

# VXLAN and BGP EVPN Configuration Guide for Dell SmartFabric OS10

Release 10.5.3.2

## Notes, cautions, and warnings

 **NOTE:** A NOTE indicates important information that helps you make better use of your product.

 **CAUTION:** A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

 **WARNING:** A WARNING indicates a potential for property damage, personal injury, or death.

<b>Chapter 1: VXLAN .....</b>	<b>6</b>
VXLAN concepts.....	7
VXLAN as NVO solution.....	8
Configure VXLAN.....	8
Configure source IP address on VTEP.....	8
Configure a VXLAN virtual network.....	9
Configure VLAN-tagged access ports.....	9
Configure untagged access ports.....	10
Enable overlay routing between virtual networks.....	11
Advertise VXLAN source IP address .....	13
Configure VLT.....	13
L3 VXLAN route scaling .....	15
DHCP relay on VTEPs .....	16
DHCP relay custom source IP.....	16
View VXLAN configuration.....	20
VXLAN MAC addresses.....	22
Example: VXLAN with static VTEP.....	24
Controller-provisioned VXLAN.....	37
Configure controller-provisioned VXLAN.....	37
Configure and control VXLAN from VMware vCenter.....	41
Example: VXLAN with a controller configuration.....	44
VXLAN Controller commands.....	48
BGP EVPN for VXLAN.....	53
BGP EVPN compared to static VXLAN.....	54
VXLAN BGP EVPN operation.....	54
Configure BGP EVPN for VXLAN.....	56
BGP EVPN with VXLAN overlay - Multi tenancy .....	60
Enabling EVPN services .....	63
EVPN constructs .....	69
BGP EVPN with VLT.....	78
ARP suppression.....	91
EVPN route selection based on AS path length.....	93
VXLAN BGP EVPN routing.....	94
Example: VXLAN with BGP EVPN with asymmetric IRB.....	98
Example: VXLAN BGP EVPN — Multiple AS topology with asymmetric IRB.....	119
Example: VXLAN BGP EVPN — Centralized L3 gateway with asymmetric IRB.....	140
Example: VXLAN BGP EVPN — Border leaf gateway with asymmetric IRB.....	142
Example: VXLAN BGP EVPN—Symmetric IRB.....	146
Example - VXLAN BGP EVPN symmetric IRB with unnumbered BGP peering.....	169
Example - Route leaking across VRFs in a VXLAN BGP EVPN symmetric IRB topology.....	184
Example: Migrating from Asymmetric IRB to Symmetric IRB.....	192
DHCP relay commands.....	195
ipv6 dhcp-relay source-interface.....	195
show ipv6 dhcp-relay.....	195
show ip dhcp-relay.....	197

show vlt mismatch dhcpv6-relay.....	200
show vlt mismatch dhcp-relay.....	201
VXLAN MAC commands.....	202
clear mac address-table dynamic nve remote-vtep.....	202
clear mac address-table dynamic virtual-network.....	202
show mac address-table count extended.....	203
show mac address-table count nve.....	203
show mac address-table count virtual-network.....	204
show mac address-table extended.....	205
show mac address-table nve.....	205
show mac address-table virtual-network.....	206
VXLAN BGP commands.....	207
arp-nd-suppression.....	207
activate (I2vpn evpn).....	207
address-family I2vpn evpn.....	208
allowas-in.....	208
sender-side-loop-detection.....	209
show evpn arp-nd-suppression.....	209
show ip bgp I2vpn evpn.....	209
VXLAN commands.....	212
arp-nd-suppression.....	212
hardware overlay-routing-profile.....	213
interface virtual-network.....	214
ip virtual-router address.....	214
ip virtual-router mac-address.....	214
member-interface.....	215
nve.....	215
remote-vtep.....	216
show hardware overlay-routing-profile mode.....	216
show interface virtual-network.....	217
show nve remote-vtep.....	217
show nve remote-vtep counters.....	218
show nve vxlan-vni.....	218
show virtual-network.....	218
show virtual-network counters.....	219
show virtual-network interface counters.....	219
show virtual-network interface.....	220
show virtual-network vlan.....	220
show vlan (virtual network).....	221
source-interface loopback.....	221
virtual-network.....	222
virtual-network untagged-vlan.....	222
vxlan-vni.....	222
VXLAN EVPN commands.....	223
advertise.....	223
auto-evi.....	224
disable-rt-asn.....	224
evi.....	225
evpn.....	225
rd.....	226

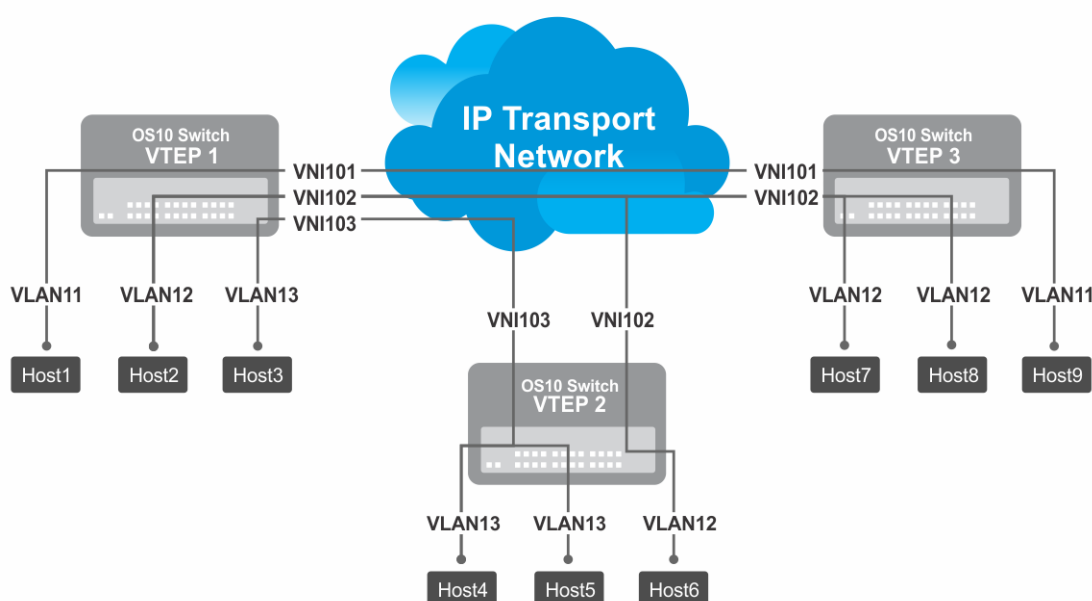
redistribute l2vpn evpn.....	226
route-target.....	227
router-mac.....	228
show evpn evi.....	228
show evpn mac.....	228
show evpn mac-ip.....	229
show evpn router-mac remote-vtep.....	231
show evpn vrf.....	231
show evpn vrf l3-vni.....	232
show evpn vxlan-vni.....	232
vni.....	233
vrf.....	233
<b>Chapter 2: Support resources.....</b>	<b>234</b>
<b>Index.....</b>	<b>235</b>

# VXLAN

A virtual extensible LAN (VXLAN) extends Layer 2 (L2) server connectivity over an underlying Layer 3 (L3) transport network in a virtualized data center. A virtualized data center consists of virtual machines (VMs) in a multitenant environment. OS10 supports VXLAN as described in RFC 7348.

VXLAN provides a L2 overlay mechanism on an existing L3 network by encapsulating the L2 frames in L3 packets. The VXLAN-shared forwarding domain allows hosts such as virtual and physical machines, in tenant L2 segments to communicate over the shared IP network. Each tenant L2 segment is identified by a 24-bit ID called a VXLAN network identifier (VNI).

Deployed as a VXLAN gateway, an OS10 switch performs encapsulation/de-encapsulation of L2 frames in L3 packets while tunneling server traffic. In this role, an OS10 switch operates as a VXLAN tunnel endpoint (VTEP). Using VXLAN tunnels, server VLAN segments communicate through the extended L2 forwarding domain.



**Figure 1. VXLAN topology**

## **NOTE:**

- The platforms that support only L2 VXLAN gateway include: S4048-ON, Z9100-ON, and Z9264F-ON
- The platforms that support both L2 VXLAN and L3 VXLAN routing (Routing In and Out of Tunnels (RIOT)) include:
  - Asymmetric IRB: S4048T-ON, S4248-ON series, S4100-ON series, S5200-ON series, and S6010-ON
  - Symmetric IRB: S4048T-ON, S4100-ON series, S5200-ON series, and S6010-ON

After VXLAN decapsulation, routing between virtual networks and regular VLANs (VLAN that is not configured as a virtual network) is supported only on the following platforms:

- S4200-ON series
- S5200-ON series

On other platforms, routing after decapsulation is performed only between virtual networks. If routing is needed for a regular VLAN after decapsulation, a virtual network should be configured instead of a regular VLAN (even though that VLAN exists only on access ports) to overcome this limitation on other platforms. On border leaf switches, an access port of this virtual network could then be connected to an external router and a protocol such as BGP or static routing could be used on this virtual network interface for external reachability.

This feature is not supported on the following platforms:

- S3048-ON
- Z9332F-ON
- N3224F-ON

**NOTE:** The Layer 2 service in the VXLAN overlay network does not participate in the Spanning tree protocol. As a result, blocking of a link in a loop-free overlay network does not prevent a loop. To prevent the network from forming loops, you can perform either of the following two actions:

- Ensure that the network topology is loop free.
- Configure BPDU guard on all the access ports.

**NOTE:** Z9100-ON platform supports 16K port, VLAN combination table (hash table), but about 90% of this table only can be used due to hash collision. Due to this hardware constraint, using whole 16K is not supported.

