config dst-address 0.0.0.0
set interfaces interface GRE_DMVPN_t4-2 gre-config key 1001
set interfaces interface GRE_DMVPN_t4-2 gre-config ipsec-connection DMVPN-2_t4
set interfaces interface GRE_DMVPN_t4-2 filter input IN_TRUSTED
set interfaces interface GRE_DMVPN_t4-2 filter output OUT_TRUSTED
set interfaces interface GRE_DMVPN_t4-2 ipv4
set interfaces interface GRE_DMVPN_t4-2 ipv4 mtu 1300
set interfaces interface GRE_DMVPN_t4-2 ipv4 address 172.16.2.24 prefix-length 24
set interfaces interface LO1 type loopback
set interfaces interface LO1 enabled true
set interfaces interface LO1 ipv4
set interfaces interface LO1 ipv4 address 10.1.1.24 prefix-length 32

set services nhrp enabled true
set services nhrp interface GRE_DMVPN_t4 authentication
8r0bbbdRZfzNRtAuv0LMdsAQ4kNhwaPWpEPGC5fsAdDI=
set services nhrp interface GRE_DMVPN_t4-2 authentication
8dPURRFO4UOXm2XTNs7t4+KLsYSv7JHMH8RsH/FGpU0c=

set services vpn enabled true
set services vpn ike
set services vpn ike policy DMVPN-2_t4_ike_policy version ikev2
set services vpn ike policy DMVPN-2_t4_ike_policy auth-method pre-shared-key
set services vpn ike policy DMVPN-2_t4_ike_policy pre-shared-key
8Haa+vSrXiZmTUGvB2cJPsGqJ2u6u72fBOhQYLEU30LM=

set services vpn ike policy DMVPN-2_t4_ike_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ike policy DMVPN-2_t4_ike_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ike policy DMVPN-2_t4_ike_policy ciphersuite CS1 dh-group dh14
set services vpn ike policy DMVPN-2_t4_ike_policy life-time 180
set services vpn ike policy DMVPN-2_t4_ike_policy reauth false
set services vpn ike policy DMVPN-2_t4_ike_policy mobike true
set services vpn ike policy DMVPN_t4_ike_policy version ikev2
set services vpn ike policy DMVPN_t4_ike_policy auth-method pre-shared-key
set services vpn ike policy DMVPN_t4_ike_policy pre-shared-key
8Ir2YD+v64EMNutkwIk7K4Yp+/H41XUn08G2DqyfPzE4=
set services vpn ike policy DMVPN_t4_ike_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ike policy DMVPN_t4_ike_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ike policy DMVPN_t4_ike_policy ciphersuite CS1 dh-group dh14
set services vpn ike policy DMVPN_t4_ike_policy life-time 180
set services vpn ike policy DMVPN_t4_ike_policy reauth false
set services vpn ike policy DMVPN_t4_ike_policy mobike true
set services vpn ike peer DMVPN-2_t4_ike_peer ike-policy DMVPN-2_t4_ike_policy
