EVPN for VXLAN Tunnels (Layer 3)

In This Chapter

This section provides information about EVPN for VXLAN tunnels (Layer 3).

Topics in this section include:

Applicability

This example is applicable to the 7950 XRS, 7750 SR-c4/c12, 7750 SR-7/12 and 7450 ESS-6/6v/7/12, but it is not supported on 7750 SR-1, 7450 ESS-1 or 7710 SR. Virtual eXtensible Local Area Network (VXLAN) requires IOM3-XP/IMM or higher-based line cards and chassis-mode D. Ethernet Virtual Private Networks (EVPN) is a control plane technology and does not have line card hardware dependencies.

The configuration was tested in release 12.0.R4. The EVPN for VXLAN Tunnels (Layer 2) on page 281 example is pre-requisite reading.

Overview

As discussed in the EVPN for VXLAN Tunnels (Layer 2) on page 281 example, EVPN and VXLAN can be enabled on VPLS or R-VPLS services in SR OS. While that example focuses on the use of EVPN-VXLAN layer 2 services, that is how EVPN-VXLAN is configured in VPLS services, this example describes how EVPN-VXLAN can be used to provide inter-subnet forwarding in R-VPLS and VPRN services. Inter-subnet forwarding can be provided by regular R-VPLS and VPRN services, however EVPN provides an efficient and unified way to populate FDBs (Forwarding Data Bases), ARP (Address Resolution Protocol) tables and routing tables using a single BGP address family. Inter-subnet forwarding in overlay networks would otherwise require data plane learning and the use of routing protocols on a per VPRN basis.

The SR OS solution for inter-subnet forwarding using EVPN is based on building blocks described in draft-sajassi-l2vpn-evpn-inter-subnet-forwarding and the use of the EVPN ip-prefix routes (routes type-5) as explained in draft-rabadan-l2vpn-evpn-prefix-advertisement. This example describes three supported common scenarios and provides the CLI configuration and required tools to troubleshoot EVPN-VXLAN in each case. The scenarios tested and explained are:

- EVPN-VXLAN in R-VPLS services
- EVPN-VXLAN in IRB (Integrated Routing Bridging) Backhaul R-VPLS services
- EVPN-VXLAN in EVPN Tunnel R-VPLS services

In all these scenarios redundant PEs are usually deployed. If that is the case, the interaction of EVPN, IP-VPN and the routing table manager (RTM) may lead to some routing loop situations that must be avoided by the use of routing policies (note that this also may happen in traditional IP-VPN deployments when eBGP and MP-BGP interact to populate VPRN routing tables in multi-homed networks). This section explains when those routing loops can happen and how to avoid them.

The term IRB interface refers to an R-VPLS service bound to a VPRN IP interface. The terms IRB interface and R-VPLS interface are used interchangeably throughout this example.

Configuration

This section describes the configuration of EVPN-VXLAN for Layer 3 services on the 7x50, as well as the available troubleshooting and show commands. The three scenarios described in the overview are analyzed independently.

EVPN-VXLAN in an R-VPLS Service

Figure 47 shows the topology used in the first scenario.

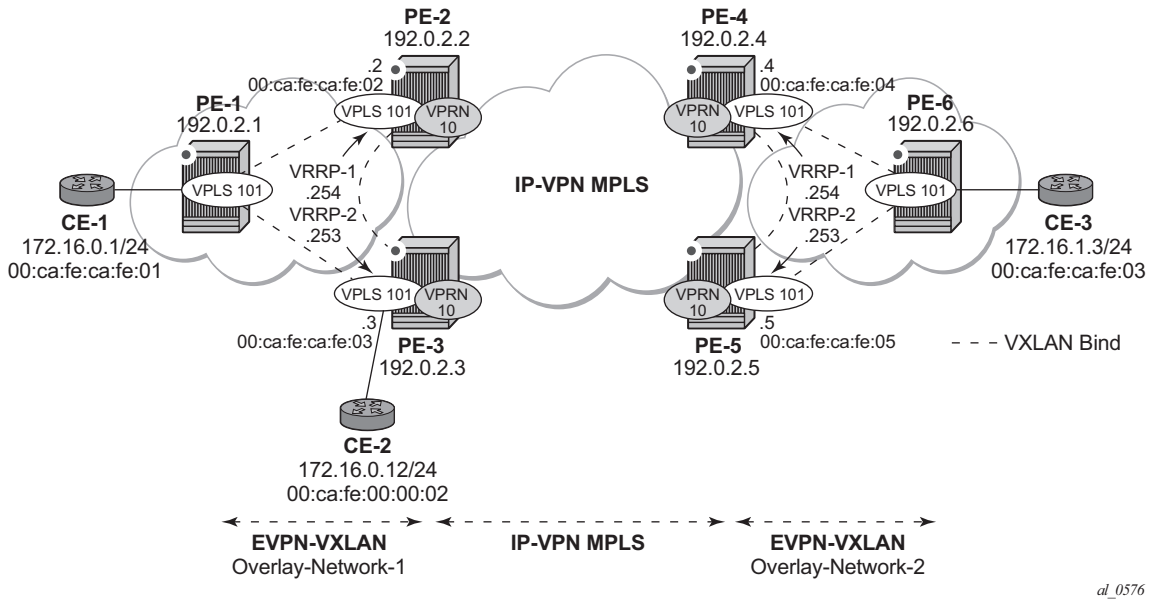

Figure 47: EVPN-VXLAN for R-VPLS Services

The network topology shows two overlay (VXLAN) networks interconnected by an MPLS network:

- PE-1, PE-2 and PE-3 are part of Overlay-Network-1
- PE-4, PE-5 and PE-6 are part of Overlay-Network-2

A Layer 2/Layer 3 service is provided to a customer to connect CE-1, CE-2 and CE-3. In this scenario, Layer 2 connectivity is provided within each overlay network and inter-subnet connectivity (Layer 3) is provided between the overlay networks, hence VPLS 101 is defined within each overlay network and VPRN 10 connects both Layer 2 services through an IP-VPN MPLS network.

Note that the above topology can illustrate a Data Center Interconnect (DCI) example, where Overlay-Network-1 and Overlay-Network-2 are two data centers interconnected through an MPLS WAN. In this application, CE-1, CE-2 and CE-3 would simulate virtual machines or appliances, PE-2/3/4/5 would act as Data Center gateways (DC GW) GWs and PE-1/6 as Network Virtualization Edge devices (or virtual PEs running on a compute infrastructure).

The following protocols and objects are configured beforehand:

- The ports interconnecting the six PEs in Figure 47 are configured as network ports (or hybrid) and will have router network interfaces defined in them. Only the ports connected to the CEs are configured as access ports.
- The six PEs are running IS-IS for the global routing table with the four core PEs interconnected using IS-IS Level-2 point-to-point interfaces and each overlay network using IS-IS Level-1 point-to-point interfaces.
- LDP is used as the MPLS protocol to signal transport tunnel labels among PE-2, PE-3, PE-4 and PE-5. There is no LDP running within each overlay network.
- Note that the network port MTU (in all the ports sending/receiving VXLAN packets) must be at least 50-bytes (54 if dot1q encapsulation is used) greater than the service-mtu in order to accommodate the size of the VXLAN header.

Once the IGP infrastructure and LDP in the core are enabled, BGP has to be configured. In this scenario, two BGP families have to be enabled: EVPN within each overlay-network for the exchange of MAC/IP addresses and setting up the flooding domains, and VPN-IPv4 among the four core PEs so that IP-prefixes can be exchanged and resolved to MPLS tunnels in the core.

As an example, the following CLI output shows the relevant BGP configuration of PE-1, which only needs the EVPN family. PE-6 has a similar BGP configuration, that is, only EVPN family is configured for its peers. Note that the use of Route-Reflectors (RRs) in these type of scenarios is common. Although this scenario does not use RRs, an EVPN RR could have been used in Overlay-Network-1 and Overlay-Network-2 and a separate VPN-IPv4 RR could have been used in the core IP-VPN MPLS network.

```
A:PE-1>config>router>bgp# info
----------------------------------------------
            vpn-apply-import
            vpn-apply-export
            enable-peer-tracking
            rapid-withdrawal
            rapid-update evpn
            group "DC"
                family evpn
                type internal
                neighbor 192.0.2.2
                exit
                neighbor 192.0.2.3
                exit
            exit
            no shutdown
----------------------------------------------
```

The BGP configuration of PE-2 and PE-3 follows (PE-4 and PE-5 have an equivalent configuration).

```
A:PE-2>config>router>bgp# info
----------------------------------------------
            vpn-apply-import
            vpn-apply-export
            min-route-advertisement 1
            enable-peer-tracking
            rapid-withdrawal
            rapid-update evpn
            group "DC"
                family vpn-ipv4 evpn
                type internal
                neighbor 192.0.2.1
                exit
                neighbor 192.0.2.3
                exit
            exit
            group "WAN"
                family vpn-ipv4
                type internal
                neighbor 192.0.2.4
                exit
                neighbor 192.0.2.5
                exit
            exit
            no shutdown
----------------------------------------------
A:PE-3>config>router>bgp# info
----------------------------------------------
            vpn-apply-import
            vpn-apply-export
            min-route-advertisement 1
            enable-peer-tracking
            rapid-withdrawal
            rapid-update evpn
            group "DC"
                family vpn-ipv4 evpn
                type internal
                neighbor 192.0.2.1
                exit
                neighbor 192.0.2.2
                exit
            exit
            group "WAN"
                family vpn-ipv4
                type internal
                neighbor 192.0.2.4
                exit
                neighbor 192.0.2.5
                exit
            exit
            no shutdown
----------------------------------------------
```

Figure 48 shows the BGP peering sessions among the PEs and the enabled BGP families. Note that PE-1 and PE-6 only establish an EVPN peering session with their peers (only the EVPN family is enabled on both PEs, even if the peer PEs are VPN-IPv4 capable as well).