## VXLAN concepts

<b>Network virtualization overlay (NVO)</b>	<p>An overlay network extends L2 connectivity between server virtual machines (VMs) in a tenant segment over an underlay L3 IP network. A tenant segment can be a group of hosts or servers that are spread across an underlay network.</p> <ul style="list-style-type: none"><li>• The NVO overlay network uses a separate L2 bridge domain (virtual network), which is independent of legacy VLAN forwarding.</li><li>• The NVO underlay network operates in the default VRF using the existing L3 infrastructure and routing protocols.</li></ul>
<b>Virtual extensible LAN (VXLAN)</b>	<p>A type of network virtualization overlay that encapsulates a tenant payload into IP UDP packets for transport across the IP underlay network.</p>
<b>VXLAN network identifier (VNI)</b>	<p>A 24-bit ID number that identifies a tenant segment and transmits in a VXLAN-encapsulated packet.</p>
<b>VXLAN tunnel endpoint (VTEP)</b>	<p>A switch with connected end hosts that are assigned to virtual networks. The virtual networks map to VXLAN segments. Local and remote VTEPs perform encapsulation and de-capsulation of VXLAN headers for the traffic between end hosts. A VTEP is also known as a network virtualization edge (NVE) node.</p>
<b>Bridge domain</b>	<p>A L2 domain that receives packets from member interfaces and forwards or floods them to other member interfaces based on the destination MAC address of the packet. OS10 supports two types of bridge domains: simple VLAN and virtual network.</p> <ul style="list-style-type: none"><li>• Simple VLAN: A bridge domain a VLAN ID represents. Traffic on all member ports is assigned with the same VLAN ID.</li><li>• Virtual network: A bridge domain a virtual network ID (VNID) represents. A virtual network supports overlay encapsulation and maps with either a single VLAN ID in a <i>switch-scoped VLAN</i> or with multiple (Port,VLAN) pairs in a <i>port-scoped VLAN</i>.</li></ul>
<b>Distributed routing</b>	<p>All VTEPs in a virtual network perform intersubnet routing and serve as L3 gateways in two possible modes:</p> <ul style="list-style-type: none"><li>• Asymmetric routing: All VTEPs can perform routing. Routing decisions are made only on ingress VTEPs. Egress VTEPs perform bridging.</li><li>• Symmetric routing: All VTEPs perform routing. Routing decisions are made on both ingress and egress VTEPs.</li></ul>
<b>Virtual network</b>	<p>In OS10, each L2 flooding domain in the overlay network is represented as a <i>virtual network</i>.</p>
<b>Virtual network identifier (VNID)</b>	<p>A 16-bit ID number that identifies a virtual network in OS10.</p>
<b>Virtual-network interface</b>	<p>A router interface that connects a virtual network bridge to a tenant VRF routing instance.</p>
<b>Access port</b>	<p>A port on a VTEP switch that connects to an end host and is part of the overlay network.</p>
<b>Network port</b>	<p>A port on a VTEP switch that connects to the underlay network.</p>
<b>Switch-scoped VLAN</b>	<p>A VLAN that is mapped to a virtual network ID (VNID) in OS10. All member ports of the VLAN are automatically added to the virtual network.</p>

- You can map only one VLAN ID to a virtual network.
- Ideally suited for existing tenant VLANs that stretch over an IP fabric using VXLAN.

### Port-scoped VLAN

A Port,VLAN pair that maps to a virtual network ID (VNID) in OS10. Assign an individual member interface to a virtual network either with an associated tagged VLAN or as an untagged member. Using a port-scoped VLAN, you can configure:

- The same VLAN ID on different access interfaces to different virtual networks.
- Different VLAN IDs on different access interfaces to the same virtual network.

## VXLAN as NVO solution

Network virtualization overlay (NVO) is a solution that addresses the requirements of a multi-tenant data center, especially one with virtualized hosts. An NVO network is an overlay network that is used to extend L2 connectivity among VMs belonging to a tenant segment over an underlay IP network. Each tenant payload is encapsulated in an IP packet at the originating VTEP. To access the payload, the tenant payload is stripped of the encapsulation at the destination VTEP. Each tenant segment is also known as a *virtual-network* and is uniquely identified in OS10 using a virtual network ID (VNID).

VXLAN is a type of encapsulation used as an NVO solution. VXLAN encapsulates a tenant payload into IP UDP packets for transport across the IP underlay network. In OS10, each virtual network is assigned a 24-bit number that is called a *VXLAN network identifier* (VNI) that the VXLAN-encapsulated packet carries. The VNI uniquely identifies the tenant segment on all VTEPs. OS10 sets up ASIC tables to:

- Enables creation of a L2 bridge flooding domain across a L3 network.
- Facilitates packet forwarding between local ports and tunneling packets from the local device to a remote device.

## Configure VXLAN

To extend a L2 tenant segment using VXLAN, follow these configuration steps on each VTEP switch:

1. Configure the source IP address used in encapsulated VXLAN packets.
2. Configure a virtual network and assign a VXLAN VNI.
3. Configure VLAN-tagged access ports.
4. Configure untagged access ports.
5. (Optional) Enable routing for hosts on different virtual networks.
6. Advertise the local VXLAN source IP address to remote VTEPs.
7. (Optional) Configure VLT.

## Configure source IP address on VTEP

When you configure a switch as a VXLAN tunnel endpoint (VTEP), configure a Loopback interface, whose IP address is used as the source IP address in encapsulated packet headers. Only a Loopback interface assigned to a network virtualization edge (NVE) instance is used as a source VXLAN interface.

- Do not reconfigure the VXLAN source interface or the IP address assigned to the source interface if there is at least one VXLAN network ID (VNI) already assigned to a virtual-network ID (VNID) on the switch.
- The source Loopback IP address must be reachable from a remote VTEP.
- An IPv6 address is not supported as the source VXLAN address.
- Do not assign the source Loopback interface to a non-default VRF instance.
- Underlay reachability of remote tunnel endpoints is supported only in the default VRF.
- Do not assign the IP address that is configured as the source IP address to end hosts in any VRF.

To configure source IP address on VTEP:

1. Configure a Loopback interface to serve as the source VXLAN tunnel endpoint in CONFIGURATION mode. The range is from 0 to 255.

```
interface loopback number
```

2. Configure an IP address on the Loopback interface in INTERFACE mode. The IP address allows the source VTEP to send VXLAN frames over the L3 transport network.

```
ip address ip-address/mask
```

3. Return to CONFIGURATION mode.

```
exit
```

4. Enter NVE mode from CONFIGURATION mode. NVE mode allows you to configure the VXLAN tunnel endpoint on the switch.

```
nve
```

5. Configure the Loopback interface as the source tunnel endpoint for all virtual networks on the switch in NVE mode.

```
source-interface loopback number
```

6. Return to CONFIGURATION mode.

```
exit
```

## Configure a VXLAN virtual network

To create a VXLAN, assign a VXLAN segment ID (VNI) to a virtual network ID (VNID) and configure a remote VTEP. A unique 2-byte VNID identifies a virtual network. You cannot assign the same VXLAN VNI to more than one virtual network. Manually configure VXLAN tunnel endpoints in a static VXLAN or use BGP EVPN to automatically discover the VXLAN tunnel endpoints.

1. Create a virtual-network bridge domain in CONFIGURATION mode. Valid VNID numbers are from 1 to 65535.

```
virtual-network vn-id
```

2. Assign a VXLAN VNI to the virtual network in VIRTUAL-NETWORK mode. The range is from 1 to 16,777,215. Configure the VNI for the same tenant segment on each VTEP switch.

```
vlan-vni vni
```

3. (Optional) If you use BGP EVPN for VXLAN, this step is not required — To set up a static VXLAN, configure the source IP address of a remote VTEP in VXLAN-VNI mode. You can configure up to 1024 remote VTEP addresses for a VXLAN VNI.

```
remote-vtep ip-address
```

After you configure the remote VTEP, when the IP routing path to the remote VTEP IP address in the underlay IP network is known, the virtual network sends and receives VXLAN-encapsulated traffic from and to downstream servers and hosts. All broadcast, multicast, and unknown unicast (BUM) traffic received on access interfaces replicate and are sent to all configured remote VTEPs. Each packet contains the VXLAN VNI in its header.

By default, MAC learning from a remote VTEP is enabled and unknown unicast packets flood to all remote VTEPs. To configure additional remote VTEPs, re-enter the `remote-vtep ip-address` command.

## Configure VLAN-tagged access ports

Configure local access ports in the VXLAN overlay network using either a switch-scoped VLAN or port-scoped VLAN. Only one method is supported. You cannot assign tagged VLAN member interfaces to a virtual network using both switch-scoped and port-scoped VLANs.

- You cannot assign the same Port, VLAN member interface pair to more than one virtual network.
- You can assign the same `vlan-tag` VLAN ID with different member interfaces to different virtual networks.
- You can assign a member interface with different `vlan-tag` VLAN IDs to different virtual networks.
- To use a switch-scoped VLAN to add VLAN-tagged member ports to a virtual network:

1. Assign a VLAN to the virtual network in VLAN Interface mode.

```
interface vlan vlan-id
virtual-network vn-id
```

2. Configure port interfaces as trunk members of the VLAN in Interface mode.

```
interface ethernet node/slot/port[:subport]
switchport mode trunk
switchport trunk allowed-vlan vlan-id
exit
```

The local physical ports assigned to the VLAN transmit packets over the virtual network.

**NOTE:** A switch-scoped VLAN assigned to a virtual network cannot have a configured IP address and cannot participate in L3 routing; for example:

```
OS10(config)# interface vlan 102
OS10(conf-if-vlan-102)# ip address 1.1.1.1/24
% Error: vlan102, IP address cannot be configured for VLAN attached to Virtual
Network.
```

- To use a port-scoped VLAN to add VLAN-tagged member ports to a virtual network:

1. Configure interfaces as trunk members in Interface mode.

```
interface ethernet node/slot/port[:subport]
switchport mode trunk
exit
```

2. Assign a trunk member interface as a Port,VLAN ID pair to the virtual network in VIRTUAL-NETWORK mode. All traffic sent and received for the virtual network on the interface carries the VLAN tag. Multiple tenants connected to different switch interfaces can have the same `vlan-tag` VLAN ID.

```
virtual-network vn-id
member-interface ethernet node/slot/port[:subport] vlan-tag vlan-id
```

The Port,VLAN pair starts to transmit packets over the virtual network.

The VLAN ID tag is removed from packets transmitted in a VXLAN tunnel. Each packet is encapsulated with the VXLAN VNI in the packet header before it is sent from the egress source interface for the tunnel. At the remote VTEP, the VXLAN VNI is removed and the packet transmits on the virtual-network bridge domain. The VLAN ID regenerates using the VLAN ID associated with the virtual-network egress interface on the VTEP and is included in the packet header.

## Configure untagged access ports

Add untagged access ports to the VXLAN overlay network using either a switch-scoped VLAN or port-scoped VLAN. Only one method is supported.

- To use a switch-scoped VLAN to add untagged member ports to a virtual network:

1. Assign a VLAN to a virtual network in VLAN Interface mode.

```
interface vlan vlan-id
virtual-network vn-id
exit
```

2. Configure port interfaces as access members of the VLAN in Interface mode.

```
interface ethernet node/slot/port[:subport]
switchport access vlan vlan-id
exit
```

Packets received on the untagged ports transmit over the virtual network.

- To use a port-scoped VLAN to add untagged member ports to a virtual network:

1. Create a reserved VLAN ID to assign untagged traffic on member interfaces to a virtual network in CONFIGURATION mode. The VLAN ID is used internally for all untagged member interfaces on the switch that belong to virtual networks.

```
virtual-network untagged-vlan untagged-vlan-id
```

2. Configure port interfaces as trunk members and remove the access VLAN in Interface mode.

```
interface ethernet node/slot/port[:subport]
switchport mode trunk
no switchport access vlan
exit
```

3. Assign the trunk interfaces as untagged members of the virtual network in VIRTUAL-NETWORK mode. You cannot use the reserved VLAN ID for a legacy VLAN or for tagged traffic on member interfaces of virtual networks.

```
virtual-network vn-id
member-interface ethernet node/slot/port[:subport] untagged
exit
```

If at least one untagged member interface is assigned to a virtual network, you cannot delete the reserved untagged VLAN ID. If you reconfigure the reserved untagged VLAN ID, you must either reconfigure all untagged member interfaces in the virtual networks to use the new ID or reload the switch.