set services vpn ike peer DMVPN-2_t4_ike_peer local-endpoint any
set services vpn ike peer DMVPN-2_t4_ike_peer local-identity default
set services vpn ike peer DMVPN-2_t4_ike_peer peer-endpoint address 172.16.23.22
set services vpn ike peer DMVPN-2_t4_ike_peer peer-identity default
set services vpn ike peer DMVPN-2_t4_ike_peer role responder
set services vpn ike peer DMVPN-2_t4_ike_peer dpd-enabled true
set services vpn ike peer DMVPN-2_t4_ike_peer dpd-interval 300
set services vpn ike peer DMVPN_t4_ike_peer ike-policy DMVPN_t4_ike_policy
set services vpn ike peer DMVPN_t4_ike_peer local-endpoint any
set services vpn ike peer DMVPN_t4_ike_peer local-identity default
set services vpn ike peer DMVPN_t4_ike_peer peer-endpoint address 192.168.3.22
set services vpn ike peer DMVPN_t4_ike_peer peer-identity default
set services vpn ike peer DMVPN_t4_ike_peer role responder
set services vpn ike peer DMVPN_t4_ike_peer dpd-enabled true
set services vpn ike peer DMVPN_t4_ike_peer dpd-interval 300
set services vpn ipsec
set services vpn ipsec policy DMVPN-2_t4_ipsec_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ipsec policy DMVPN-2_t4_ipsec_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ipsec policy DMVPN-2_t4_ipsec_policy life-time 60
set services vpn ipsec policy DMVPN_t4_ipsec_policy ciphersuite CS1 encryption-algo aes128-cbc
set services vpn ipsec policy DMVPN_t4_ipsec_policy ciphersuite CS1 mac-algo sha256-hmac
set services vpn ipsec policy DMVPN_t4_ipsec_policy life-time 60
set services vpn ipsec connection DMVPN-2_t4 ike-peer DMVPN-2_t4_ike_peer
set services vpn ipsec connection DMVPN-2_t4 ipsec-policy DMVPN-2_t4_ipsec_policy
set services vpn ipsec connection DMVPN-2_t4 host-to-host
set services vpn ipsec connection DMVPN-2_t4 failure-retry-interval 5
set services vpn ipsec connection DMVPN-2_t4 filter input IN_TRUSTED
set services vpn ipsec connection DMVPN-2_t4 filter output OUT_TRUSTED
set services vpn ipsec connection DMVPN_t4 ike-peer DMVPN_t4_ike_peer
set services vpn ipsec connection DMVPN_t4 ipsec-policy DMVPN_t4_ipsec_policy
set services vpn ipsec connection DMVPN_t4 host-to-host
set services vpn ipsec connection DMVPN_t4 failure-retry-interval 5
set services vpn ipsec connection DMVPN_t4 filter input IN_TRUSTED
set services vpn ipsec connection DMVPN_t4 filter output OUT_TRUSTED