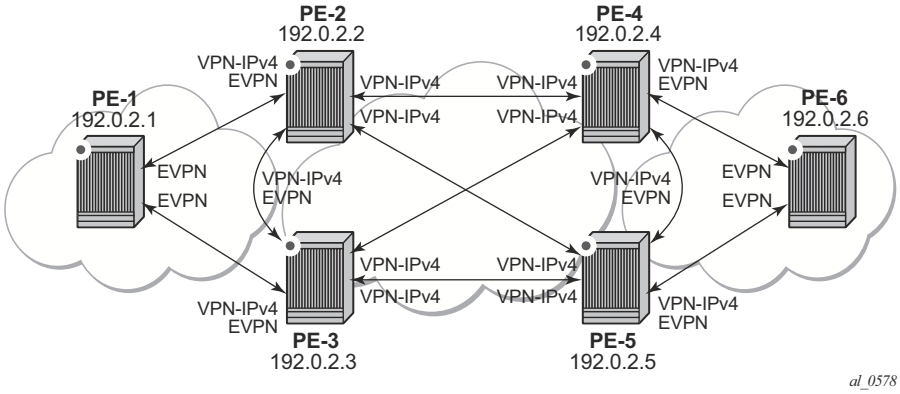

Figure 48: BGP adjacencies and enabled families

Once the network infrastructure is running properly, the actual service configuration, as illustrated in Figure 47, can be carried out. The following CLI output shows the configuration for VPLS 101 and VPRN 10 in PE-1, PE-2 and PE-3. The other overlay network has a similar configuration.
*A:PE-1# configure service vpls 101

```
*A:PE-1>config>service>vpls# info
----------------------------------------------
            vxlan vni 101 create
            exit
            bgp
                route-distinguisher 192.0.2.1:101
                route-target export target:64500:101 import target:64500:101
            exit
            bgp-evpn
                vxlan
                    no shutdown
                exit
            exit
            proxy-arp
                no shutdown
            exit
            stp
                shutdown
            exit
            service-name "evi-101"
            sap 1/1/1:101 create
            exit
            no shutdown
----------------------------------------------
```

```
*A:PE-2# configure service vpls 101
*A:PE-2>config>service>vpls# info
----------------------------------------------
            allow-ip-int-binding
            vxlan vni 101 create
            exit
            bgp
                route-distinguisher 192.0.2.2:101
                route-target export target:64500:101 import target:64500:101
            exit
            bgp-evpn
                vxlan
                    no shutdown
                exit
            exit
            proxy-arp
                shutdown
            exit
            stp
                shutdown
            exit
            service-name "evi-101"
            no shutdown
----------------------------------------------

*A:PE-2# configure service vprn 10
*A:PE-2>config>service>vprn# info
----------------------------------------------
            ecmp 2
            route-distinguisher 192.0.2.2:10
            auto-bind mpls
            vrf-target target:64500:10
            interface "int-1" create
                address 172.16.0.2/24
                mac 00:ca:fe:ca:fe:02
                vrrp 1
                    backup 172.16.0.254
                    priority 254
                    ping-reply
                    traceroute-reply
                    mac 00:ca:fe:ca:fe:54
                exit
                vrrp 2
                    backup 172.16.0.253
                    ping-reply
                    traceroute-reply
                    mac 00:ca:fe:ca:fe:53
                exit
                vpls "evi-101"
                exit
            exit
            no shutdown
----------------------------------------------

*A:PE-3# configure service vpls 101
*A:PE-3>config>service>vpls# info
----------------------------------------------
            allow-ip-int-binding
            vxlan vni 101 create
```

```
            exit
            bgp
                route-distinguisher 192.0.2.3:101
                route-target export target:64500:101 import target:64500:101
            exit
            bgp-evpn
                vxlan
                    no shutdown
                exit
            exit
            proxy-arp
                shutdown
            exit
            stp
                shutdown
            exit
            service-name "evi-101"
            no shutdown
-----------------------------------------------
*A:PE-3# configure service vprn 10
*A:PE-3>config>service>vprn# info
-----------------------------------------------
            ecmp 2
            route-distinguisher 192.0.2.3:10
            auto-bind mpls
            vrf-target target:64500:10
            interface "int-1" create
                address 172.16.0.3/24
                mac 00:ca:fe:ca:fe:03
                vrrp 1
                    backup 172.16.0.254
                    ping-reply
                    traceroute-reply
                    mac 00:ca:fe:ca:fe:54
                exit
                vrrp 2
                    backup 172.16.0.253
                    priority 254
                    ping-reply
                    traceroute-reply
                    mac 00:ca:fe:ca:fe:53
                exit
                vpls "evi-101"
                exit
            exit
            no shutdown
-----------------------------------------------
```

For details about the EVPN and VXLAN configuration on PE-1 VPLS 101, refer to EVPN for VXLAN Tunnels (Layer 2) on page 281. The configuration of VPLS 101 on PE-2 and PE-3 has the following important aspects:

- The **allow-ip-int-binding** command is required so that the R-VPLS can be bound to VPRN 10.
- The **service-name** command is required and the configured name must match the name configured in the VPRN 10 VPLS interface.
- Even though EVPN and VXLAN are properly configured, **proxy-arp** cannot be enabled in VPLS 101. In an R-VPLS with EVPN-VXLAN, proxy-arp is not supported and the VPRN ARP table is used instead. When an EVPN MAC route that includes an IP address is received in an R-VPLS, the MAC-IP pair encoded in the route is added to the VPRN's ARP table, as opposed to the proxy-arp table.

```
*A:PE-2>config>service>vpls# proxy-arp no shutdown
MINOR: SVCMGR #8007 Cannot modify proxy arp - Not supported on routed vpls services
```

When configuring VPRN 10 on PE-2 and PE-3 the following considerations must be taken into account:

- When trying to enable existing VPRN features on interfaces linked to EVPN-VXLAN R-VPLS interfaces, the following commands are not supported:
  - arp-populate.
  - authentication-policy.
  - IPv6 and ingress>v6-routed-override-filter.
- Dynamic routing protocols such as ISIS, RIP or OSPF are not supported.
- In general, no 7x50 control plane generated packets are sent to the egress VXLAN bindings except for ARP, VRRP, ICMP and BFD.
- As depicted in Figure 47 and shown in the CLI excerpts, VRRP can be configured on the VPRN 10 VPLS interfaces to provide default gateway redundancy to the hosts connected to VPLS 101. Note that two VRRP instances are configured so that VPLS 101 upstream traffic can be load-balanced to PE-2 and PE-3. With VRRP on EVPN-VXLAN R-VPLS interfaces:
  - Ping and traceroute reply can be configured and are supported. BFD is also supported to speed up the fault detection.
  - Note that **standby-forwarding**, even if it were configured for VRRP, would not have any effect in this configuration: the standby PE will never see any flooded traffic sent to it, therefore this command is not applicable to this scenario.
- When a VPRN 10 VPLS interface is bound to VPLS 101, EVPN advertises all the IP addresses configured for that VPLS interface as MAC routes with a static MAC indication. For the remote EVPN peers, that means that those MAC addresses linked to

remote IP interfaces are protected. Note that VRRP virtual IP/MACs are also advertised by EVPN as “static” and so protected. In the example of Figure 47, the VPLS 101 FDB in PE-1 shows the IP interface MACs and VRRP MACs as **EvpnS** (Static) as shown in the following output. VPRN 10 VRRP instances and ARP entries for PE-2 are also shown:

```
*A:PE-1# show service id 101 fdb detail
===============================================================================
Forwarding Database, Service 101
===============================================================================
ServId    MAC               Source-Identifier        Type     Last Change
                                                     Age
-------------------------------------------------------------------------------
101       00:ca:fe:ca:fe:53 vxlan:                   EvpnS    07/05/14 00:02:16
                            192.0.2.3:101
101       00:ca:fe:ca:fe:54 vxlan:                   EvpnS    07/05/14 00:02:16
                            192.0.2.2:101
101       00:ca:fe:ca:fe:01 vxlan:                   Evpn     07/05/14 00:02:16
                            192.0.2.1:101
101       00:ca:fe:ca:fe:02 vxlan:                   EvpnS    07/05/14 00:02:16
                            192.0.2.2:101
101       00:ca:fe:ca:fe:03 vxlan:                   EvpnS    07/05/14 00:01:54
                            192.0.2.3:101
-------------------------------------------------------------------------------
No. of MAC Entries: 5
-------------------------------------------------------------------------------
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
===============================================================================
*A:PE-2# show router 10 vrrp instance
===============================================================================
VRRP Instances
===============================================================================
Interface Name                   VR Id Own Adm  State       Base Pri   Msg Int
                                 IP        Opr  Pol Id      InUse Pri  Inh Int
-------------------------------------------------------------------------------
int-1                            1     No  Up   Master       254       1
                                 IPv4      Up   n/a         254        No
  Backup Addr: 172.16.0.254
int-1                            2     No  Up   Backup       100       1
                                 IPv4      Up   n/a         100        No
  Backup Addr: 172.16.0.253
-------------------------------------------------------------------------------
Instances : 2
===============================================================================
*A:PE-2# show router 10 arp
===============================================================================
ARP Table (Service: 10)
===============================================================================
IP Address      MAC Address       Expiry    Type   Interface
-------------------------------------------------------------------------------
172.16.0.2      00:ca:fe:ca:fe:02 00h00m00s Oth[I] int-1
172.16.0.3      00:ca:fe:ca:fe:03 00h00m00s Evp[I] int-1
172.16.0.253    00:ca:fe:ca:fe:53 00h00m00s Oth    int-1
172.16.0.254    00:ca:fe:ca:fe:54 00h00m00s Oth[I] int-1
-------------------------------------------------------------------------------
No. of ARP Entries: 4
===============================================================================
```