## Enable overlay routing between virtual networks

The previous sections describe how a VTEP switches traffic between hosts in the same L2 tenant segment on a virtual network, and transports traffic over an IP underlay fabric. This section describes how a VTEP enables hosts *in different* L2 segments belonging to the same tenant VRF to communicate with each other.

**NOTE:** On the S4248-ON switch, IPv6 overlay routing between virtual networks is not supported with static VXLAN. IPv6 overlay routing is, however, supported with BGP EVPN asymmetric IRB.

Each tenant is assigned a VRF and each virtual-network interface is assigned an IP subnet in the tenant VRF. The VTEP acts as the L3 gateway that routes traffic from one tenant subnet to another in the overlay before encapsulating it in the VXLAN header and transporting it over the IP underlay fabric.

To enable host traffic routing between virtual networks, configure an interface for each virtual network and associate it to a tenant VRF. Assign a unique IP address in the IP subnet range associated with the virtual network to each virtual-network interface on each VTEP.

To enable efficient traffic forwarding on a VTEP, OS10 supports distributed and centralized gateway routing. A distributed gateway means that multiple VTEPs act as the gateway router for a tenant subnet. The VTEP nearest to a host acts as its gateway router. To support seamless migration of hosts and virtual machines on different VTEPs, configure a common virtual IP address, known as an anycast IP address, on all VTEPs for each virtual network. Use this anycast IP address as the gateway IP address on VMs.

To support multiple tenants when each tenant has its own L2 segments, configure a different IP VRF for each tenant. All tenants share the same VXLAN underlay IP fabric in the default VRF.

1. Create a non-default VRF instance for overlay routing in Configuration mode. For multi-tenancy, create a VRF instance for each tenant.

```
ip vrf tenant-vrf-name
exit
```

2. Configure the anycast gateway MAC address all VTEPs use in all VXLAN virtual networks in Configuration mode.

When a VM sends an Address Resolution Protocol (ARP) request for the anycast gateway IP address in a VXLAN virtual network, the nearest VTEP responds with the configured anycast MAC address. Configure the same MAC address on all VTEPs so that the anycast gateway MAC address remains the same if a VM migrates to a different VTEP. Because the configured MAC address is automatically used for all VXLAN virtual networks, configure it in global Configuration mode.

```
ip virtual-router mac-address mac-address
```

Example:

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

3. Configure a virtual-network interface, assign it to the tenant VRF, and configure an IP address.

The interface IP address must be unique on each VTEP, including VTEPs in VLT pairs. You can configure an IPv6 address on the virtual-network interface. Different virtual-network interfaces you configure on the same VTEP must have virtual-

network IP addresses in different subnets. If you do not assign the virtual-network interface to a tenant VRF, it is assigned to the default VRF.

```
interface virtual-network vn-id
ip vrf forwarding tenant-vrf-name
ip address ip-address/mask
no shutdown
exit
```

4. Configure an anycast gateway IPv4 or IPv6 address for each virtual network in INTERFACE-VIRTUAL-NETWORK mode. This anycast IP address must be in the same subnet as the IP address of the virtual-network interface in Step 3.

Configure the same IPv4 or IPv6 address as the anycast IP address on all VTEPs in a virtual network. All hosts use the anycast gateway IP address as the default gateway IP address in the subnet that connects to the virtual-network interface configured in Step 3. Configure the anycast gateway IP address on all downstream VMs. Using the same anycast gateway IP address allows host VMs to move from one VTEP to another VTEP in a VXLAN. Dell Technologies recommends using an anycast gateway in both VLT and non-VLT VXLAN configurations.

```
interface virtual-network vn-id
ip virtual-router address ip-address
```

#### Configuration notes for virtual-network routing:

- VXLAN overlay routing includes routing tenant traffic on the ingress VTEP and bridging the traffic on the egress VTEP. The ingress VTEP learns ARP entries and associates all destination IP addresses of tenant VMs with the corresponding VM MAC addresses in the overlay. On the ingress VTEP, configure a virtual network for each destination IP subnet even if there are no locally attached hosts for an IP subnet.
- Open Shortest Path First (OSPF) routing protocol is not supported on the virtual-network interface in the overlay network. However, BGP and static routes that point to a virtual-network interface or to a next-hop IP address that belongs to a virtual-network subnet are supported.
- VLT peer routing is not supported in a virtual network. A packet destined to the virtual-network peer MAC address L2 switches instead of IP routes. To achieve active-active peer routing in a virtual network, configure the same virtual anycast gateway IP and MAC addresses on both VTEP VLT peers and use the anycast IP as the default gateway on the VMs.
- Virtual Router Redundancy Protocol (VRRP) is not supported on a virtual-network interface. Configure the virtual anycast gateway IP address to share a single gateway IP address on both VTEP VLT peers and use the anycast IP as default gateway on the VMs.
- Internet Group Management Protocol (IGMP) and Protocol-Independent Multicast (PIM) are not supported on a virtual-network interface.

The following tables show how to use anycast gateway IP and MAC addresses in a data center with three virtual networks and multiple VTEPs:

- Globally configure an anycast MAC address for all VTEPs in all virtual networks. For example, if you use three VTEP switches in three virtual networks:

**Table 1. MAC address for all VTEPs**

Virtual network	VTEP	Anycast gateway MAC address
VNID 11	VTEP 1	00.11.22.33.44.55
	VTEP 2	00.11.22.33.44.55
	VTEP 3	00.11.22.33.44.55
VNID 12	VTEP 1	00.11.22.33.44.55
	VTEP 2	00.11.22.33.44.55
	VTEP 3	00.11.22.33.44.55
VNID 13	VTEP 1	00.11.22.33.44.55
	VTEP 2	00.11.22.33.44.55
	VTEP 3	00.11.22.33.44.55

- Configure a unique IP address on the virtual-network interface on each VTEP across all virtual networks. Configure the same anycast gateway IP address on all VTEPs in a virtual-network subnet. For example:

**Table 2. IP address on the virtual-network interface on each VTEP**

Virtual network	VTEP	Virtual-network IP address	Anycast gateway IP address
VNID 11	VTEP 1	10.10.1.201	10.10.1.254
	VTEP 2	10.10.1.202	10.10.1.254
	VTEP 3	10.10.1.203	10.10.1.254
VNID 12	VTEP 1	10.20.1.201	10.20.1.254
	VTEP 2	10.20.1.202	10.20.1.254
	VTEP 3	10.20.1.203	10.20.1.254
VNID 13	VTEP 1	10.30.1.201	10.30.1.254
	VTEP 2	10.30.1.202	10.30.1.254
	VTEP 3	10.30.1.203	10.30.1.254

### Configuration notes

In a static VXLAN, overlay routing is supported on:

- S4100-ON Series
- S4200-ON Series
- S5200-ON Series
- S4048T-ON
- S6010-ON

## Advertise VXLAN source IP address

1. Advertise the IP address of the local source tunnel interface to all VTEPs in the underlay IP network using the existing routing infrastructure. This example uses OSPF to advertise the VXLAN source IP address on Ethernet1/1/3, which is the underlay network-facing interface:

```
OS10(config)# router ospf 100
OS10(config-ospf)# router-id 110.111.170.195
OS10(config-ospf)# exit
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# ip ospf 100 area 0.0.0.0
OS10(config-if-eth1/1/3)# exit
OS10(config)# interface loopback 1
OS10(config-if-lo-1)# ip ospf 100 area 0.0.0.0
```

Each VTEP switch in the underlay IP network learns the IP address of the VXLAN source interface. If a remote VTEP switch is not reachable, its status displays as `DOWN` in the `show nve remote-vtep` output.

2. Configure the MTU value on L3 underlay network-facing interfaces in Interface mode to be at least 50 bytes higher than the MTU on the server-facing links to allow for VXLAN encapsulation. The range is from 1312 to 9216.

```
mtu value
```

3. Return to CONFIGURATION mode.

```
exit
```

## Configure VLT

(Optional) To use VXLAN in a VLT domain, configure the VLT domain — including the VLT Interconnect (VLTi) interfaces, backup heartbeat, and VLT MAC address — as described in the *OS10 Enterprise Edition User Guide* in the *Virtual link trunking* section.

Required VLT VXLAN configuration:

- The IP address of the VTEP source Loopback interface must be same on the VLT peers.
- If you use a port-scoped VLAN to assign tagged access interfaces to a virtual network, to identify traffic belonging to each virtual network, you must configure a unique VLAN ID for the VLT Interconnect (VLTi) link.
- Configure a VLAN to transmit VXLAN traffic over the VLTi link in VIRTUAL-NETWORK mode. All traffic sent and received from a virtual network on the VLTi carries the VLTi VLAN ID tag.

Configure the same VLTi VLAN ID on both VLT peers. You cannot use the ID of an existing VLAN on a VLT peer or the reserved untagged VLAN ID. You can use the VLTi VLAN ID to assign tagged or untagged access interfaces to a virtual network.

```
virtual-network vn-id
vlti-vlan vlan-id
```

- Although a VXLAN virtual network has no access port members that connect to downstream servers, you must configure a switch-scoped VLAN or VLTi VLAN. The presence of this VLAN ensures that the VLTi link is added as a member of the virtual network so that mis-hashed ARP packets received from the VXLAN tunnel reach the intended VLT node.

#### Best practices:

- If a VLT peer loses connectivity to the underlay L3 network, it continues to transmit routing traffic to the network through the VLTi link on a dedicated L3 VLAN to the other VLT peer. Configure a L3 VLAN between VLT peers in the underlay network and enable routing on the VLAN; for example:

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 41.1.1.1/24
OS10(config-if-vl-4000)# ip ospf 1 area 0.0.0.0
```

- To reduce traffic loss when a VLT peer boots up and joins an existing VLT domain, or when the VLTi links fails and the VLT peer is still up as detected by the VLT heartbeat, create an uplink state group. Configure all access VLT port channels on the peer as upstream links. Configure all network-facing links as downstream link. For example:

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel 10
```

#### Configuration notes

All Dell PowerSwitches except S4200-Series, S5200 Series, and Z9332F-ON:

To check mismatch of MAC address table entries between VLT peers, use the `show vlt mac-inconsistency` command. To identify mismatches in VLT configuration on peer switches, use the `show vlt domain-name mismatch` command.

```
OS10# show vlt-mac-inconsistency
Checking Vlan 228 .. Found 7 inconsistencies .. Progress 100%
VLAN 128
-----
MAC 00:00:00:00:00:02 is missing from Node(s) 2
VLAN 1
-----
MAC 00:a0:c9:00:00:18 is missing from Node(s) 2
MAC 00:a0:c9:00:00:20 is missing from Node(s) 2
VLAN 131
-----
MAC 00:00:00:00:00:02 is missing from Node(s) 2
VLAN 132
-----
MAC 00:00:00:00:00:02 is missing from Node(s) 2
VLAN 135
-----
MAC 00:00:00:00:00:02 is missing from Node(s) 2
VLAN 137
-----
MAC 00:00:00:00:00:02 is missing from Node(s) 2
Run "show vlt dl mismatch ..." commands to identify configuration issues
```

## L3 VXLAN route scaling

The S4100-ON series, S5200-ON series, S4048T-ON, S4248-ON series, and S6010-ON switches support native VxLAN routing — routing in and out of tunnels (RIOT). RIOT requires dedicated hardware resources reserved for overlay routing. You cannot use these dedicated resources for underlay routing.