set routing router-id 10.1.1.24

```
set routing route-filter DEFAULT-OVER-CELL rule 1 match dest-prefix 0.0.0.0/0
set routing route-filter DEFAULT-OVER-CELL rule 1 actions action accept
set routing route-filter DEFAULT-OVER-CELL rule 1 actions set bgp-as-path 65000
set routing route-filter DEFAULT-OVER-LAN rule 1 match dest-prefix 0.0.0.0/0
set routing route-filter DEFAULT-OVER-LAN rule 1 actions action accept
set routing bgp
set routing bgp neighbor SPOKE-CELL peer-address 172.16.1.22
set routing bgp neighbor SPOKE-CELL local-address 172.16.1.24
set routing bgp neighbor SPOKE-CELL enabled true
set routing bgp neighbor SPOKE-CELL export-filter DEFAULT-OVER-CELL
set routing bgp neighbor SPOKE-CELL passive false
set routing bgp neighbor SPOKE-CELL local-as 65000
set routing bgp neighbor SPOKE-CELL peer-as 65001
set routing bgp neighbor SPOKE-CELL next-hop-self false
set routing bgp neighbor SPOKE-CELL hold-time 240
set routing bgp neighbor SPOKE-CELL keepalive-time 80
set routing bgp neighbor SPOKE-CELL connect-retry-time 120
set routing bgp neighbor SPOKE-LAN peer-address 172.16.2.22
set routing bgp neighbor SPOKE-LAN local-address 172.16.2.24
set routing bgp neighbor SPOKE-LAN enabled true
set routing bgp neighbor SPOKE-LAN export-filter DEFAULT-OVER-LAN
set routing bgp neighbor SPOKE-LAN passive false
set routing bgp neighbor SPOKE-LAN local-as 65000
set routing bgp neighbor SPOKE-LAN peer-as 65001
set routing bgp neighbor SPOKE-LAN next-hop-self false
set routing bgp neighbor SPOKE-LAN hold-time 30
set routing bgp neighbor SPOKE-LAN keepalive-time 10
set routing bgp neighbor SPOKE-LAN connect-retry-time 30

set services firewall enabled true
set services firewall address-set CELL-IP
set services firewall filter IN_TRUSTED rule 10 match protocol all
set services firewall filter IN_TRUSTED rule 10 actions
set services firewall filter IN_TRUSTED rule 10 actions action accept
set services firewall filter IN_UNTRUSTED rule 1 match protocol icmp
set services firewall filter IN_UNTRUSTED rule 1 actions
set services firewall filter IN_UNTRUSTED rule 1 actions action accept
set services firewall filter IN_UNTRUSTED rule 2 match protocol udp
set services firewall filter IN_UNTRUSTED rule 2 match src-port
set services firewall filter IN_UNTRUSTED rule 2 match src-port services [ dns ]
set services firewall filter IN_UNTRUSTED rule 3 match protocol tcp
set services firewall filter IN_UNTRUSTED rule 3 match dst-port
set services firewall filter IN_UNTRUSTED rule 3 match dst-port services [ http https netconf ssh ]
set services firewall filter IN_UNTRUSTED rule 4 match protocol udp
set services firewall filter IN_UNTRUSTED rule 4 match dst-port
set services firewall filter IN_UNTRUSTED rule 4 match dst-port services [ ike ]
set services firewall filter IN_UNTRUSTED rule 4 actions
set services firewall filter IN_UNTRUSTED rule 4 actions action accept
set services firewall filter IN_UNTRUSTED rule 5 match protocol esp
set services firewall filter IN_UNTRUSTED rule 5 actions
set services firewall filter IN_UNTRUSTED rule 5 actions action accept
set services firewall filter IN_UNTRUSTED rule 10 match protocol all
set services firewall filter IN_UNTRUSTED rule 10 actions
```

```
set services firewall filter IN_UNTRUSTED rule 10 actions action drop
set services firewall filter OUT_TRUSTED rule 10 match protocol all
set services firewall filter OUT_TRUSTED rule 10 actions
set services firewall filter OUT_TRUSTED rule 10 actions action accept
set services firewall filter OUT_UNTRUSTED rule 1 match protocol gre
set services firewall filter OUT_UNTRUSTED rule 1 actions
set services firewall filter OUT_UNTRUSTED rule 1 actions action drop
set services firewall filter OUT_UNTRUSTED rule 2 match src-address
set services firewall filter OUT_UNTRUSTED rule 2 match src-address address-set CELL-IP
set services firewall filter OUT_UNTRUSTED rule 2 match src-address add-interface-address true
set services firewall filter OUT_UNTRUSTED rule 2 actions
set services firewall filter OUT_UNTRUSTED rule 2 actions action accept
set services firewall filter OUT_UNTRUSTED rule 10 match protocol all
set services firewall filter OUT_UNTRUSTED rule 10 actions
set services firewall filter OUT_UNTRUSTED rule 10 actions action drop
set services firewall nat source rule-set MASQ rule 1 source-nat interface
```