EVPN-VXLAN in IRB Backhaul R-VPLS Services

Figure 49 illustrates the second inter-subnet forwarding scenario, where Layer 3 connectivity must be provided not only between the overlay networks but also within each overlay network. In the example depicted in Figure 49, a given customer (tenant) has different subnets and connectivity must be provided across all of them (CE-1, CE-2 and CE-3 must be able to communicate), bearing in mind that EVPN-VXLAN is enabled in each overlay network and IP-VPN MPLS is used to inter-connect both overlay networks. VPLS 201 is an IRB Backhaul R-VPLS service since it provides connectivity to the VPRN instances. Only the two least significant octets of the R-VPLS interface MAC addresses are shown.

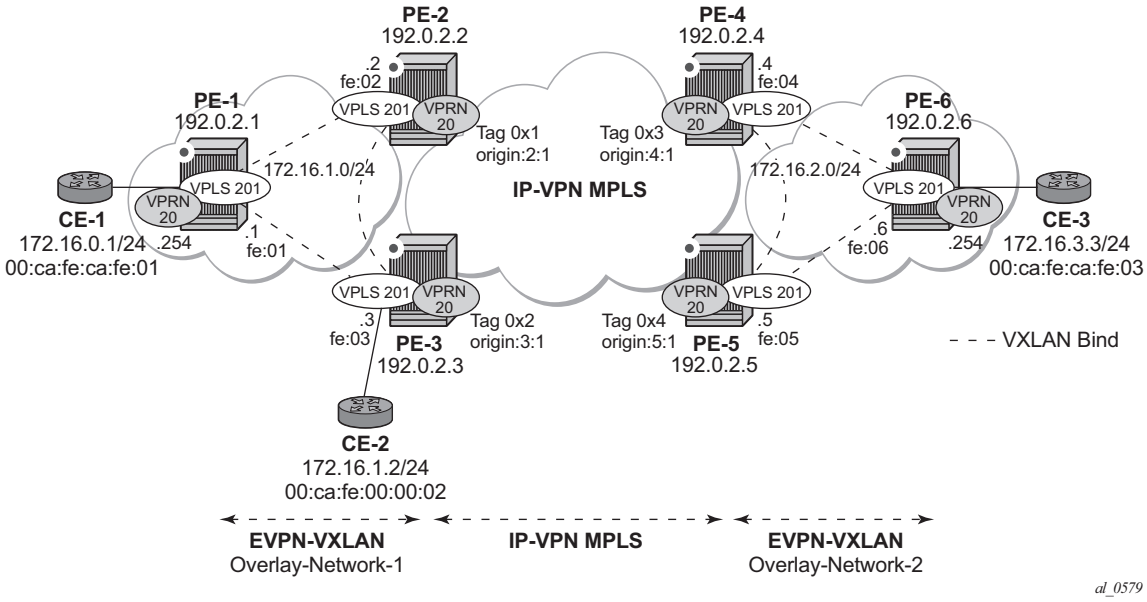

Figure 49: EVPN-VXLAN for IRB Backhaul R-VPLS services

From a BGP peering perspective, there is no change in this scenario compared to the previous one: PE-1 and PE-6 only support the EVPN address family. However in this scenario CE-1 is not connected to an R-VPLS directly linked to the VPRN instances in PE-2/PE-3. As a result of that, IP prefixes must be exchanged between PE-1 and PE-2/PE-3. EVPN is able to advertise not only MAC routes and Inclusive Multicast routes, but also IP prefix routes that contain IP prefixes that can be installed in the attached VPRN routing table.

As an example, the VPRN 20 and VPLS 201 configurations on PE-1, PE-2 and PE-3 are shown below. Similar configurations are needed in PE-3, PE-4 and PE-6.

```
*A:PE-1# configure service vprn 20
*A:PE-1>config>service>vprn# info
----------------------------------------------
            route-distinguisher 192.0.2.1:20
            vrf-target target:64500:20
            interface "int-evi-201" create
                address 172.16.1.1/24
                vpls "evi-201"
                exit
            exit
            interface "int-PE-1-CE-1" create
                address 172.16.0.254/24
                sap 1/1/1:20 create
                exit
            exit
            no shutdown
----------------------------------------------

*A:PE-1# configure service vpls 201
*A:PE-1>config>service>vpls# info
----------------------------------------------
            allow-ip-int-binding
            vxlan vni 201 create
            exit
            bgp
                route-distinguisher 192.0.2.1:201
                route-target export target:64500:201 import target:64500:201
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-201"
            no shutdown
----------------------------------------------

*A:PE-2# configure service vprn 20
*A:PE-2>config>service>vprn# info
----------------------------------------------
            route-distinguisher 192.0.2.2:20
            auto-bind mpls
            vrf-target target:64500:20
            interface "int-evi-201" create
                address 172.16.1.2/24
                vpls "evi-201"
                exit
            exit
            no shutdown
----------------------------------------------

*A:PE-2# configure service vpls 201
*A:PE-2>config>service>vpls# info
----------------------------------------------
            allow-ip-int-binding
```

```
            vxlan vni 201 create
            exit
            bgp
                route-distinguisher 192.0.2.2:201
                route-target export target:64500:201 import target:64500:201
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-201"
            no shutdown
-----------------------------------------------

*A:PE-3# configure service vprn 20
*A:PE-3>config>service>vprn# info
-----------------------------------------------
            route-distinguisher 192.0.2.3:20
            auto-bind mpls
            vrf-target target:64500:20
            interface "int-evi-201" create
                address 172.16.1.3/24
                vpls "evi-201"
                exit
            exit
            no shutdown
-----------------------------------------------

*A:PE-3# configure service vpls 201
*A:PE-3>config>service>vpls# info
-----------------------------------------------
            allow-ip-int-binding
            vxlan vni 201 create
            exit
            bgp
                route-distinguisher 192.0.2.3:201
                route-target export target:64500:201 import target:64500:201
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-201"
            sap 1/1/1:20 create
            exit
            no shutdown
-----------------------------------------------
```

As shown in the CLI excerpt, the configuration in the three nodes (PE-1/2/3) for VPLS 201 and VPRN 20 is very similar. The main difference is the **auto-bind mpls** command existing in PE-2/3's VPRN 20. This command allows the VPRN 20 on PE-2/3 to receive IP-VPN routes from the core and resolve them to MPLS tunnels. VPRN 20 on PE-1 does not require such command since all its IP prefixes are resolved to local interfaces or to EVPN peers.

The **ip-route-advertisement** command enables:

- The advertisement of IP prefixes in EVPN, in routes type 5. All the existing IP prefixes in the attached VPRN 20 routing table are advertised in EVPN within the VPLS 201 context (except for the ones associated to VPLS 201 itself).
- The installation of IP prefixes in the attached VPRN 20 routing table with a preference of 169 (bgp-vpn routes for IP-VPN have a preference of 170) and a next-hop of the GW-IP (Gateway IP) address included in the EVPN IP prefix route.