Each overlay ARP entry requires a routing next-hop in the hardware to bind a destination tenant VM IP address to the corresponding tenant VM MAC address and VNI. Each virtual-network interface assigned to an IP subnet requires a routing interface in the hardware.

OS10 supports preset profiles to re-allocate the number of resources reserved for overlay ARP entries. The number of entries reserved for each preset mode differs according to OS10 switch.

**Table 3. Routing next-hops reserved on OS10 switches**

OS10 Switch	Overlay next-hop entries	Underlay next-hop entries	Overlay L3 RIF entries	Underlay L3 RIF entries
S41xx-ON series:	—	—	—	—
default-overlay-routing	4096	28672	2048	10240
disable-overlay-routing	0	32768	0	12288
balanced-overlay-routing	16384	16384	6144	6144
scaled-overlay-routing	24576	8192	10240	2048
S4048T-ON and S6010-ON:	—	—	—	—
default-overlay-routing	8192	4096	2048	2048
disable-overlay-routing	0	49152	49152	0
balanced-overlay-routing	24576	24576	24576	6144
scaled-overlay-routing	40960	8192	8192	10240
S52xx-ON series:	—	—	—	—
default-overlay-routing	8192	57344	2048	14336
disable-overlay-routing	0	65536	0	16384
balanced-overlay-routing	32768	32768	8192	8192
scaled-overlay-routing	53248	12288	12288	4096
S4248-ON:	—	—	—	—
default-overlay-routing	20480	110592	4096	28672

**NOTE:** The S4248-ON switch supports only one default profile to reserve resources for overlay ARP entries.

To activate the profile after you configure an overlay routing profile, save the configuration and reload the switch.

### Configure an overlay routing profile

- Enable an overlay routing profile in Configuration mode or disable the configured profile and return to the default.

```
OS10(config)# hardware overlay-routing-profile {disable-overlay-routing | balanced-
overlay-routing |
scaled-overlay-routing}
```

### Display overlay routing profiles

- View the hardware resources available for overlay routing in different profiles; for example, in the S5200-ON series:

```
OS10# show hardware overlay-routing-profile mode all
Mode                Overlay Next-hop  Underlay Next-hop  Overlay L3 RIF
Underlay L3 RIF
                   Entries                Entries                Entries                Entries
```

default-overlay-routing	8192	57344	2048	14336
disable-overlay-routing	0	65536	0	16384
balanced-overlay-routing	32768	32768	8192	8192
scaled-overlay-routing	53248	12288	12288	4096

- View the currently configured overlay routing profile; for example, in the S5200-ON series:

```
show hardware overlay-routing-profile mode
```

Setting	Mode	Overlay Next-hop Entries	Underlay Next-hop Entries	Overlay L3 RIF Entries	Underlay L3 RIF Entries
Current	default-overlay-routing	8192	57344	2048	14336
Next-boot	default-overlay-routing	8192	57344	2048	14336

## DHCP relay on VTEPs

Dynamic Host Configuration Protocol (DHCP) clients in overlay communicate with a DHCP server using the DHCP relay on the VTEP switch. DHCP server and the client can reside in the same VRF or in different VRFs. If they are in different VRFs, configure the route-leaking to allow communication between the client subnet in the client-VRF and the server in the server-VRF. If they are in the same VRF, route leaking need not be configured.

DHCPv4 relay on VTEPs supports the following option 82 sub-options:

- Server ID override suboption - Sub-option 11 (0xb)
- Link selection suboption- Sub-option 5 (0x5)
- DHCPv4 virtual subnet selection option - Sub-option 151 (0x97)
- DHCPv4 virtual subnet selection control - Sub-option 152 (0x98)
- source-interface CLI for relay agents. The gateway address (`giaddr`) field carries the source interface address.

Use the Link selection suboption, Server ID override suboption, and source-interface to minimize the route leaking configurations. Only the DHCP server subnet needs to be leaked into client-VRF and the DHCP client-subnets in client-VRF need not be leaked into server-VRF. The source-interface must be reachable from the server-VRF, and the DHCP server sends responses to the source-interface IP.

Use the VSS suboption to send the configured client VRF information to the DHCP server to allocate an IP address based on the VRF.

### Configure DHCP relay on VTEPs

To configure DHCP relay on the virtual-network interface of the tenant VRF, run the following commands:

```
OS10(config)# interface virtual-network 10
OS10(conf-if-vn-10)# ip helper-address 40.1.1.1 vrf tenant01
```

## DHCP relay custom source IP

A DHCP relay agent is any host that forwards DHCP packets between clients and servers. In SmartFabric DellEMC OS10, DHCP relay agents forward a DHCP client packet with the source IP address set to the IP address of the outgoing interface of the DHCP server. This feature provides the capability to configure the source IP address for the DHCP requests.

The DHCPv6 relay agent source-interface configuration supports the following interface types:

- ethernet
- loopback
- port-channel
- VLAN
- virtual-network

## Restrictions and limitations

The following restrictions and limitations apply to this feature:

- The DHCPv6 relay agent source-interface configuration is supported at the interface level. SmartFabric DellEMC OS10 recommends that the specified source interface should belong to the server connected interface VRF.
- SmartFabric DellEMC OS10 recommends to configure different source interface IP addresses in VLT peers. Synchronization for DHCPv6 functionality across VLT peers is not supported.
- The DHCPv6 relay agent source-interface configuration is not supported at the VRF level when applied in batch mode. The IP VRF context appears at the beginning of running configuration file even before the loopback interface is configured. As a result, the IP VRF configurations fail to execute as the loopback interfaces do not exist. This behavior is similar to the update-source interface configuration command.

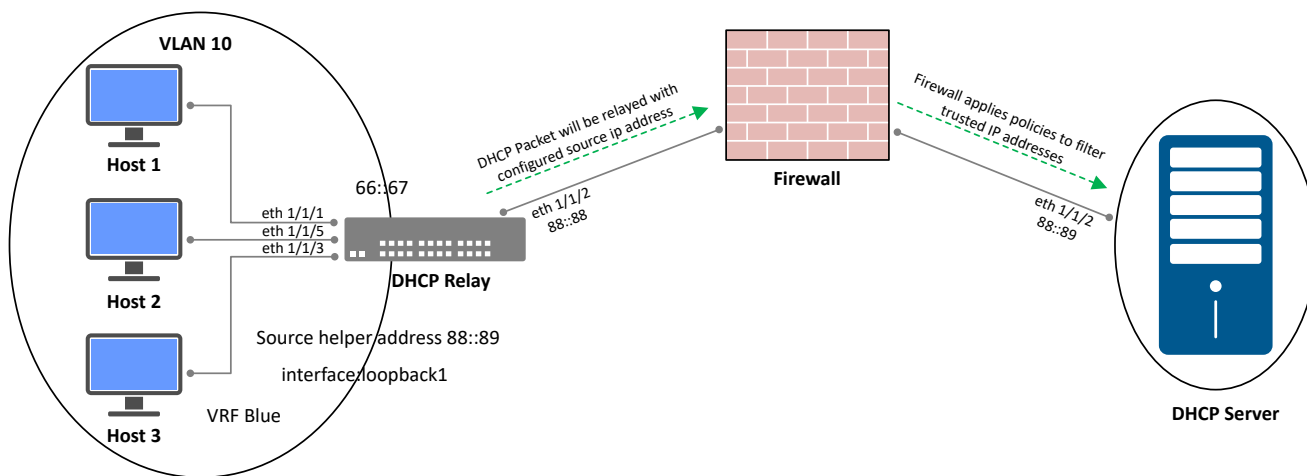
## Impact on software upgrade and downgrade

**NOTE:** The existing IPv4 source interface is used to fill relay agent gateway IP address. After upgrade, the same IPv4 source interface is used to fill the relay agent gateway IP address along with the source IP header.

## System operation and behavior

In network configurations where a firewall gateway exists between the DHCP relay agent and the DHCP server, only the trusted IP addresses pass through this firewall. In such a scenario, the DHCP unicast packets do not pass through the firewall and are discarded.

The DHCP relay agent source-interface IP configuration places the loopback address in the IP headers and DHCP messages. This action enables the DHCP unicast packets to pass through the firewall to the DHCP server.



Following is the format of the source-interface value with interface loopback and Ethernet:

```
loopback as interface :
  <loopback> <loopback_interface_no>
  e.g: loopback_interface_no ranges from <0-16383>
  e.g.: ipv6 dhcp-relay source-interface loopback 1

Ethernet as interface :
  <ethernet > <node/slot/port[:subport]>
  e.g.: ipv6 dhcp-relay source-interface ethernet 1/1/1
```

## DHCP relay agent - Client interfacing interface configuration

```
OS10-Relay(conf-if-eth1/1/1)# show configuration
!
interface ethernet1/1/1
no shutdown
no switchport
ipv6 address 66::67/64
flowcontrol receive off
ipv6 helper-address 88::89
ipv6 dhcp-relay source-interface loopback1
OS10-Relay(conf-if-eth1/1/1)#
```

## DHCP relay agent - Server interfacing interface configuration

```
OS10-Relay(conf-if-eth1/1/2)# show configuration
!
interface ethernet1/1/2
no shutdown
no switchport
ipv6 address 88::88/64
flowcontrol receive off
OS10-Relay(conf-if-eth1/1/2)#
```

## DHCP server configuration

```
OS10-Server(conf-if-eth1/1/2)# show configuration
!
interface ethernet1/1/2
no shutdown
no switchport
ipv6 address 88::89/64
flowcontrol receive off
OS10-Server(conf-if-eth1/1/2)#

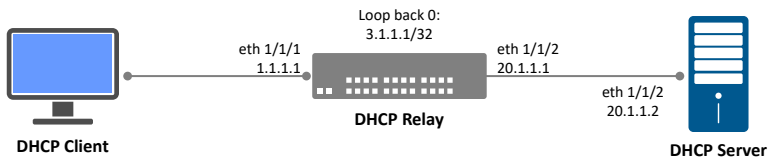
OS10-Server(config-dhcp-pool1)# show configuration
!
pool pool1
network 66::60/64
OS10-Server(config-dhcp-pool1)#
```

## DHCP client configuration

```
OS10-Client(conf-if-eth1/1/1)# show configuration
!
interface ethernet1/1/1
no shutdown
no switchport
ipv6 address dhcp
flowcontrol receive off
```

## Usecase - Example for source-interface command for IPv4

In the following example, the DHCP client is connected on Eth 1/1/1.