BGP WITHIN VRF

This section describes configuration of BGP operating within VRF and importing/exporting routes from the routing table associated with VRF routing instance. This example builds off the “WAN redundancy on DMVPN spoke” use case, where now, the GRE_DMVPN_t4-2 tunnel interface is moved into its own VRF and BGP is run over that tunnel in this VRF.

SPOKE CONFIG

1. Create VRF3 interface and add GRE_DMVPN_t4-2 interface as a member.

```
set interfaces interface VRF3 type vrf
set interfaces interface VRF3 vrf-config
set interfaces interface VRF3 vrf-config member GRE_DMVPN_t4-2
set interfaces interface VRF3 filter input IN_TRUSTED
set interfaces interface VRF3 filter output OUT_TRUSTED
```

2. Create routing instance named VRF3 and associate it with the VRF3 interface.

```
set routing routing-instance VRF3 interface VRF3
```

3. Configure BGP within VRF3

```
set routing routing-instance VRF3 bgp
set routing routing-instance VRF3 bgp neighbor HUB-LAN peer-address 172.16.2.24
set routing routing-instance VRF3 bgp neighbor HUB-LAN local-address 172.16.2.22
set routing routing-instance VRF3 bgp neighbor HUB-LAN enabled true
set routing routing-instance VRF3 bgp neighbor HUB-LAN export-filter LO1
set routing routing-instance VRF3 bgp neighbor HUB-LAN passive false
set routing routing-instance VRF3 bgp neighbor HUB-LAN local-as 65001
set routing routing-instance VRF3 bgp neighbor HUB-LAN peer-as 65000
set routing routing-instance VRF3 bgp neighbor HUB-LAN next-hop-self false
set routing routing-instance VRF3 bgp neighbor HUB-LAN hold-time 240
set routing routing-instance VRF3 bgp neighbor HUB-LAN keepalive-time 80
set routing routing-instance VRF3 bgp neighbor HUB-LAN connect-retry-time 120
```

STATUS

1. Following command shows BGP status under VRF3.

```
admin@MCR-MC7455 14:27:29> show routing-state routing-instance VRF3 bgp
bgp neighbor HUB-LAN
 state         up
 preference    100
 import-filter  ACCEPT
 export-filter  LO1
 statistics import-updates-received 1
 statistics import-updates-rejected 0
```

```
 statistics import-updates-filtered 0
 statistics import-updates-ignored 0
 statistics import-updates-accepted 1
 statistics import-withdraws-received 0
 statistics import-withdraws-rejected 0
 statistics import-withdraws-ignored 0
 statistics import-withdraws-accepted 0
 statistics export-updates-received 1
 statistics export-updates-rejected 1
 statistics export-updates-filtered 0
 statistics export-updates-accepted 0
 statistics export-withdraws-received 0
 statistics export-withdraws-accepted 0
 local-state    established
 peer-address   172.16.2.24
 peer-as        65000
 peer-id        10.1.1.24
 local-address  172.16.2.22
 hold-time      159/240
 keepalive-time 54/80
[ok][2021-03-04 14:27:36]
```

2. Following command show the routing table within VRF3

```
admin@MCR-MC7455 14:27:26> show routing-state routing-instance VRF3 routes
                      OUTGOING
DEST PREFIX    NEXT HOP     INTERFACE      SOURCE
-------------------------------------------------------
0.0.0.0/0      172.16.2.24  GRE_DMVPN_t4-2  dynamic
172.16.2.0/24  -            GRE_DMVPN_t4-2  kernel

[ok][2021-03-04 14:27:29]
```

3. Following command can be used to ping HUB router's loopback address from within VRF3

```
admin@MCR-MC7455 14:29:46> ping 10.1.1.24 vrf VRF3
PING 10.1.1.24 (10.1.1.24) 56(84) bytes of data.
64 bytes from 10.1.1.24: icmp_seq=1 ttl=64 time=1.69 ms
64 bytes from 10.1.1.24: icmp_seq=2 ttl=64 time=1.95 ms
^C
--- 10.1.1.24 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.695/1.825/1.955/0.130 ms
```

RIP WITHIN VRF

This section describes configuration of RIP operating within VRF and importing/exporting routes from the routing table associated with VRF routing instance. This example builds off the “WAN redundancy on DMVPN spoke” use case, where now, the GRE_DMVPN_t4-2 tunnel interface is moved into its own VRF and RIP is run over that tunnel in this VRF.