For instance, the following output shows that PE-1 advertises the IP prefix 172.16.0.0/24 as a EVPN route to PE-3 (similar route is sent to PE-2), captured by a **debug router bgp update** session. The VPRN 20 routing tables in PE-1, PE-2 and PE-3 are also shown.

```
4 2014/07/05 23:58:54.88 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 89
    Flag: 0x90 Type: 14 Len: 45 Multiprotocol Reachable NLRI:
        Address Family EVPN
        NextHop len 4 NextHop 192.0.2.1
        Type: EVPN-IP-Prefix Len: 34 RD: 192.0.2.1:201, tag: 201, ip_prefix: 17
2.16.0.0/24 gw_ip 172.16.1.1 Label: 0
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 16 Extended Community:
        target:64500:201
        bgp-tunnel-encap:VXLAN
"

*A:PE-1# show router 20 route-table
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                    Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Local   Local     23h57m35s  0
       int-PE-1-CE-1                                                0
172.16.1.0/24                                   Local   Local     23h57m48s  0
       int-evi-201                                                  0
172.16.2.0/24                                   Remote  BGP EVPN  00h00m17s  169
       172.16.1.2                                                   0
172.16.3.0/24                                   Remote  BGP EVPN  00h00m17s  169
       172.16.1.2                                                   0
-------------------------------------------------------------------------------
```

```
No. of Routes: 4
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
===============================================================================

*A:PE-2# show router 20 route-table
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                    Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Remote  BGP EVPN  00h11m04s  169
       172.16.1.1                                                   0
172.16.1.0/24                                   Local   Local     01d00h08m  0
       int-evi-201                                                  0
172.16.2.0/24                                   Remote  BGP VPN   01d00h07m  170
       192.0.2.4  (tunneled)                                        0
172.16.3.0/24                                   Remote  BGP VPN   01d00h07m  170
       192.0.2.4  (tunneled)                                        0
-------------------------------------------------------------------------------
No. of Routes: 4
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
===============================================================================

*A:PE-3# show router 20 route-table
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                    Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Remote  BGP EVPN  00h11m23s  169
       172.16.1.1                                                   0
172.16.1.0/24                                   Local   Local     01d00h09m  0
       int-evi-201                                                  0
172.16.2.0/24                                   Remote  BGP VPN   01d00h08m  170
       192.0.2.4  (tunneled)                                        0
172.16.3.0/24                                   Remote  BGP VPN   01d00h08m  170
       192.0.2.4  (tunneled)                                        0
-------------------------------------------------------------------------------
No. of Routes: 4
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
===============================================================================
```

When checking the operation of EVPN in this scenario, it is important to observe that the right next hops and prefixes are successfully installed in the VPRN 20 routing table:

- EVPN IP prefixes are sent using a GW-IP matching the primary IP interface address of the R-VPLS for which the routes are sent. For instance, as shown above, IP prefix 172.16.0.0/24 is advertised from PE-1 with GW-IP 172.16.1.1 (the IP address configured for the VPRN 20 VPLS interface in PE-1). In the PE-2/3 VPRN 20 routing tables, IP prefix 172.16.0.0/24 is installed with next hop 172.16.1.1. Traffic arriving at PE-2/3 on VPRN 20 with IP Destination Address (DA) in the 172.16.0.0 subnet matches the mentioned routing table entry. As usual, the next hop is resolved by the ARP table to a MAC and the MAC resolved by the FDB table to an egress VTEP, VNI.
- IP prefixes in the VPRN 20 routing table are advertised in IP-VPN to the remote IP-VPN MPLS peers. Received IP-VPN prefixes are installed in the VPRN 20 routing table using the remote PE system IP address as the next hop, as usual. For instance, 172.16.3.0/24 is installed in PE-2 VPRN 20's routing table with next hop (tunneled) 192.0.2.4 and preference 170.

The following considerations of how the routing table manager (RTM) handles EVPN and IP-VPN prefixes must be taken into account:

- Only VPRN interface primary addresses are advertised as GW-IP in EVPN IP prefix routes. Secondary addresses are never sent as GW-IP addresses.
- EVPN IP prefixes are advertised by default as soon as the **ip-route-advertisement** command is enabled and there are active IP prefixes in the attached VPRN routing table.
- If the same IP prefix is received on a PE via EVPN and IP-VPN at the same time for the same VPRN, by default the EVPN prefix is selected since its preference (169) is better than the IP-VPN preference (170).
- Since EVPN has a better preference compared to IP-VPN, when the VPRNs on redundant PEs are attached to the same R-VPLS service, routing loops may occur. The use case described here is an example where routing loops can occur. Check Use of Routing Policies to Avoid Routing Loops in Redundant PEs on page 340 to avoid routing loops in redundant PEs for more information.
- When the command **ip-route-advertisement** is enabled, the subnet IP prefixes are advertised in EVPN but not the "host" IP prefixes (/32 prefixes associated with the local interfaces). If the user wants to advertise the host IP prefixes as well, the **incl-host** keyword must be added to the **ip-route-advertisement** command. The following example illustrates this. The host routes can be shown with the **show router route-table all** command. When the **incl-host** keyword is added to PE-1's VPLS 201, PE-1 advertises the host routes as well and these are installed in the remote PEs' routing tables.

```
A:PE-1# show router 20 route-table
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                      Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Local   Local     01d04h17m  0
       int-PE-1-CE-1                                                0
172.16.1.0/24                                   Local   Local     01d04h18m  0
       int-evi-201                                                  0
172.16.2.0/24                                   Remote  BGP EVPN  04h20m31s  169
       172.16.1.2                                                   0
172.16.3.0/24                                   Remote  BGP EVPN  04h20m31s  169
       172.16.1.2                                                   0
-------------------------------------------------------------------------------
No. of Routes: 4
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
===============================================================================

A:PE-1# show router 20 route-table all
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                           Active    Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Local   Local     01d04h17m  0
       int-PE-1-CE-1                                      Y          0
172.16.0.254/32                                 Local   Host      01d04h17m  0
       int-PE-1-CE-1                                      Y          0
172.16.1.0/24                                   Local   Local     01d04h18m  0
       int-evi-201                                        Y          0
172.16.1.1/32                                   Local   Host      01d04h18m  0
       int-evi-201                                        Y          0
172.16.2.0/24                                   Remote  BGP EVPN  04h20m34s  169
       172.16.1.2                                         Y          0
172.16.3.0/24                                   Remote  BGP EVPN  04h20m34s  169
       172.16.1.2                                         Y          0
-------------------------------------------------------------------------------
No. of Routes: 6
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
       E = Inactive best-external BGP route
===============================================================================
A:PE-1# configure service vpls 201 bgp-evpn ip-route-advertisement incl-host

A:PE-2# show router 20 route-table
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                      Metric
-------------------------------------------------------------------------------
```

```
172.16.0.0/24                                   Remote  BGP EVPN  04h25m22s  169
       172.16.1.1                                                      0
172.16.0.254/32                                 Remote  BGP EVPN  00h03m52s  169
       172.16.1.1                                                      0
172.16.1.0/24                                   Local   Local     01d04h22m  0
       int-evi-201                                                     0
172.16.2.0/24                                   Remote  BGP VPN   01d04h22m  170
       192.0.2.4  (tunneled)                                          0
172.16.3.0/24                                   Remote  BGP VPN   01d04h22m  170
       192.0.2.4  (tunneled)                                          0
-------------------------------------------------------------------------------
No. of Routes: 5
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
```

- ECMP is fully supported for the VPRN for EVPN IP prefix routes coming from different GW-IP next-hops. However ECMP is not supported for IP prefixes routes belonging to different owners (EVPN and IP-VPN). ECMP behavior for EVPN is illustrated in the following output. When **ecmp** is enabled in PE-1's VPRN 20, an additional route with a different GW-IP as next-hop is installed in the routing table for the IP-prefixes 172.16.2.0/24 and 172.16.3.0/24.

```
*A:PE-1# configure service vprn 20 ecmp 2
*A:PE-1# show router 20 route-table
===============================================================================
Route Table (Service: 20)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                    Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Local   Local     01d04h50m  0
       int-PE-1-CE-1                                                   0
172.16.1.0/24                                   Local   Local     01d04h50m  0
       int-evi-201                                                     0
172.16.2.0/24                                   Remote  BGP EVPN  00h00m01s  169
       172.16.1.2                                                      0
172.16.2.0/24                                   Remote  BGP EVPN  00h00m01s  169
       172.16.1.3                                                      0
172.16.3.0/24                                   Remote  BGP EVPN  00h00m01s  169
       172.16.1.2                                                      0
172.16.3.0/24                                   Remote  BGP EVPN  00h00m01s  169
       172.16.1.3                                                      0
-------------------------------------------------------------------------------
No. of Routes: 6
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
===============================================================================
```

EVPN-VXLAN in EVPN Tunnel R-VPLS Services

The previous scenario shows how to use EVPN-VXLAN to provide inter-subnet forwarding for a given tenant, where R-VPLS services can contain hosts and also offer transit services between VPRN instances. For example, in the use case depicted in Figure 49, VPLS 201 in Overlay-Network-1 is an R-VPLS that can provide intra-subnet connectivity to all the hosts in subnet 172.16.1.0/24 (for example, CE-2 belongs to this subnet) but it can also provide "transit" or "backhaul" connectivity to hosts in subnet 172.16.0.0/24 (for example, CE-1) sending packets to subnets 172.16.2.0/24 or 172.16.3.0/24. In some cases, the R-VPLS where EVPN-VXLAN is enabled does not need to provide intra-subnet connectivity and it is purely a transit or backhaul service where VPRN IRB interfaces are connected. Figure 50 illustrates this use case.