Loopback 0 is used as the relay source-interface for the default VRF clients.

```
Gi addr ( source-interface) - 3.1.1.1
```

## DHCP client

```
Client(conf-if-eth1/1/1)# show configuration
!
interface ethernet1/1/1
no shutdown
no switchport
ip address dhcp
flowcontrol receive off
Client(conf-if-eth1/1/1)#
```

## DHCP server

```
OS10(config)# ip dhcp server
OS10(config-dhcp)# pool dell_1
OS10(config-dhcp-dell_1)# network 1.1.1.0/24
OS10(config-dhcp-dell_1)# range 1.1.1.2 1.1.1.10
Server(config-dhcp-dell_1)# show configuration
!
pool dell_1
network 1.1.1.0/24
range 1.1.1.2 1.1.1.10
Server(config-dhcp-dell_1)#

Server(conf-if-eth1/1/2)# show configuration
!
interface ethernet1/1/2
no shutdown
no switchport
ip address 20.1.1.2/24
flowcontrol receive off
Server(conf-if-eth1/1/2)#
```

## DHCP relay

```
Relay(config)# interface loopback 0
Relay(conf-if-lo-0)# show configuration
!
interface loopback0
no shutdown
Relay(conf-if-lo-0)# ip address 3.1.1.1/32
Relay(conf-if-lo-0)# show configuration
!
interface loopback0
no shutdown
ip address 3.1.1.1/32
Relay(conf-if-lo-0)#

Relay(conf-if-eth1/1/1)# show configuration
```

```

!
interface ethernet1/1/1
no shutdown
no switchport
ip address 1.1.1.1/24
flowcontrol receive off
ip helper-address 20.1.1.2
ip dhcp-relay source-interface loopback1

Relay(conf-if-eth1/1/2)# show configuration
!
interface ethernet1/1/2
no shutdown
no switchport
ip address 20.1.1.1/24
flowcontrol receive off
Relay(conf-if-eth1/1/2)#

```

## View VXLAN configuration

Use show commands to verify the VXLAN configuration and monitor VXLAN operation.

### View the VXLAN virtual network

```

OS10# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
Un-tagged VLAN: 888
Virtual Network: 60000
  VLTi-VLAN: 2500
Members:
  VLAN 1000: port-channell, ethernet1/1/9, ethernet1/1/10
  VLAN 2500: port-channell000
VxLAN Virtual Network Identifier: 16775000
Source Interface: loopback100(222.222.222.222)
Remote-VTEPs (flood-list): 55.55.55.55(DP),77.1.1.1(DP)

```

### View the VXLAN virtual-network port

```

OS10# show virtual-network interface ethernet 1/1/1
Interface      Vlan      Virtual-network
ethernet1/1/1  100       1000
ethernet1/1/1  200       2000
ethernet1/1/1  300       3000

```

### View the VXLAN virtual-network VLAN

```

OS10# show virtual-network vlan 100
Vlan  Virtual-network  Interface
100   1000               ethernet1/1/1,ethernet1/1/2
100   5000               ethernet1/1/2

```

### View the VXLAN virtual-network VLANs

```

OS10# show vlan
Codes: * - Default VLAN, M - Management VLAN, R - Remote Port Mirroring VLANs,
@ - Attached to Virtual Network
Q: A - Access (Untagged), T - Tagged

  NUM   Status Description  Q Ports
* 1     up          A Eth1/1/1-1/1/48
@ 100   up          T Eth1/1/2,Eth1/1/3
        A Eth1/1/1
@ 101   up          T port-channel5
200     up          T Eth1/1/11-1/1/15

```

## View the VXLAN virtual-network statistics

```
OS10# show virtual-network counters
Virtual-Network      Input (Packets/Bytes)      Output (Packets/Bytes)
1000                 857/8570                  257/23709
2000                 457/3570                  277/13709
```

```
OS10# show virtual-network counters interface 1/1/3 vlan 100
Virtual-Network      Input (Packets/Bytes)      Output (Packets/Bytes)
1000                 857/8570                  257/23709
2000                 457/3570                  277/13709
```

**i** **NOTE:** Using flex counters, OS10 may display additional packets in the Output field number, but the additional packets do not transmit. For an accurate count, use the Output Bytes number.

## View the VXLAN remote VTEPs

```
OS10# show nve remote-vtep summary
Remote-VTEP      State
-----
2.2.2.2          up
```

```
OS10# show nve remote-vtep
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
IP Address: 2.2.2.2, State: up, Encap: VxLAN
VNI list: 10000 (DP), 200 (DP), 300 (DP)
```

The `show nve remote-vtep` command displays the L2 VxLAN VNI and the EVPN VRF L3 VNI spanned with the other remote VTEPs.

```
OS10# show nve remote-vtep
IP Address: 31.1.1.1, State: up, Encap: VxLAN
VNI list: 100-101,200,300,400,500,600,700,800
L3VNI list: 65000
```

The `show nve remote-vtep summary` command displays the state of the remote VTEP as up when an L2 VxLAN VNI or an EVPN VRF L3 VNI is spanned with the specific remote VTEP.

```
OS10# show nve remote-vtep summary
Remote-VTEP State
-----
31.1.1.1 up
```

## View the VXLAN statistics on the remote VTEPs

The `show nve remote-vtep counters` command displays the packet counters and byte counter statistics for a specific remote VTEP. The counters for a remote VTEP include both the counters corresponding to the L2 VNI spanned with the VTEP as well as the EVPN-VRF L3 VNI spanned with the VTEP.

```
OS10# show nve remote-vtep counters
Remote-VTEP      Input (Packets/Bytes)      Output (Packets/Bytes)
-----
10.10.10.10      857/8570                  257/23709
20.20.20.20      457/3570                  277/13709
```

```
OS10# show nve remote-vtep 31.1.1.1 counters
Remote-VTEP: 31.1.1.1
Input (Packets/Bytes) : 277/38776
Output (Packets/Bytes) : 261/36400
```

## View the VXLAN virtual network by VNID

```
OS10# show nve vxlan-vni
VNI      Virtual-Network      Source-IP      Remote-VTEPs
-----
101      101                  44.44.44.44    11.11.11.11,22.22.22.22,33.33.33.33
102      102                  44.44.44.44    11.11.11.11,22.22.22.22,33.33.33.33
```

103	103	44.44.44.44	11.11.11.11,22.22.22.22,33.33.33.33
104	104	44.44.44.44	11.11.11.11,22.22.22.22,33.33.33.33

**View VXLAN routing between virtual networks**

The `show ip arp vrf` and `show ipv6 neighbors vrf` command output displays information about IPv4 and IPv6 neighbors learned in a non-default VRF on the switch. The `show ip route vrf` command displays the IPv4 and IPv6 routes learned.

```
OS10# show ip arp vrf tenant1
Address      Hardware address  Interface          Egress Interface
-----
111.0.0.2    00:c5:15:02:12:f1  virtual-network20  ethernet1/1/5
111.0.0.3    00:c5:15:02:12:a2  virtual-network20  port-channel5
111.0.0.4    00:12:98:1f:34:11  virtual-network20  VXLAN(20.0.0.1)
121.0.0.3    00:12:28:1f:34:15  virtual-network20  port-channel5
121.0.0.4    00:f2:34:ac:34:09  virtual-network20  VXLAN(20.0.0.1)

OS10# show ipv6 neighbors vrf tenant1
IPv6 Address  Hardware Address  State  Interface          Egress Interface
-----
200::2        00:12:28:1f:34:15  STALE  virtual-network40  port-channel5
200::f        00:f2:34:ac:34:09  REACH  virtual-network40  VXLAN(20.0.0.1)
```

```
OS10# show ip route vrf vrf_1
Codes: C - connected
       S - static
       B - BGP, IN - internal BGP, EX - external BGP
       O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
       N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
       E2 - OSPF external type 2, * - candidate default,
       + - summary route, > - non-active route
Gateway of last resort is not set
Destination      Gateway          Dist/Metric      Last Change
-----
C 100.1.0.0/16    via 100.1.1.4   virtual-network60000  0/0  00:36:24
C 100.33.0.0/16   via 100.33.1.4  virtual-network60032  0/0  00:36:23
C 100.65.0.0/16   via 100.65.1.4  virtual-network60064  0/0  00:36:22
C 100.97.0.0/16   via 100.97.1.4  virtual-network60096  0/0  00:36:21

OS10# show ipv6 route vrf vrf_1
Codes: C - connected
       S - static
       B - BGP, IN - internal BGP, EX - external BGP
       O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
       N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
       E2 - OSPF external type 2, * - candidate default,
       + - summary route, > - non-active route
Gateway of last resort is not set
Destination      Gateway          Dist/Metric      Last Change
-----
C 1000:100:10:1::/64  via 1000:100:10:1::4  virtual-network60000  0/0  00:37:08
C 1000:100:10:21::/64 via 1000:100:10:21::4  virtual-network60032  0/0  00:37:07
C 1000:100:10:41::/64 via 1000:100:10:41::4  virtual-network60064  0/0  00:37:06
C 1000:100:10:61::/64 via 1000:100:10:61::4  virtual-network60096  0/0  00:37:05
```

## VXLAN MAC addresses

Use the `show mac address-table virtual-network` or `show mac address-table extended` commands to display the MAC addresses learned on a VXLAN virtual network or learned on both VXLAN virtual networks and legacy VLANs.

Use the `clear mac address-table dynamic virtual-network` and `clear mac address-table dynamic nve remote-vtep` commands to delete address entries from the MAC address virtual-network table.

**NOTE:** The existing `show mac address-table` and `clear mac-address table` commands do not display and clear MAC addresses in a virtual-network bridge domain even when access ports in a switch-scoped VLAN are assigned to a VXLAN virtual network.