SPOKE CONFIG

1. Create VRF3 interface and add GRE_DMVPN_t4-2 interface as a member.

```
set interfaces interface VRF3 type vrf
set interfaces interface VRF3 vrf-config
set interfaces interface VRF3 vrf-config member GRE_DMVPN_t4-2
set interfaces interface VRF3 filter input IN_TRUSTED
set interfaces interface VRF3 filter output OUT_TRUSTED
```

2. Create routing instance named VRF3 and associate it with the VRF3 interface.

```
set routing routing-instance VRF3 interface VRF3
```

3. Configure RIP within VRF3

```
set routing routing-instance VRF3 rip
set routing routing-instance VRF3 rip enabled true
set routing routing-instance VRF3 rip export-filter LO1
set routing routing-instance VRF3 rip interface GRE_DMVPN_t4-2
```

HUB CONFIG

```
set routing rip
set routing rip enabled true
set routing rip export-filter DEFAULT-OVER-LAN
set routing rip interface GRE_DMVPN_t4-2
```

STATUS

1. Following command show RIP status within VRF3

```
admin@MCR-MC7455 15:14:50> show routing-state routing-instance VRF3 rip
rip routing-instance VRF3
rip state       up
rip preference   120
rip import-filter ACCEPT
rip export-filter LO1
rip statistics import-updates-received 1
```

```
rip statistics import-updates-rejected 0
rip statistics import-updates-filtered 0
rip statistics import-updates-ignored 0
rip statistics import-updates-accepted 1
rip statistics import-withdraws-received 0
rip statistics import-withdraws-rejected 0
rip statistics import-withdraws-ignored 0
rip statistics import-withdraws-accepted 0
rip statistics export-updates-received 1
rip statistics export-updates-rejected 0
rip statistics export-updates-filtered 1
rip statistics export-updates-accepted 0
rip statistics export-withdraws-received 0
rip statistics export-withdraws-accepted 0
```

2. Following command shows the routing table within VRF3

```
admin@MCR-MC7455 15:15:00> show routing-state routing-instance VRF3 routes
                    OUTGOING
DEST PREFIX    NEXT HOP     INTERFACE      SOURCE
-----------------------------------------------------------
0.0.0.0/0      172.16.2.24  GRE_DMVPN_t4-2  dynamic
172.16.2.0/24  -            GRE_DMVPN_t4-2  kernel
```

3. Following command can be used to ping HUB router's loopback address from within VRF3

```
admin@MCR-MC7455 15:15:31> ping 10.1.1.24 vrf VRF3
PING 10.1.1.24 (10.1.1.24) 56(84) bytes of data.
64 bytes from 10.1.1.24: icmp_seq=1 ttl=64 time=1.75 ms
64 bytes from 10.1.1.24: icmp_seq=2 ttl=64 time=2.04 ms
^C
--- 10.1.1.24 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.759/1.903/2.048/0.150 ms
```

OSPF WITHIN VRF

This section describes configuration of OSPFv2 operating within VRF and importing/exporting routes from the routing table associated with VRF routing instance. This example builds off the “WAN redundancy on DMVPN spoke” use case, where now, the GRE_DMVPN_t4-2 tunnel interface is moved into its own VRF and OSPFv2 is run over that tunnel in this VRF.

SPOKE CONFIG

1. Create VRF3 interface and add GRE_DMVPN_t4-2 interface as a member.

```
set interfaces interface VRF3 type vrf
set interfaces interface VRF3 vrf-config
set interfaces interface VRF3 vrf-config member GRE_DMVPN_t4-2
set interfaces interface VRF3 filter input IN_TRUSTED
set interfaces interface VRF3 filter output OUT_TRUSTED
```

2. Create routing instance named VRF3 and associate it with the VRF3 interface.

```
set routing routing-instance VRF3 interface VRF3
```

3. Configure OSPFv2 within VRF3

```
set routing routing-instance VRF3 ospf enabled true
set routing routing-instance VRF3 ospf export-filter LO1
set routing routing-instance VRF3 ospf area 0.0.0.0 interface GRE_DMVPN_t4-2
```

HUB CONFIG

```
set routing ospf enabled true
set routing ospf export-filter DEFAULT-OVER-LAN
set routing ospf area 0.0.0.0 interface GRE_DMVPN_t4-2
```