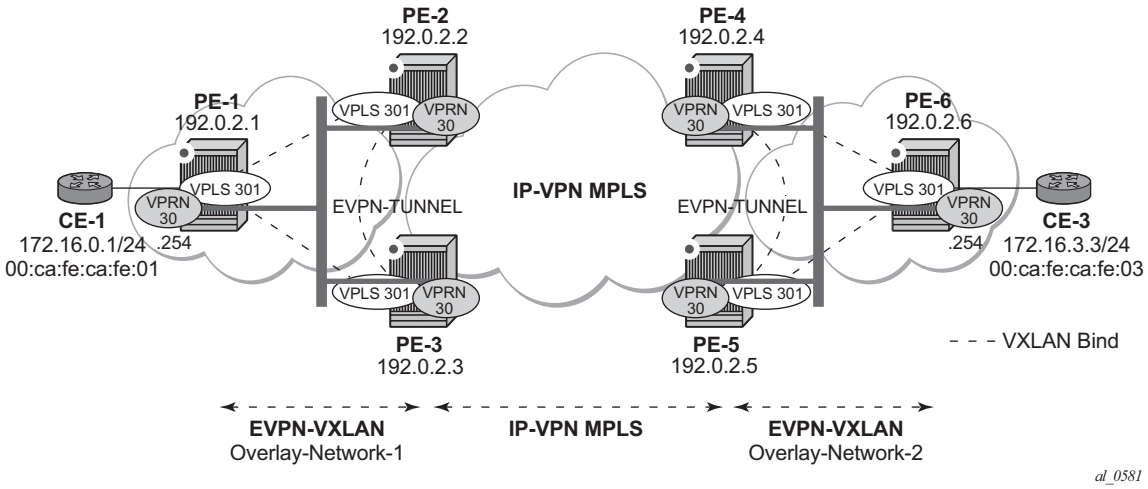

Figure 50: EVPN-VXLAN in EVPN-Tunnel R-VPLS Services

Compared to the use case in Figure 49, in this case the R-VPLS connecting the IRB interfaces in Overlay-Network-1 (VPLS 301) does not have any connected host. If that is the case, VPLS 301 can be configured as an EVPN tunnel.

EVPN tunnels are enabled using the evpn-tunnel command under the R-VPLS interface configured on the VPRN. EVPN tunnels bring the following benefits to EVPN-VXLAN IRB backhaul R-VPLS services:

- Easier and simpler provisioning of the tenant service: if an EVPN tunnel is configured in an IRB backhaul R-VPLS there is no need to provision the IRB IP addresses in the VPRN. This makes the provisioning easier to automate and saves IP addresses from the tenant IP space.
- Higher scalability of the IRB backhaul R-VPLS: if EVPN tunnels are enabled, BUM traffic is suppressed in the EVPN-VXLAN IRB backhaul R-VPLS service (it is not

required). As a result, the number of VXLAN bindings in IRB backhaul R-VPLS services with EVPN tunnels can be much higher.

As an example, the VPRN 30 and VPLS 301 configurations on PE-1, PE-2 and PE-3 are shown below. Note that similar configurations are needed in PE-4, PE-5 and PE-6.

```
A:PE-1# configure service vprn 30
A:PE-1>config>service>vprn# info
----------------------------------------------
            route-distinguisher 192.0.2.1:30
            vrf-target target:64500:30
            interface "int-PE-1-CE-1" create
                address 172.16.0.254/24
                sap 1/1/1:30 create
                exit
            exit
            interface "int-evi-301" create
                vpls "evi-301"
                    evpn-tunnel
                exit
            exit
            no shutdown
----------------------------------------------

A:PE-1# configure service vpls 301
A:PE-1>config>service>vpls# info
----------------------------------------------
            allow-ip-int-binding
            vxlan vni 301 create
            exit
            bgp
                route-distinguisher 192.0.2.1:301
                route-target export target:64500:301 import target:64500:301
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-301"
            no shutdown
----------------------------------------------
A:PE-2# configure service vprn 30
A:PE-2>config>service>vprn# info
----------------------------------------------
            route-distinguisher 192.0.2.2:30
            auto-bind mpls
            vrf-target target:64500:30
            interface "int-evi-301" create
                vpls "evi-301"
                    evpn-tunnel
                exit
            exit
            no shutdown
```

```
-----------------------------------------------
A:PE-2# configure service vpls 301
A:PE-2>config>service>vpls# info
-----------------------------------------------
            allow-ip-int-binding
            vxlan vni 301 create
            exit
            bgp
                route-distinguisher 192.0.2.2:301
                route-target export target:64500:301 import target:64500:301
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-301"
            no shutdown
-----------------------------------------------
A:PE-3# configure service vprn 30
A:PE-3>config>service>vprn# info
-----------------------------------------------
            route-distinguisher 192.0.2.3:30
            auto-bind mpls
            vrf-target target:64500:30
            interface "int-evi-301" create
                vpls "evi-301"
                    evpn-tunnel
                exit
            exit
            no shutdown
-----------------------------------------------
A:PE-3# configure service vpls 301
A:PE-3>config>service>vpls# info
-----------------------------------------------
            allow-ip-int-binding
            vxlan vni 301 create
            exit
            bgp
                route-distinguisher 192.0.2.3:301
                route-target export target:64500:301 import target:64500:301
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-301"
            no shutdown
-----------------------------------------------
```

As shown in the output above, the configuration in the three nodes (PE-1/2/3) for VPLS 301 and VPRN 30 is similar to the configuration of VPLS 201 and VPRN 20 in the previous scenario, however, when the **evpn-tunnel** command is added to the VPRN interface, there is no need to configure an IP interface address. Note that **evpn-tunnel** can be enabled independently of **ip-route-advertisement** (although no route-type 5 advertisements are sent in that case).

A given VPRN supports regular IRB backhaul R-VPLS services as well as EVPN tunnel R-VPLS services. A maximum of eight R-VPLS services with **ip-route-advertisement** enabled per VPRN is supported (in any combination of regular IRB R-VPLS or EVPN tunnel R-VPLS services). Note that EVPN tunnel R-VPLS services do not support SAPs or SDP-binds. No frames are flooded in an EVPN tunnel R-VPLS service, and, in fact no inclusive multicast routes are exchanged in R-VPLS services that are configured as EVPN tunnels. The show service id vxlan command for an R-VPLS service configured as an EVPN tunnel shows <egress VTEP, VNI> bindings excluded from the “multicast list”, in other words, the VXLAN bindings are not used to flood BUM traffic:

```
*A:PE-2# show service id 301 vxlan
===============================================================================
VPLS VXLAN, Ingress VXLAN Network Id: 301
===============================================================================
Egress VTEP, VNI
===============================================================================
VTEP Address           Egress VNI     Num. MACs    In Mcast List?  Oper State
-------------------------------------------------------------------------------
192.0.2.1              301            1            No              Up
192.0.2.3              301            1            No              Up
-------------------------------------------------------------------------------
Number of Egress VTEP, VNI : 2
-------------------------------------------------------------------------------
===============================================================================
```

The process followed upon receiving a route-type 5 on a regular IRB R-VPLS interface (previous scenario) differs from the one for an EVPN tunnel type (this scenario):

- IRB backhaul R-VPLS VPRN interface:
  - When a route-type 2 that includes an IP address is received and it becomes active, the MAC/IP information is added to the FDB and ARP tables. This can be checked with the **show>router>arp** command and the **show>service>id>fdb detail** command.
  - When a route-type 5 is received on (for instance) PE-2, and becomes active for the R-VPLS service, the IP prefix is added to the VPRN routing table regardless of the existence of a route-type 2 that can resolve the GW IP address. If a packet is received from the WAN side and the IP lookup hits an entry for which the GW IP (IP next-hop) does not have an active ARP entry, the system will ARP to get the MAC. If the ARP is resolved but the MAC is unknown in the FDB table, the system will flood the ARP message into the R-VPLS multicast list. Routes type 5 can be checked in the routing table with the **show>router>route-table** command and the **show>router>fib** command.

- EVPN tunnel R-VPLS VPRN interface:
  - When a route-type 2 is received and becomes active, the MAC address is added to the FDB (only). This MAC address is normally a GW-MAC.
  - When a route-type 5 is received on (for instance) PE-1, the system looks for the GW-MAC. The IP prefix is added to the VPRN routing table with next hop equal to EVPN-tunnel-GW-MAC; for example (see below), ET-d8:45:ff:00:00:6a is an EVPN tunnel with GW-MAC d8:45:ff:00:00:6a. The GW-MAC is added from the GW-MAC extended community sent along with the route-type 5 for prefix 172.16.3.0/24. If a packet is received from the CE-1 and the IP lookup hits an entry for which the next hop is a EVPN tunnel:GW-MAC, the system looks up the GW-MAC in the FDB. Normally a route-type 2 with the GW-MAC has already been received so that the GW-MAC has been added to the FDB. If the GW-MAC is not present in the FDB, the packet will be dropped.
  - Note that the IP prefixes with GW-MACs as next hops are displayed in the show router route-table command, as shown below for the setup in Figure 4. The show service id fdb detail command can be used to look for the forwarding information for a given GW-MAC:

```
A:PE-1# show router 30 route-table
===============================================================================
Route Table (Service: 30)
===============================================================================
Dest Prefix[Flags]                              Type    Proto     Age        Pref
      Next Hop[Interface Name]                                      Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                   Local   Local     00h06m15s  0
       int-PE-1-CE-1                                                0
172.16.3.0/24                                   Remote  BGP EVPN  00h05m31s  169
       int-evi-301 (ET-d8:45:ff:00:00:6a)                           0
-------------------------------------------------------------------------------
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
===============================================================================
A:PE-1# show service id 301 fdb detail
===============================================================================
Forwarding Database, Service 301
===============================================================================
ServId    MAC               Source-Identifier        Type     Last Change
                                                     Age
-------------------------------------------------------------------------------
301       d8:45:ff:00:00:6a vxlan:                   EvpnS    07/05/14 00:02:46
                            192.0.2.2:301
301       d8:47:ff:00:00:6a cpm                      Intf     07/05/14 00:01:48
301       d8:48:ff:00:00:6a vxlan:                   EvpnS    07/05/14 00:02:18
                            192.0.2.3:301
-------------------------------------------------------------------------------
No. of MAC Entries: 3
-------------------------------------------------------------------------------
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
===============================================================================
```