## Display VXLAN MAC addresses

**Table 4. Display VXLAN MAC addresses**

Command	Description
<pre>show mac address-table virtual-network [vn-id   local   remote   static   dynamic   address mac-address   interface {ethernet node/slot/ port:subport   port-channel number}]</pre>	<p>Displays all MAC addresses learned on all or a specified virtual network.</p> <p><i>vn-id</i>: Displays only information about the specified virtual network.</p> <p><i>local</i>: Displays only locally-learned MAC addresses.</p> <p><i>remote</i>: Displays only remote MAC addresses.</p> <p><i>static</i>: Displays only static MAC addresses.</p> <p><i>dynamic</i>: Displays only dynamic MAC addresses.</p> <p><i>address mac-address</i>: Displays only information about the specified MAC address.</p> <p><i>interface ethernet node/slot/port:subport</i>: Displays only MAC addresses learned on the specified interface.</p> <p><i>interface port-channel number</i>: Displays only MAC addresses learned on the specified port channel.</p>
<pre>show mac address-table extended [address mac-address   interface {ethernet node/slot/ port:subport   port-channel number}   static   dynamic]</pre>	<p>Displays MAC addresses learned on all VLANs and VXLANs (default).</p> <p><i>address mac-address</i>: Displays only information about the specified MAC address.</p> <p><i>interface ethernet node/slot/port:subport</i>: Displays only MAC addresses learned on the specified interface.</p> <p><i>interface port-channel number</i>: Displays only MAC addresses learned on the specified port channel.</p> <p><i>static</i>: Displays only static MAC addresses.</p> <p><i>dynamic</i>: Displays only dynamic MAC addresses.</p>
<pre>show mac address-table nve {vxlan-vni vn-id   remote-vtep ip-address}</pre>	<p><i>vxlan-vni vn-id</i>: Displays MAC addresses learned on NVE from the specified VXLAN virtual-network ID.</p> <p><i>remote-vtep ip-address</i>: Displays MAC addresses learned on NVE from the specified remote VTEP.</p>
<pre>show mac address-table count virtual-network [dynamic   local   remote   static   interface {ethernet node/slot/port:subport   port-channel number}   vn-id]</pre>	<p>Displays the number of MAC addresses learned on all virtual networks (default).</p> <p><i>dynamic</i>: Displays the number of dynamic MAC addresses learned on all or a specified virtual network.</p> <p><i>local</i>: Displays the number of locally-learned MAC addresses.</p> <p><i>remote</i>: Displays the number of remote MAC addresses learned on all or a specified virtual network.</p> <p><i>static</i>: Displays the number of static MAC addresses learned on all or a specified virtual network.</p> <p><i>interface ethernet node/slot/port:subport</i>: Displays the number of MAC addresses learned on the specified interface.</p>

**Table 4. Display VXLAN MAC addresses (continued)**

Command	Description
	<p><code>interface port-channel number</code>: Displays the number of MAC addresses learned on the specified port channel.</p> <p><code>vn-id</code>: Displays the number of MAC addresses learned on the specified virtual network.</p>
<pre>show mac address-table count nve {remote-vtep ip-address   vxlan-vni vn-id}</pre>	<p>Displays the number of MAC addresses learned for a virtual network or from a remote VTEP.</p> <p><code>remote-vtep ip-address</code>: Displays the number of MAC addresses learned on the specified remote VTEP.</p> <p><code>vxlan-vni vn-id</code>: Displays the number of MAC addresses learned on the specified VXLAN virtual network.</p>
<pre>show mac address-table count extended [interface ethernet node/slot/port:subport   port-channel number]</pre>	<p>Displays the number of MAC addresses learned on all VLANs and VXLAN virtual networks.</p> <p><code>interface ethernet node/slot/port:subport</code>: Displays the number of MAC addresses learned from VLANs and VXLANs on the specified interface.</p> <p><code>port-channel number</code>: Displays the number of MAC addresses learned from VLANs and VXLANs on the specified port channel.</p>

**Clear VXLAN MAC addresses**

**Table 5. Clear VXLAN MAC addresses**

Command	Description
<pre>clear mac address-table dynamic virtual-network [interface {ethernet node/slot/port:subport   port-channel number}   local   vn-id [address mac-address   local]]</pre>	<p>Clears all MAC addresses learned on all VXLAN virtual networks.</p> <p><code>interface ethernet node/slot/port:subport</code>: Clears only MAC addresses learned on the specified interface.</p> <p><code>interface port-channel number</code>: Clears only MAC addresses learned on the specified port channel.</p> <p><code>local</code>: Clears only locally-learned MAC addresses.</p> <p><code>vn-id</code>: Clears only the MAC addresses learned on the specified virtual network.</p> <p><code>vn-id address mac-address</code>: Clears only the MAC address learned on the specified virtual network.</p>
<pre>clear mac address-table dynamic nve remote-vtep ip-address</pre>	<p>Clears all MAC addresses learned from the specified remote VTEP.</p>

## Example: VXLAN with static VTEP

This example uses a typical Clos leaf-spine topology with static VXLAN tunnel endpoints (VTEPs) in VLT dual-homing domains. The individual switch configuration shows how to set up an end-to-end VXLAN. The underlay IP network routes advertise using OSPF.

- On VTEPs 1 and 2, access ports are assigned to the virtual network using a switch-scoped VLAN configuration.
- On VTEPs 3 and 4, access ports are assigned to the virtual network using a port-scoped VLAN configuration.
- Overlay routing between hosts in different IP subnets is configured on the VTEPs.

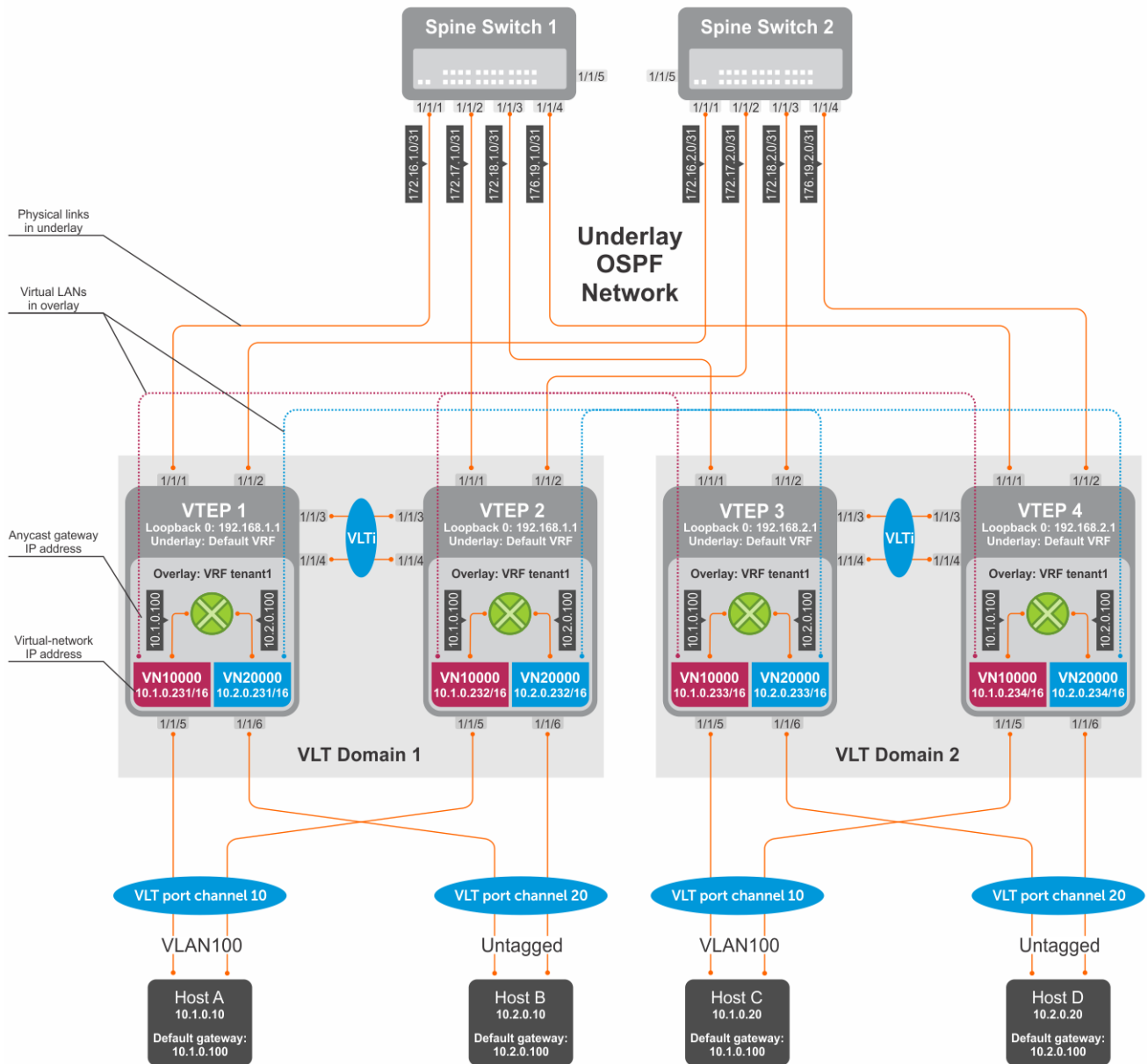


Figure 2. Static VXLAN use case

## VTEP 1 Leaf Switch

### 1. Configure the underlay OSPF protocol.

Do not configure the same IP address for the router ID and the source loopback interface in Step 2.

```
OS10(config)# router ospf 1
OS10(config-router-ospf-1)# router-id 172.16.0.1
OS10(config-router-ospf-1)# exit
```

### 2. Configure a Loopback interface.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# ip ospf 1 area 0.0.0.0
OS10(conf-if-lo-0)# exit
```

### 3. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 4. Configure VXLAN virtual networks with a static VTEP.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.2.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.2.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 5. Assign VLAN member interfaces to virtual networks.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
OS10(config)# interface vlan200
OS10(config-if-vl-100)# virtual-network 20000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
```

### 6. Configure access ports as VLAN members for switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
OS10(conf-if-po-10)# switchport trunk allowed vlan 100
OS10(conf-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# no switchport
OS10(conf-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# no shutdown
OS10(conf-if-po-20)# switchport access vlan 200
OS10(conf-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(conf-if-eth1/1/6)# no shutdown
OS10(conf-if-eth1/1/6)# channel-group 20 mode active
OS10(conf-if-eth1/1/6)# no switchport
OS10(conf-if-eth1/1/6)# exit
```

### 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.16.1.0/31
OS10(conf-if-eth1/1/1)# ip ospf 1 area 0.0.0.0
OS10(conf-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
```

```
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.16.2.0/31
OS10(config-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/2)# exit
```

## 8. Configure VLT

**Configure a dedicated L3 underlay path to reach the VLT Peer in case of network failure.**

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.1/30
OS10(config-if-vl-4000)# ip ospf 1 area 0.0.0.0
OS10(config-if-vl-4000)# exit
```

**Configure the VLT port channel.**

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt-port-channel 10
OS10(config-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt-port-channel 20
OS10(config-if-po-20)# exit
```

**Configure the VLTi member links.**

```
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

**Configure the VLT domain.**

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.1
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(config-vlt-1)# exit
```

**Configure UFD with uplink VLT ports and downlink network ports.**

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

## 9. Configure overlay IP routing

**Create the tenant VRF.**

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

**Configure the anycast L3 gateway MAC address for all VTEPs.**

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

**Configure routing with an anycast gateway IP address for each virtual network.**

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
```

```

OS10(config-if-vn-10000)# ip address 10.1.0.231/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit
OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.231/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit

```

## VTEP 2 Leaf Switch

### 1. Configure the underlay OSPF protocol.

Do not configure the same router ID on other VTEP switches.