STATUS

1. Following command shows OSPFv2 status within VRF3

```
admin@MCR-MC7455 15:26:09> show routing-state routing-instance VRF3 ospf

ospf routing-instance VRF3
ospf state      up
ospf preference  150
ospf import-filter ACCEPT
ospf export-filter LO1
ospf statistics import-updates-received 3
```

```
ospf statistics import-updates-rejected 0
ospf statistics import-updates-filtered 0
ospf statistics import-updates-ignored 0
ospf statistics import-updates-accepted 3
ospf statistics import-withdraws-received 1
ospf statistics import-withdraws-rejected 0
ospf statistics import-withdraws-ignored 0
ospf statistics import-withdraws-accepted 1
ospf statistics export-updates-received 3
ospf statistics export-updates-rejected 3
ospf statistics export-updates-filtered 0
ospf statistics export-updates-accepted 0
ospf statistics export-withdraws-received 1
ospf statistics export-withdraws-accepted 0
ospf area 0.0.0.0
 stub                false
 nssa                false
 transit             false
 nssa-translation    false
 num-interfaces      1
 num-neighbors       1
 num-adjacent-neighbors 1
ospf interface GRE_DMVPN_t4-2
 virtual-link              false
 network-type              bcast
 area-id                   0.0.0.0
 state                     bdr
 priority                  1
 cost                      10
 hello-interval            10
 wait-interval             40
 dead-interval             40
 retransmit-interval       5
 designated-router-id      0.0.0.0
 designated-router-address 0.0.0.0
 backup-designated-router-id 0.0.0.0
 backup-designated-router-address 0.0.0.0
                                    DEAD
ID       ADDRESS     INTERFACE      STATE    PRIORITY  TIME
-------------------------------------------------------------------
10.1.1.24  172.16.2.24  GRE_DMVPN_t4-2  Full/DR  1       34

                        ADV
SCOPE       TYPE  LS ID       ROUTER     AGE  SEQUENCE  CHECKSUM
-------------------------------------------------------------------------
Global      0005  0.0.0.0     10.1.1.24  200  80000008  db94
Area 0.0.0.0  0001  10.1.1.22   10.1.1.22  12   80000002  c78f
Area 0.0.0.0  0002  172.16.2.24  10.1.1.24  13   80000001  ff19
Area 0.0.0.0  0001  10.1.1.24   10.1.1.24  13   80000009  b594
```

2. Following command shows the routing table within VRF3

```
admin@MCR-MC7455 15:26:14> show routing-state routing-instance VRF3 routes

                    OUTGOING
DEST PREFIX    NEXT HOP     INTERFACE      SOURCE
-----------------------------------------------------
0.0.0.0/0      172.16.2.24  GRE_DMVPN_t4-2  dynamic
172.16.2.0/24  -            GRE_DMVPN_t4-2  kernel
```

3. Following command can be used to ping HUB router's loopback address from within VRF3

```
admin@MCR-MC7455 15:26:59> ping 10.1.1.24 vrf VRF3
PING 10.1.1.24 (10.1.1.24) 56(84) bytes of data.
64 bytes from 10.1.1.24: icmp_seq=1 ttl=64 time=1.75 ms
64 bytes from 10.1.1.24: icmp_seq=2 ttl=64 time=1.94 ms
^C
--- 10.1.1.24 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.759/1.852/1.946/0.102 ms
```