Note that IP prefix routes sent for EVPN tunnel R-VPLS services do not contain a GW-IP (the GW-IP will be zero) but convey a GW-MAC address that is used in the peer VPRN routing table. The following output shows PE-2's VPRN 30 interface MAC address and the route-type 5 sent to PE-1 using the MAC as GW-MAC:

```
*A:PE-2# show router 30 interface detail | match MAC
MAC Address      : d8:45:ff:00:00:6a    Mac Accounting    : Disabled

*A:PE-2# configure service vpls 301 bgp-evpn ip-route-advertisement
*A:PE-2#
6 2014/07/05 00:29:41.79 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.1
"Peer 1: 192.0.2.1: UPDATE
Peer 1: 192.0.2.1 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 105
    Flag: 0x90 Type: 14 Len: 45 Multiprotocol Reachable NLRI:
        Address Family EVPN
        NextHop len 4 NextHop 192.0.2.2
        Type: EVPN-IP-Prefix Len: 34 RD: 192.0.2.2:301, tag: 301, ip_prefix: 17
2.16.3.0/24 gw_ip 0.0.0.0 Label: 0
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 32 Extended Community:
        origin:69:1
        target:64500:301
        mac-nh:d8:45:ff:00:00:6a
        bgp-tunnel-encap:VXLAN
"
```

Looking at the VPRN 30 routing table, since IP prefixes are shown with an EVPN tunnel next-hop (GW-MAC) as opposed to an IP next-hop, the user may think that no ARP entries are consumed by VPRN 30. However internal ARP entries are still consumed in VPRN 30. Although not shown in the show router 30 arp command, the **summary** option shows the consumption of internal ARP entries for EVPN.

```
*A:PE-2# show router 30 route-table
===============================================================================
Route Table (Service: 30)
===============================================================================
Dest Prefix[Flags]                            Type    Proto     Age        Pref
      Next Hop[Interface Name]                                    Metric
-------------------------------------------------------------------------------
172.16.0.0/24                                 Remote  BGP EVPN  00h31m34s  169
       int-evi-301 (ET-d8:47:ff:00:00:6a)                          0
172.16.3.0/24                                 Remote  BGP VPN   00h59m09s  170
       192.0.2.4 (tunneled)                                        0
-------------------------------------------------------------------------------
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
```

```
       S = Sticky ECMP requested
===============================================================================
*A:PE-2# show router 30 arp

===============================================================================
ARP Table (Service: 30)
===============================================================================
IP Address      MAC Address       Expiry    Type   Interface
-------------------------------------------------------------------------------
No Matching Entries Found
===============================================================================
*A:PE-2# show router 30 arp summary

============================================================
ARP Table Summary (Service: 30)
============================================================
Local ARP Entries    : 1
Static ARP Entries   : 0
Dynamic ARP Entries  : 0
Managed ARP Entries  : 0
Internal ARP Entries : 0
BGP-EVPN ARP Entries : 1
------------------------------------------------------------
No. of ARP Entries   : 2
============================================================
```

The number of BGP-EVPN ARP Entries in the **show router 30 arp summary** command matches the number of remote valid GW-MACs for VPRN 30.

Routing Policies for IP Prefixes in EVPN

Routing policies are supported for IP prefixes imported/exported through BGP EVPN. The default import/export behavior for IP prefixes in EVPN can be modified by the use of routing policies applied either at peer level (**config>router>bgp>group/group>neighbor>import/export**) or VPLS level (**config>service>vpls>bgp>vsi-import/vsi-export**).

When applying routing policies to control the distribution of prefixes between EVPN and IP-VPN, the user must take into account that both families are completely separated as far as BGP is concerned and that when prefixes from a family are imported in the RTM, the BGP attributes are lost to the other family. The use of tags allows the controlled distribution of prefixes across the two families.

Figure 51 illustrates how vpn-ipv4 routes are imported into the RTM and then passed onto EVPN for its own processing. Note that vpn-ipv4 routes can be tagged at ingress and this tag is preserved throughout the RTM and EVPN processing so that the tag can be "matched" by the egress BGP routing policy. In this particular example, egress EVPN routes matching tag 10, are modified to add a site-of-origin community origin:64500:1.

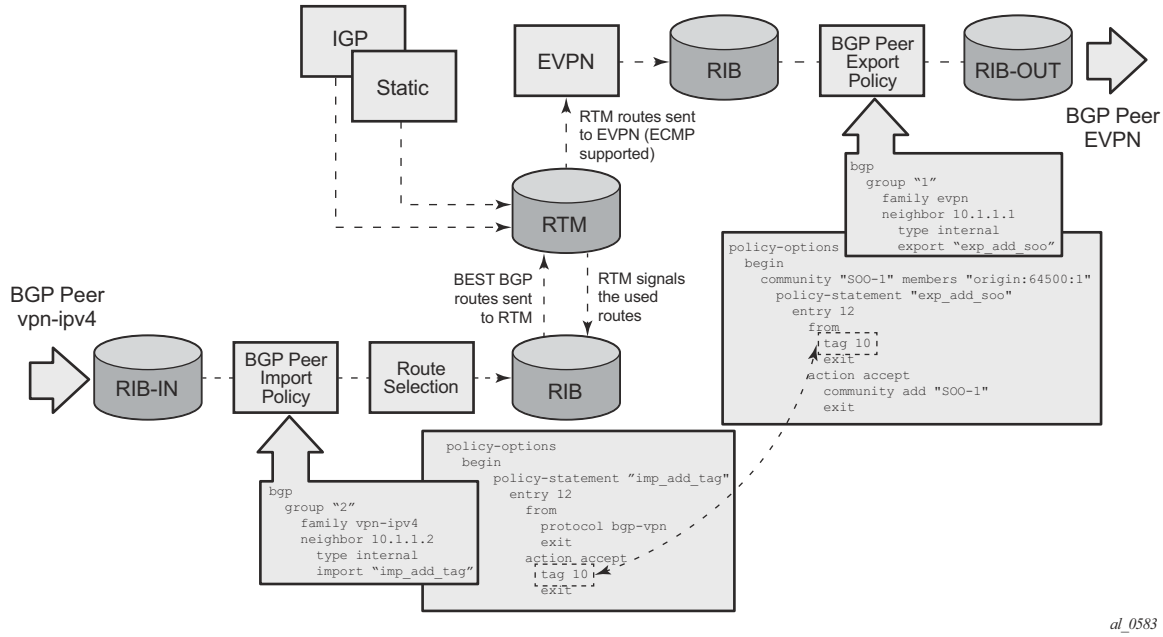

Figure 51: Routing Policies for Egress EVPN Routes

Policy TAGS can be used to match EVPN IP-prefixes that were learned not only from BGP vpn-ipv4 but also from other routing protocols. Note that the tag range supported for each protocol is different:

```
<tag>  : accepts in decimal or hex
         [0x1..0xFFFFFFFF]H (for OSPF and ISIS)
         [0x1..0xFFFF]H (for RIP)
         [0x1..0xFF]H (for BGP)
```

Figure 52 illustrates the reverse workflow: routes imported from EVPN and exported from RTM to BGP vpn-ipv4. In this example, EVPN routes received with community VM-mob are tagged with TAG 200. At the egress vpn-ipv4 peers, only the routes with TAG 200 are advertised.

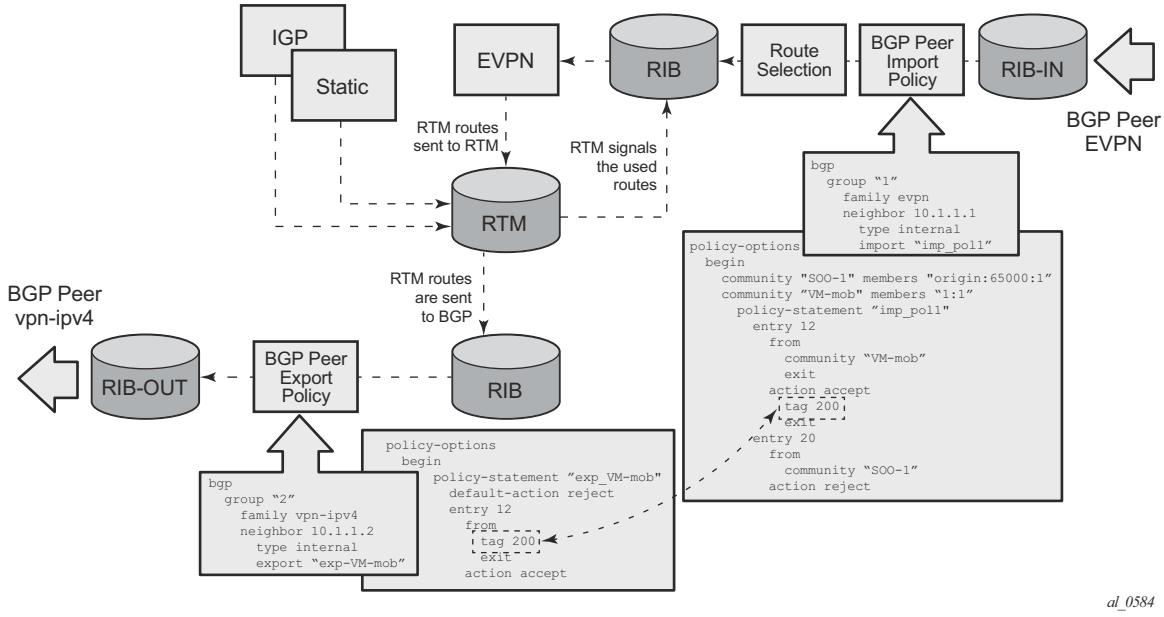

Figure 52: Routing Policies for Ingress EVPN Routes

The above behavior and the use of tags is also valid for **vsi-import** and **vsi-export** policies. The behavior can be summarized in the following statements:

- For EVPN prefix routes received and imported in RTM:
  - Routes can be matched on communities and tags can be added to them. This works at peer level or vsi-import level.
  - Well-known communities (no-export|no-export-subconfed|no-advertise) also require that the routing policies add a tag if the user wants to modify the behavior when exporting to BGP.
  - Routes can be matched based on family evpn.
  - Routes cannot be matched on prefix-list.
- For exporting RTM to EVPN prefix routes:
  - Routes can be matched on tags and based on that, communities added, or routes accepted or rejected, etc. This works at peer level or vsi-export level.
  - Tags can be added for static-routes, rip, ospf, isis and bgp and then be matched in the vsi-export policy for EVPN IP-prefix route advertisement.
  - Tags cannot be added for direct routes.