```

OS10(config)# router ospf 1
OS10(config-router-ospf-1)# router-id 172.17.0.1
OS10(config-router-ospf-1)# exit

```

### 2. Configure a Loopback interface.

The source-interface IP address must be same as the source-interface IP address on the VLT peer.

```

OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# ip ospf 1 area 0.0.0.0
OS10(conf-if-lo-0)# exit

```

### 3. Configure the Loopback interface as the VXLAN source tunnel interface.

```

OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit

```

### 4. Configure VXLAN virtual networks with a static VTEP.

```

OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.2.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.2.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit

```

### 5. Assign a switch-scoped VLAN to a virtual network.

```

OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
OS10(config)# interface vlan200
OS10(config-if-vl-100)# virtual-network 20000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit

```

### 6. Configure access ports as VLAN members.

```

OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode access

```

```

OS10(config-if-po-10)# switchport access vlan 200
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode access
OS10(config-if-po-20)# switchport access vlan 200
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit

```

## 7. Configure upstream network-facing ports.

```

OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.17.1.0/31
OS10(config-if-eth1/1/1)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.17.2.0/31
OS10(config-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/2)# exit

```

## 8. Configure VLT

### Configure a dedicated L3 underlay path to reach the VLT Peer in case of network failure.

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.2/30
OS10(config-if-vl-4000)# ip ospf 1 area 0.0.0.0
OS10(config-if-vl-4000)# exit

```

### Configure a VLT port channel.

```

OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt port-channel 10
OS10(config-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt port-channel 20
OS10(config-if-po-20)# exit

```

### Configure VLTi member links.

```

OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit

```

### Configure a VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.2
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(conf-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

## 9. Configure overlay IP routing

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

### Configure an anycast L3 gateway MAC address for all VTEPs.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing with anycast gateway IP address for each virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.232/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit
OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.232/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 3 Leaf Switch

### 1. Configure the underlay OSPF protocol.

Do not configure the same IP address for the router ID and the source loopback interface in Step 2.

```
OS10(config)# router ospf 1
OS10(config-router-ospf-1)# router-id 172.18.0.1
OS10(config-router-ospf-1)# exit
```

### 2. Configure a Loopback interface.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.2.1/32
OS10(conf-if-lo-0)# ip ospf 1 area 0.0.0.0
OS10(conf-if-lo-0)# exit
```

### 3. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

#### 4. Configure VXLAN virtual networks with a static VTEP.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.1.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.1.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

#### 5. Configure a reserved VLAN ID for untagged member interfaces.

```
OS10(config)# virtual-network untagged-vlan 1000
```

#### 6. Configure access ports.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
OS10(conf-if-po-10)# no switchport access vlan
OS10(conf-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# no switchport
OS10(conf-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# no shutdown
OS10(conf-if-po-20)# switchport mode trunk
OS10(conf-if-po-20)# no switchport access vlan
OS10(conf-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(conf-if-eth1/1/6)# no shutdown
OS10(conf-if-eth1/1/6)# channel-group 20 mode active
OS10(conf-if-eth1/1/6)# no switchport
OS10(conf-if-eth1/1/6)# exit
```

#### 7. Add access ports to the VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn-20000)# exit
```

**NOTE:** This step shows how to add access ports using port-scoped VLAN-to-VNI mapping. You can also add access ports using a switch-scoped VLAN-to-VNI mapping. However, you cannot use both methods at the same time; you must use either a port-scoped or switch-scoped VLAN-to-VNI mapping.

#### 8. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.18.1.0/31
OS10(conf-if-eth1/1/1)# ip ospf 1 area 0.0.0.0
OS10(conf-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
```

```
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.18.2.0/31
OS10(config-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/2)# exit
```

## 9. Configure VLT

### Configure VLTi VLAN for the VXLAN virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vlti-vlan 100
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit
```

### Configure a dedicated L3 underlay path to reach the VLT Peer in case of network failure.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.9/30
OS10(config-if-vl-4000)# ip ospf 1 area 0.0.0.0
OS10(config-if-vl-4000)# exit
```

### Configure a VLT port channel.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt port-channel 10
OS10(config-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt port-channel 20
OS10(config-if-po-20)# exit
```

### Configure VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

### Configure a VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.3
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:dd:cc:ff:ee
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

## 10. Configure overlay IP routing

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast L3 gateway.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing with an anycast gateway IP address for each virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.233/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit
OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.233/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 4 Leaf Switch

### 1. Configure the underlay OSPF protocol.

Do not configure the same IP address for the router ID and the source loopback interface in Step 2..

```
OS10(config)# router ospf 1
OS10(config-router-ospf-1)# router-id 172.19.0.1
OS10(config-router-ospf-1)# exit
```

### 2. Configure a Loopback interface.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.2.1/32
OS10(conf-if-lo-0)# ip ospf 1 area 0.0.0.0
OS10(conf-if-lo-0)# exit
```

### 3. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 4. Configure VXLAN virtual networks with a static VTEP.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.1.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# remote-vtep 192.168.1.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 5. Configure a reserved VLAN ID for untagged member interfaces.

```
OS10(config)# virtual-network untagged-vlan 1000
```

### 6. Configure access ports.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
```

```

OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit

```

## 7. Add access ports to the VXLAN virtual network.

```

OS10(config)# virtual-network 10000
OS10(config-vn-10000)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn-20000)# exit

```

## 8. Configure upstream network-facing ports.

```

OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.19.1.0/31
OS10(config-if-eth1/1/1)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.19.2.0/31
OS10(config-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/2)# exit

```

## 9. Configure VLT

### Configure VLTi VLAN for the VXLAN virtual network.

```

OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vlti-vlan 200
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 100
OS10(config-vn-20000)# exit

```

### Configure a dedicated L3 underlay path to reach the VLT Peer in case of network failure.

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.10/30
OS10(config-if-vl-4000)# ip ospf 1 area 0.0.0.0
OS10(config-if-vl-4000)# exit

```

### Configure a VLT port channel.

```

OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt port-channel 10

```

```
OS10(config-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt port-channel 20
OS10(config-if-po-20)# exit
```

### Configure VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

### Configure a VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.4
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:dd:cc:ff:ee
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

## 10. Configure overlay IP routing.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast L3 gateway for all VTEPs in all virtual networks.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing with an anycast gateway IP address for each virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.234/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit
OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.234/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## Spine Switch 1

### 1. Configure downstream ports on underlay links to leaf switches.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
```

```

OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# ip address 172.16.1.1/31
OS10(config-if-eth1/1/1)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# ip address 172.17.1.1/31
OS10(config-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/2)# exit

OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# ip address 172.18.1.1/31
OS10(config-if-eth1/1/3)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# ip address 172.19.1.1/31
OS10(config-if-eth1/1/4)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/4)# exit

```

## 2. Configure the underlay OSPF protocol.

```

OS10(config)# router ospf 1
OS10(config-router-ospf-1)# router-id 172.200.0.1
OS10(config-router-ospf-1)# exit

```

## Spine Switch 2

### 1. Configure downstream ports on underlay links to leaf switches.

```

OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# ip address 172.16.2.1/31
OS10(config-if-eth1/1/1)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# ip address 172.17.2.1/31
OS10(config-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/2)# exit

OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# ip address 172.18.2.1/31
OS10(config-if-eth1/1/3)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# ip address 172.19.2.1/31
OS10(config-if-eth1/1/4)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/4)# exit

```

### 2. Configure the underlay OSPF protocol.

```

OS10(config)# router ospf 1
OS10(config-router-ospf-1)# router-id 172.201.0.1
OS10(config-router-ospf-1)# exit

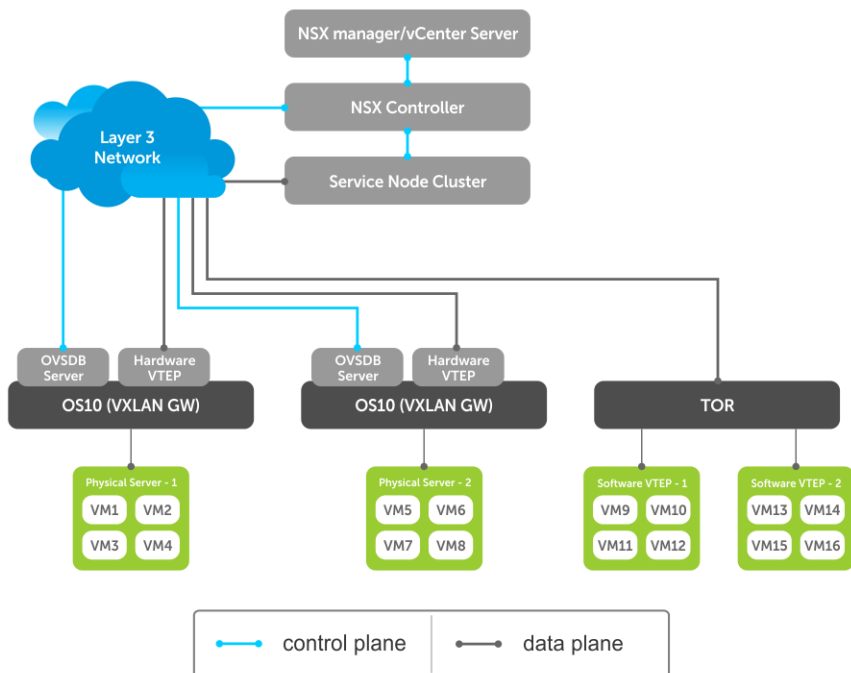
```

# Controller-provisioned VXLAN

OS10 supports VXLAN provisioning using an Open vSwitch Database (OVSDB) controller. The only supported OVSDB controller is the VMware NSX controller. In a controller-provisioned VXLAN, the controller manages VXLAN-related configurations and other control-plane operations, such as MAC address propagation.

**NOTE:** Controller-provisioned VXLAN is not supported on S5148F-ON and S3048-ON switches. Also, controller-provisioned VXLAN is not supported on VTEPs configured as peers in a VLT domain. Only VTEPs in stand-alone mode are supported.

## Controller-provisioned VXLAN



The NSX controller communicates with an OS10 VTEP using the OVSDB management protocol over a Secure Sockets Layer (SSL) connection. To establish communication between the controller and VTEP, generate the SSL certificate at a VTEP and copy the certificate to the NSX controller. After SSL authentication, a secure connection over SSL is established between the controller and the VTEP. The VTEP then receives and processes the configuration data from the controller.

### Controller-provisioned VXLAN: Manual configuration

Manually configure the underlay network using the OS10 CLI:

- Configure the L3 protocol used for underlay routing. Underlay reachability to VTEP peers is learned using the configured routing protocol.
- Configure the loopback interface in the default VRF that is used as the VTEP source IP address for controller-based provisioning.
- Assign the VTEP interfaces to that you want the controller to manage.

### Controller-provisioned VXLAN: Automatic provisioning

The controller automatically provisions:

- L2 overlay network
- VXLAN virtual networks, including remote VTEP source addresses
- Local access ports in a virtual network

An OS10 VTEP sends the addition or deletion of server MAC addresses at the VXLAN access port to the NSX controller using the OVSDB protocol. The controller then propagates the information to VTEP peers. The VTEPs program their forwarding tables accordingly.

## Configure controller-provisioned VXLAN

To configure the NSX controller, follow these steps on each OS10 VTEP:

1. Configure the source interface used for controller-based VXLAN provisioning. Assign an IPv4 address to a loopback interface. Assign the loopback interface to an NVE instance. The loopback interface must belong to the default VRF. For detailed information, see the [Configure source IP address on VTEP](#).
2. Configure NSX controller reachability.
3. Assign local access interfaces to be managed by the controller. The VLAN IDs of member access interfaces created using the OS10 CLI must be different from the VLAN IDs of port-scoped VLANs created by the NSX controller for virtual networks.
4. (Optional) Enable BFD in the NSX and the VTEP. OS10 complies with RFC5880 for Bidirectional Forwarding Detection.

#### Configuration notes

- NSX controller-provisioned VXLAN is not supported if an OS10 switch operates in OpenFlow-only mode.
- Only one mode of VxLAN provisioning is supported at a time: NSX controller-based, static VXLAN, or BGP EVPN.
- An OS10 switch does not send VXLAN access port statistics to the NSX controller.
- Controller-provisioned VXLAN is not supported on VTEPs configured as peers in a VLT domain. Only VTEPs in standalone mode are supported.

## Specify the controller reachability information

In OS10 VTEP, the controller configuration command initializes a connection to an OVSDDB-based controller.

OS10 supports only one controller connection at a time.

 **NOTE:** Currently, the only supported OVSDDB-based controller is NSX.

To configure an OVSDDB controller on the OS10 VTEP:

1. Enable VXLAN in CONFIGURATION mode.  