VLANS IN RESPECTIVE VRFS

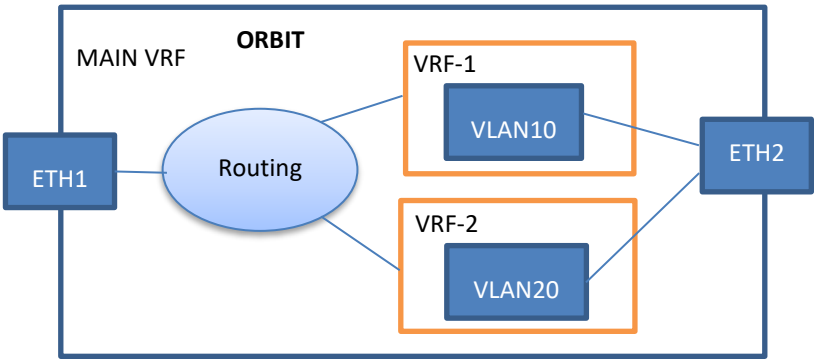

Figure 3 VLANS in respective VRFs

VLANs allow isolation of traffic at layer-2. The traffic can be routed between VLANs at layer-3 at an IP router through which the VLANs either transit or terminate. However, it may be desired to prevent routing by default across VLANs at such a router thereby isolating them at layer-3 as well. This can be done, by putting each VLAN in its respective VRF. Also, per-VLAN default gateway can be configured by adding a default route in the respective VRF.

CONFIG

1. Configure vlan 1

```
set interfaces interface VLAN1 type vlan
set interfaces interface VLAN1 vlan-config vlan-id 1
set interfaces interface VLAN1 filter input IN_TRUSTED
set interfaces interface VLAN1 filter output OUT_TRUSTED
set interfaces interface VLAN1 ipv4 address 172.16.24.22 prefix-length 24
```

2. Configure vlan 2

```
set interfaces interface VLAN2 type vlan
set interfaces interface VLAN2 vlan-config vlan-id 2
set interfaces interface VLAN2 filter input IN_TRUSTED
set interfaces interface VLAN2 filter output OUT_TRUSTED
set interfaces interface VLAN2 ipv4 address 172.16.25.22 prefix-length 24
```

3. Configure ETH2 as trunk port carrying both vlans 1 and 2.

```
delete interfaces interface Bridge bridge-settings members port ETH2
set interfaces interface ETH2 type ethernet
set interfaces interface ETH2 vlan-mode trunk
set interfaces interface ETH2 vlans [ VLAN1 VLAN2 ]
```

4. Create VRF1 and add VLAN1 to it

```
set interfaces interface VRF1 type vrf
set interfaces interface VRF1 vrf-config member VLAN1
set interfaces interface VRF1 filter input IN_UNTRUSTED
```

```
set interfaces interface VRF1 filter output OUT_TRUSTED
```

5. Create VRF2 and add VLAN2 to it

```
set interfaces interface VRF2 type vrf
set interfaces interface VRF2 vrf-config member VLAN2
set interfaces interface VRF2 filter input IN_UNTRUSTED
set interfaces interface VRF2 filter output OUT_TRUSTED
```

6. Create routing instance VRF1, associate it with VRF1 interface and add default route to it over VLAN1.

```
set routing routing-instance VRF1 interface VRF1
set routing routing-instance VRF1 static-routes ipv4 route 1 outgoing-interface VLAN1
set routing routing-instance VRF1 static-routes ipv4 route 1 dest-prefix 0.0.0.0/0
```

7. Create routing instance VRF2, associated it with VRF2 interface and add default route to it over VLAN2.

```
set routing routing-instance VRF2 interface VRF2
set routing routing-instance VRF2 static-routes ipv4 route 1 outgoing-interface VLAN2
set routing routing-instance VRF2 static-routes ipv4 route 1 dest-prefix 0.0.0.0/0
```

STATUS

1. Following command shows routes in VRF1

```
admin@MCR-MC7455 17:46:45> show routing-state routing-instance VRF1 routes
            NEXT  OUTGOING
DEST PREFIX     HOP   INTERFACE  SOURCE
-----------------------------------------
0.0.0.0/0       -     VLAN1      static
172.16.24.0/24  -     VLAN1      kernel
```

2. Following command shows routes in VRF2

```
admin@MCR-MC7455 17:46:57> show routing-state routing-instance VRF2 routes
            NEXT  OUTGOING
DEST PREFIX     HOP   INTERFACE  SOURCE
-----------------------------------------
0.0.0.0/0       -     VLAN2      static
172.16.25.0/24  -     VLAN2      kernel
```