Use of Routing Policies to Avoid Routing Loops in Redundant PEs

When redundant PE VPRN instances are connected to the same R-VPLS service (IRB backhaul or EVPN tunnel R-VPLS) with the ip-route-advertisement command enabled, routing loops can occur in two different use-cases:

1. Routing loop caused by EVPN and IP-VPN interaction in the RTM.
2. Routing loop caused by EVPN in “parallel” R-VPLS services.

Policy configuration examples for both cases are provided below.

Routing loop use-case 1: EVPN and IP-VPN interaction

This use case refers to scenarios with redundant PEs and VPRNs connected to the same R-VPLS with **ip-route-advertisement**. The scenarios in Figure 49 (EVPN-VXLAN for IRB Backhaul R-VPLS services) and Figure 50 (EVPN-VXLAN in EVPN tunnel R-VPLS services) are examples of this use case. In both scenarios the following process causes a routing loop:

1. IP prefix 172.16.3.0/24 is advertised by PE-4 to PE-2 and PE-3.
2. PE-2 imports that prefix in the VPRN routing table and re-advertises the IP prefix in EVPN to PE-1 and PE-3 (the same thing happens in PE-3).
3. PE-3 already has the 172.16.3.0/24 prefix in the VPRN routing table with preference 170 (IP-VPN) but since it receives the IP prefix from EVPN with lower preference (169), the RTM will install the EVPN prefix in the VPRN routing table (the same thing happens in PE-2).
4. PE-3 advertises the EVPN learned IP prefix to all MP-BGP vpn-ipv4 peers (also PE-2).
5. PE-2 receives the IP prefix again from PE-3 and will advertise it in EVPN again, creating a routing loop (PE-3 will do the same thing as well).

This routing loop also happens in traditional multi-homed IP-VPN scenarios where the PE-CE eBGP and MP-BGP vpn-ipv4/v6 protocols interact in the same VPRN RTM, with different router preferences. In either case (EVPN or eBGP interaction with MP-BGP) the issue can be solved by the use of routing policies and site-of-origin communities.

Routing policies are applied to PE-2 and PE-3 (also to PE-4 and PE-5) and allow the redundant PEs to reject their own generated routes in order to avoid the loops. These routing policies can be applied at vsi-import/export level or BGP group/neighbor level. The following output shows an example of routing policies applied at BGP neighbor level for PE-2 (similar policies are applied on PE-3/4/5). Note that neighbor or group level policies are the preferred way in this kind of use case: a single set of policies is sufficient, as opposed to a set of policies per service (if the policies are applied at vsi-import/export level).

```
*A:PE-2>config>router>bgp# info
----------------------------------------------
            vpn-apply-import
            vpn-apply-export
            min-route-advertisement 1
            enable-peer-tracking
            rapid-withdrawal
            rapid-update evpn
            group "DC"
                family vpn-ipv4 evpn
                type internal
                neighbor 192.0.2.1
                    import "add-tag_to_bgp-evpn_routes"
                exit
                neighbor 192.0.2.3
                    import "reject_based_on_SOO"
                    export "add-SOO_on_export"
                exit
            exit
            group "WAN"
                family vpn-ipv4
                type internal
                neighbor 192.0.2.4
                    import "add-tag_to_bgp-vpn_routes"
                exit
                neighbor 192.0.2.5
                    import "add-tag_to_bgp-vpn_routes"
                exit
            exit
            no shutdown
----------------------------------------------
*A:PE-2>config>router>policy-options# info
----------------------------------------------
            community "SOO-PE-2" members "origin:2:1"
            community "SOO-PE-3" members "origin:3:1"
            policy-statement "add-SOO_on_export"
                entry 10
                    from
                        tag 0x1
                    exit
                    action accept
                        community add "SOO-PE-2"
                    exit
                exit
                entry 20
                    from
                        tag 0x2
                    exit
                    action accept
                        community add "SOO-PE-3"
                    exit
                exit
            exit
            policy-statement "reject_based_on_SOO"
                entry 10
                    from
                        community "SOO-PE-2"
                    exit
                    action reject
```

```
                exit
                entry 20
                    from
                        community "SOO-PE-3"
                    exit
                    action reject
                exit
            exit
            policy-statement "add-tag_to_bgp-vpn_routes"
                entry 10
                    from
                        protocol bgp-vpn
                    exit
                    action accept
                        tag 0x1
                    exit
                exit
            exit
            policy-statement "add-tag_to_bgp-evpn_routes"
                entry 10
                    from
                        family evpn
                    exit
                    action accept
                        tag 0x1
                    exit
                exit
            exit
-------------------------------------------
```

EVPN and MP-BGP routes are tagged at import and add a site-of-origin community. Routes exchanged between the two redundant PEs are rejected if they are received by a PE with its own site-of-origin.

Routing loop use-case 2: EVPN in parallel R-VPLS services

If a given VPRN is connected to more than one R-VPLS with **ip-route-advertisement** enabled, IP prefixes that belong to one R-VPLS are advertised into the other R-VPLS and vice versa. When redundant PEs are used, a routing loop will occur. Figure 53 illustrates this use case. Note that the example shows R-VPLS with an EVPN tunnel configuration but the same routing loop occurs for regular IRB backhaul R-VPLS services.