3. Following command can be used to verify connectivity to peer router setup with similar VLAN configuration.

```
admin@MCR-MC7455 17:49:26> ping 172.16.24.24 vrf VRF1
PING 172.16.24.24 (172.16.24.24) 56(84) bytes of data.
64 bytes from 172.16.24.24: icmp_seq=1 ttl=64 time=2.01 ms
^C
--- 172.16.24.24 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 2.017/2.017/2.017/0.000 ms
```

4. Following command can be used to verify connectivity to peer router setup with similar VLAN configuration.

```
admin@MCR-MC7455 17:49:30> ping 172.16.25.24 vrf VRF2
PING 172.16.25.24 (172.16.25.24) 56(84) bytes of data.
64 bytes from 172.16.25.24: icmp_seq=1 ttl=64 time=1.82 ms
64 bytes from 172.16.25.24: icmp_seq=2 ttl=64 time=1.03 ms
^C
--- 172.16.25.24 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.036/1.430/1.824/0.394 ms
```

VRF AWARE CLI COMMANDS

Following CLI commands can be executed within a VRF by passing VRF interface as an option.

1. ping <host> vrf <VRF-INTERFACE>

   Example:
   admin@MCR-MC7455 15:29:49> ping 10.1.1.24 vrf VRF3
   PING 10.1.1.24 (10.1.1.24) 56(84) bytes of data.
   64 bytes from 10.1.1.24: icmp_seq=1 ttl=64 time=1.72 ms
   ^C
   --- 10.1.1.24 ping statistics ---
   1 packets transmitted, 1 received, 0% packet loss, time 0ms
   rtt min/avg/max/mdev = 1.725/1.725/1.725/0.000 ms

2. telnet <host> vrf <VRF-INTERFACE>

   Example:
   admin@MCR-MC7455 15:31:50> telnet 172.16.23.7 vrf VRF3
   Trying 172.16.23.7...
   [ok][2021-03-04 15:32:45]

3. ssh <host> --vrf <VRF-INTERFACE>

   Example:
   admin@MCR-MC7455 15:35:32> ssh admin@10.1.1.24 --vrf VRF3
   FIPS mode disabled
   admin@10.1.1.24's password:

4. traceroute <host> --vrf <VRF-INTERFACE>

   Example:

   admin@MCR-MC7455 15:33:25> traceroute 172.16.23.7 --vrf VRF2

   1?: [LOCALHOST] pmtu 1500
   1: 172.16.23.7 0.919ms reached
   1: 172.16.23.7 0.903ms reached
   Resume: pmtu 1500 hops 1 back 64

5. iperf <options> --vrf <VRF_INTERFACE>

   Example:

   admin@MCR-MC7455 15:36:26> iperf -c 10.1.1.24 -f k -i 1 --vrf VRF3

   Client connecting to 10.1.1.24, TCP port 5001
   TCP window size: 43.8 KByte (default)

```
[  4] local 172.16.2.22 port 54630 connected with 10.1.1.24 port 5001
[ ID] Interval       Transfer     Bandwidth
[  4]  0.0- 1.0 sec  3200 KBytes  26214 Kbits/sec
[  4]  1.0- 2.0 sec  3328 KBytes  27263 Kbits/sec
[  4]  0.0- 2.6 sec  8320 KBytes  26403 Kbits/sec
```

IMPORTANT NOTES

1. The incoming traffic on any member of a VRF device is subjected to filter applied to that VRF (and not to the filter applied to that member device). Therefore, if there is any UNTRUSTED interface in the VRF, IN_UNTRUSTED filter must be applied to the VRF device as the input filter. It is recommended same filters be applied to the VRF parent device as applied to the member device (in the input and output direction respectively).

2. The destination NAT (port forwarding) rule set must be applied to the VRF parent if it is applied to the member interface. The route must exist in the current VRF for the port forwarded traffic if it has to egress out of an interface that is outside of the current VRF.