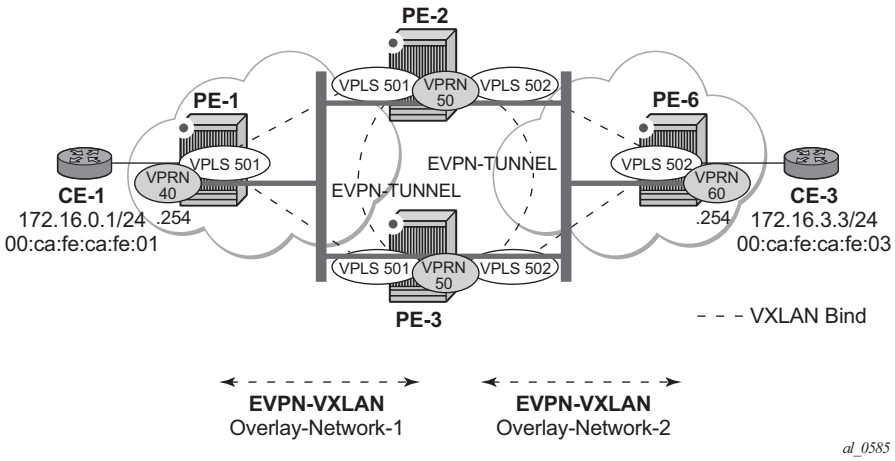

Figure 53: EVPN in Parallel R-VPLS Services

The configuration of VPRN 50 as well as VPLS 501/502 and the required policies are shown below. For this use case, policies must be applied at vsi-import/export level since more granularity is required when modifying the imported/exported routes.

```
*A:PE-2# configure service vprn 50
*A:PE-2>config>service>vprn# info
----------------------------------------------
            route-distinguisher 192.0.2.2:50
            interface "int-evi-501" create
                vpls "evi-501"
                    evpn-tunnel
                exit
            exit
            interface "int-evi-502" create
                vpls "evi-502"
                    evpn-tunnel
                exit
            exit
            no shutdown
----------------------------------------------
*A:PE-2# configure service vpls 501
*A:PE-2>config>service>vpls# info
----------------------------------------------
```

```
            allow-ip-int-binding
            vxlan vni 501 create
            exit
            bgp
                route-distinguisher 192.0.2.2:501
                vsi-export "vsi-export-policy-501"
                vsi-import "vsi-import-policy-501"
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-501"
            no shutdown
-----------------------------------------------
*A:PE-2>config>service>vpls# info
-----------------------------------------------
            allow-ip-int-binding
            vxlan vni 502 create
            exit
            bgp
                route-distinguisher 192.0.2.2:502
                vsi-export "vsi-export-policy-502"
                vsi-import "vsi-import-policy-502"
            exit
            bgp-evpn
                ip-route-advertisement
                vxlan
                    no shutdown
                exit
            exit
            stp
                shutdown
            exit
            service-name "evi-502"
            no shutdown
-----------------------------------------------
*A:PE-2>config>router>policy-options# info
-----------------------------------------------
            community "exp_RVPLS501" members "origin:2:11" "target:64500:501"
            community "exp_RVPLS502" members "origin:2:11" "target:64500:502"
            community "SOO-PE-2-RVPLS" members "origin:2:11"
            community "SOO-PE-3-RVPLS" members "origin:3:11"
            community "SOO_PE-3_RVPLS501" members "origin:3:11" "target:64500:501"
            community "SOO_PE-3_RVPLS502" members "origin:3:11" "target:64500:502"
            policy-statement "vsi-export-policy-501"
                entry 10
                    from
                        tag 0x5
                    exit
                    action accept
                        community add "SOO_PE-3_RVPLS501"
                    exit
                exit
```

```
        entry 20
            action accept
                community add "exp_RVPLS501"
            exit
        exit
    exit
    policy-statement "vsi-export-policy-502"
        entry 10
            from
                tag 0x5
            exit
            action accept
                community add "SOO_PE-3_RVPLS502"
            exit
        exit
        entry 20
            action accept
                community add "exp_RVPLS502"
            exit
        exit
    exit
    policy-statement "vsi-import-policy-501"
        entry 10
            from
                community "SOO-PE-2-RVPLS"
            exit
            action reject
        exit
        entry 20
            from
                community "SOO_PE-3_RVPLS501"
            exit
            action accept
                tag 0x5
            exit
        exit
        default-action accept
        exit
    exit
    policy-statement "vsi-import-policy-502"
        entry 10
            from
                community "SOO-PE-2-RVPLS"
            exit
            action reject
        exit
        entry 20
            from
                community "SOO_PE-3_RVPLS502"
            exit
            action accept
                tag 0x5
            exit
        exit
        default-action accept
        exit
    exit
```

Troubleshooting and Debug Commands

For general information on EVPN and VXLAN troubleshooting and debug commands, please refer to chapter EVPN for VXLAN Tunnels (Layer 2) on page 281. This information below focuses on specific commands for Layer-3 applications.

When troubleshooting and operating a EVPN-VXLAN scenario with inter-subnet forwarding, it is important to check the IP prefixes and next-hops, as well as ARP tables and FDB tables (**show router x route-table**, **show router x arp**, **show service id y fdb detail**).

ICMP commands can also help checking the connectivity. When traceroute is used on EVPN-VXLAN in EVPN tunnel interfaces, EVPN tunnel interface hops in the traceroute commands are showing the VPRN loopback address or the other non evpn-tunnel interface address. In VPRN services where all of the interfaces are of type EVPN tunnel, ICMP packets fail until an IP address is configured. The following output shows a traceroute from VPRN 30 in PE-1 to CE-3 and from PE-2 to CE-1 (see Figure 50):

```
A:PE-1# traceroute router 30 172.16.3.3
traceroute to 172.16.3.3, 30 hops max, 40 byte packets
  1  192.0.2.2 (192.0.2.2)    1.79 ms  1.60 ms  1.51 ms
  2  0.0.0.0  * * *
  3  192.0.2.6 (192.0.2.6)    3.15 ms  3.20 ms  2.93 ms
  4  172.16.3.3 (172.16.3.3)    4.24 ms  3.28 ms  3.31 ms

*A:PE-2# traceroute router 30 172.16.0.1
traceroute to 172.16.0.1, 30 hops max, 0 byte packets

Send failed. Unable to find local ip address
```

When troubleshooting R-VPLS services, specifically R-VPLS services configured as EVPN tunnels, the limit of peer PEs per EVPN tunnel service is much higher than for a regular R-VPLS service since the egress <VTEP, VNI> bindings do not have to be added to the multicast flooding list. For this reason, the following **tools dump** command has been added to check the consumed/total EVPN tunnel next hops. Note that the number of EVPN tunnel next hops matches the number of remote GW-MAC addresses per EVPN tunnel R-VPLS service.

```
A:PE-1# tools dump service id 501 evpn usage

Evpn Tunnel Interface IP Next Hop: 2/8189
```

Finally, when troubleshooting EVPN routes and routing policies, the **show router bgp routes evpn** command and its filters can help:

- Check that the expected routes are received, properly imported and communities/tags added/replaced/removed.
- Check that the expected routes are sent, properly exported and communities added/ replaced/removed.

Examples of EVPN IP prefix routes including communities and tags are shown below.

```
*A:PE-2# show router bgp routes evpn ?
  - evpn <evpn-type>

      inclusive-mcast - Display BGP EVPN Inclusive-Mcast Routes
      ip-prefix       - Display BGP EVPN IP-Prefix Routes
      mac             - Display BGP EVPN Mac Routes

*A:PE-2# show router bgp routes evpn ip-prefix ?
  - ip-prefix [hunt|detail] [rd <rd>] [prefix <ip-prefix/mask>] [community
    <comm-id>] [tag <vni-id>] [next-hop <ip-address>]

...

*A:PE-2# show router bgp routes evpn ip-prefix prefix 172.16.0.0/24 hunt community ori-
gin:69:11
===============================================================================
 BGP Router ID:192.0.2.2       AS:64500       Local AS:64500
===============================================================================
 Legend -
 Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
 Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
===============================================================================
BGP EVPN IP-Prefix Routes
===============================================================================
-------------------------------------------------------------------------------
RIB In Entries
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
RIB Out Entries
-------------------------------------------------------------------------------
...
Network        : N/A
Nexthop        : 192.0.2.2
To             : 192.0.2.1
Res. Nexthop   : n/a
Local Pref.    : 100                    Interface Name : NotAvailable
Aggregator AS  : None                   Aggregator     : None
Atomic Aggr.   : Not Atomic             MED            : 0
AIGP Metric    : None
Connector      : None
Community      : origin:2:11 target:64500:502
```

```
                mac-nh:d8:45:ff:00:01:33 bgp-tunnel-encap:VXLAN
Cluster        : No Cluster Members
Originator Id  : None                   Peer Router Id : 192.0.2.1
Origin         : IGP
AS-Path        : No As-Path
EVPN type      : IP-PREFIX
ESI            : N/A                    Tag            : 502
Gateway Address: d8:45:ff:00:01:33
Prefix         : 172.16.0.0/24          Route Dist.    : 192.0.2.2:502
MPLS Label     : 0
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                      Dest Class     : 0
-------------------------------------------------------------------------------
Routes : 2
===============================================================================

*A:PE-2# show router bgp routes evpn ip-prefix prefix 172.16.0.0/24 detail
===============================================================================
 BGP Router ID:192.0.2.2       AS:64500       Local AS:64500
===============================================================================
 Legend -
 Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
 Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
===============================================================================
BGP EVPN IP-Prefix Routes
===============================================================================
-------------------------------------------------------------------------------
Original Attributes

Network        : N/A
Nexthop        : 192.0.2.1
From           : 192.0.2.1
Res. Nexthop   : N/A
Local Pref.    : 100                    Interface Name : NotAvailable
Aggregator AS  : None                   Aggregator     : None
Atomic Aggr.   : Not Atomic             MED            : 0
AIGP Metric    : None
Connector      : None
Community      : target:64500:201 bgp-tunnel-encap:VXLAN
Cluster        : No Cluster Members
Originator Id  : None                   Peer Router Id : 192.0.2.1
Flags          : Used  Valid  Best  IGP
Route Source   : Internal
AS-Path        : No As-Path
EVPN type      : IP-PREFIX
ESI            : N/A                    Tag            : 201
Gateway Address: 172.16.1.1
Prefix         : 172.16.0.0/24          Route Dist.    : 192.0.2.1:201
MPLS Label     : 0
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                      Dest Class     : 0

Modified Attributes

Network        : N/A
```

```
Nexthop        : 192.0.2.1
From           : 192.0.2.1
Res. Nexthop   : N/A
Local Pref.    : 100                    Interface Name : NotAvailable
Aggregator AS  : None                   Aggregator     : None
Atomic Aggr.   : Not Atomic             MED            : 0
AIGP Metric    : None
Connector      : None
Community      : target:64500:201 bgp-tunnel-encap:VXLAN
Cluster        : No Cluster Members
Originator Id  : None                   Peer Router Id : 192.0.2.1
Flags          : Used  Valid  Best  IGP
Route Source   : Internal
AS-Path        : No As-Path
EVPN type      : IP-PREFIX
ESI            : N/A                    Tag            : 201
Gateway Address: 172.16.1.1
Prefix         : 172.16.0.0/24          Route Dist.    : 192.0.2.1:201
MPLS Label     : 0
Route Tag      : 1
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                      Dest Class     : 0
-------------------------------------------------------------------------------
...
```

Conclusion

SR OS supports not only the EVPN control plane for VXLAN tunnels in Layer 2 applications but also the simultaneous use of EVPN and VXLAN for VPN customers (tenants) with intra and inter-subnet connectivity requirements. R-VPLS services can be configured to provide default gateway connectivity to hosts, IRB backhaul connectivity to VPRN services and EVPN tunnel connectivity to VPRN services. When configured to do so, EVPN can advertise IP prefixes and interact with the VPRN RTM to propagate IP prefix connectivity between EVPN and other routing protocols in the VPRN, including IP-VPN. This example has shown how to configure R-VPLS services for all these functions, as well as how to configure routing policies for EVPN-based IP prefixes.