```
OS10(config)# nve
```
2. Changes the mode to CONFIGURATION-NVE-OVSDDB from where you can configure the controller parameters.  

```
OS10(config-nve)# controller ovsdb
```
3. Specify the IP address, OVSDDB controller port, and SSL as a secure connection protocol between the OS10 VTEP and the controller in CONFIGURATION-NVE-OVSDDB mode.  

```
OS10(config-nve-ovsdb)# ip ip-address port port-number ssl
```

The range of *port-number* is from 0 to 65535. Configure the *port-number* as 6640 and the connection type as SSL.
4. (Optional) Specify a time interval, in milliseconds (ms). This is the duration the switch waits between the connection attempts to the controller.  

```
OS10(config-nve-ovsdb)# max-backoff interval
```

The range is from 1000 to 180,000 ms. The default is 8000 ms.

```
OS10# configure terminal
OS10(config)# nve
OS10(config-nve)# controller ovsdb
OS10(config-nve-ovsdb)# ip 10.11.66.110 port 6640 ssl
```

## Assign interfaces to be managed by the controller

In a VTEP, explicitly assign interfaces for an OVSDDB controller to manage.

Before you assign the interface, consider the following:

- The interface must be in Switchport Trunk mode.
- The interface must not be a member of any VLAN
- The interface must not be a member of a port-channel

When the above conditions are not met when assigning the interfaces to be managed by the controller, the system returns error messages.

When the interface is assigned, you cannot:

- remove the interface from Switchport Trunk mode
- add the interface as a member of any VLAN

- remove the interface from the controller configuration if the interface has active port-scoped VLAN (Port,VLAN) pairs configured by the controller

To assign an interface to be managed by the OVSDB controller:

1. Configure an interface from CONFIGURATION mode.  
OS10(config)# interface ethernet 1/1/1
2. Configure L2 trunking in INTERFACE mode.  
OS10(config-if-eth1/1/1)# switchport mode trunk
3. Configure the access VLAN assigned to a L2 trunk port in the INTERFACE mode.  
OS10(config-if-eth1/1/1)# no switchport access vlan
4. Assign the interface to the controller.  
OS10(config-if-eth1/1/1)# nve-controller

To view the controller information and the ports the controller manages, use the `show nve controller` command.

```
OS10# show nve controller

Management IP           : 10.16.140.29/16
Gateway IP              : 55.55.5.5
Max Backoff             : 1000
Configured Controller   : 10.16.140.172:6640 ssl (connected)

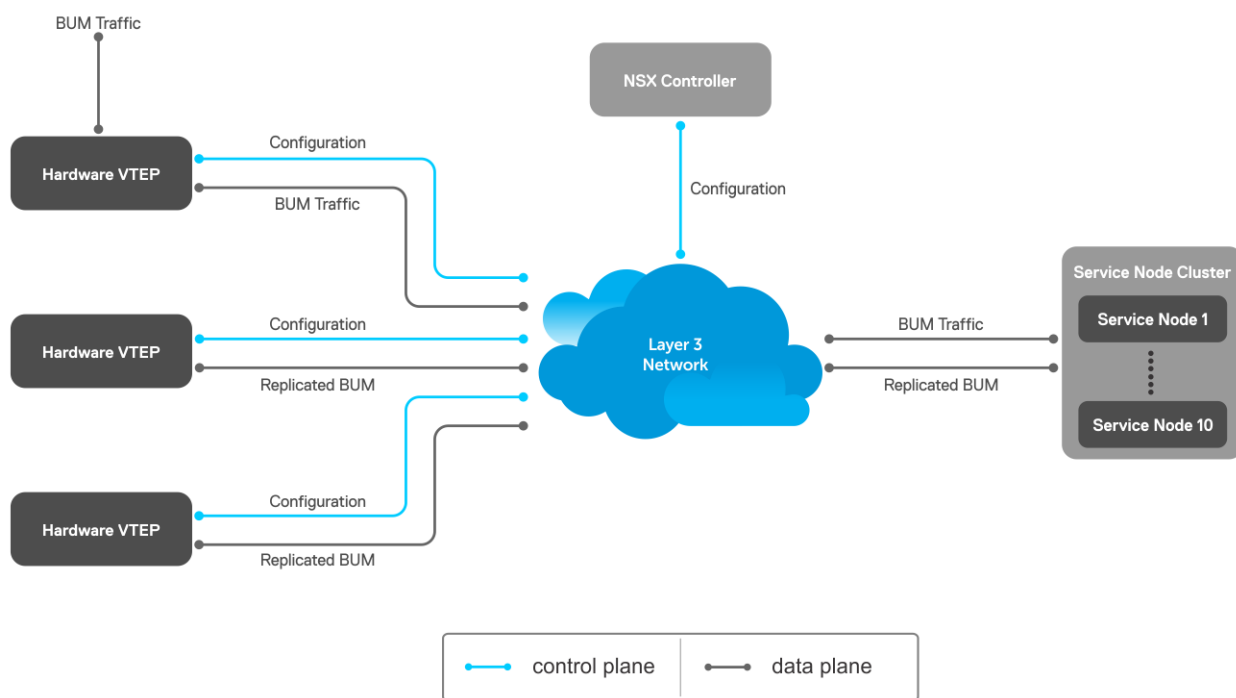
Controller Cluster
IP          Port      Protocol  Connected  State      Max-Backoff
10.16.140.173 6640    ssl      true       ACTIVE     1000
10.16.140.171 6640    ssl      false      BACKOFF    1000
10.16.140.172 6640    ssl      true       ACTIVE     1000

NVE Controller Ports
ethernet1/1/1:1
ethernet1/1/15
```

## Service Nodes

In an NSX-provisioned VXLAN environment, service nodes replicate L2 broadcast, unknown-unicast, and multicast (BUM) traffic that enter an OS10 VTEP to all other VTEPs. For the service node replication of BUM traffic to work, you need IP connectivity between the service nodes and the VTEP, so that the BUM traffic from a VTEP reaches the other remote VTEPs via a VXLAN overlay through the service nodes. The NSX controller manages a cluster of service nodes and sends the IP addresses of the nodes to the VTEP through OVSDB protocol. The service node cluster provides redundancy, and also facilitates load balancing of BUM traffic across service nodes.

The following shows BUM traffic replication in the controller-provisioned VXLAN environment:



Since VTEP relies on service nodes to replicate BUM traffic, we need a mechanism to monitor the connectivity between the VTEP and the service nodes. BFD can be used to monitor the connectivity between the VTEP and service nodes, and detects failures. The NSX controller provides parameters, such as the minimum TX and RX interval, and the multiplier, to initiate the BFD session between the VTEP and the service nodes. To establish a BFD session, enable the BFD on the controller and the VTEP. To enable BFD in the VTEP, use `bfd enable` command.

**NOTE:** In controller-provisioned VXLAN, the VTEP establishes a BFD session with the service nodes using the controller-provided parameters instead of the parameters configured at the VTEP.

If BFD is not enabled in the VTEP, the VTEP uses IP reachability information to monitor connectivity to the service node.

To view established sessions, use the `show bfd neighbors` command.

```
OS10# show bfd neighbors
* - Active session role
-----
-
  LocalAddr      RemoteAddr      Interface      State RxInt TxInt Mult VRF      Clients
-----
-
* 55.55.5.5      2.2.2.2         virtual-network0 up    1000 1000 3    default vxlan
* 55.55.5.5      2.2.2.3         virtual-network0 up    1000 1000 3    default vxlan
```

## View replicators

To view the state of the replicators, use the `show nve replicators` command.

- Show output with details about the replicators received from the controller.

```
OS10# show nve replicators
Codes: * - Active Replicator

BFD Status:Enabled
Replicators      State
-----
2.2.2.3          Up
2.2.2.2          Up
```

- Show output with details about the replicators available for the VNID.

```
OS10# show nve replicators vnid 10009
Codes: * - Active Replicator

BFDD Status:Enabled
Replicators      State
-----
2.2.2.3          Up
2.2.2.2*         Up
```

\*— indicates the replicator to which the VTEP sends the BUM traffic for the specific VNID.

## Configure and control VXLAN from VMware vCenter

You can configure and control VXLAN from the VMware vCenter GUI. Complete the following steps:

1. On an OS10 switch, generate an SSL certificate in CONFIGURATION mode.

```
OS10# nve controller ssl-key-generate
```

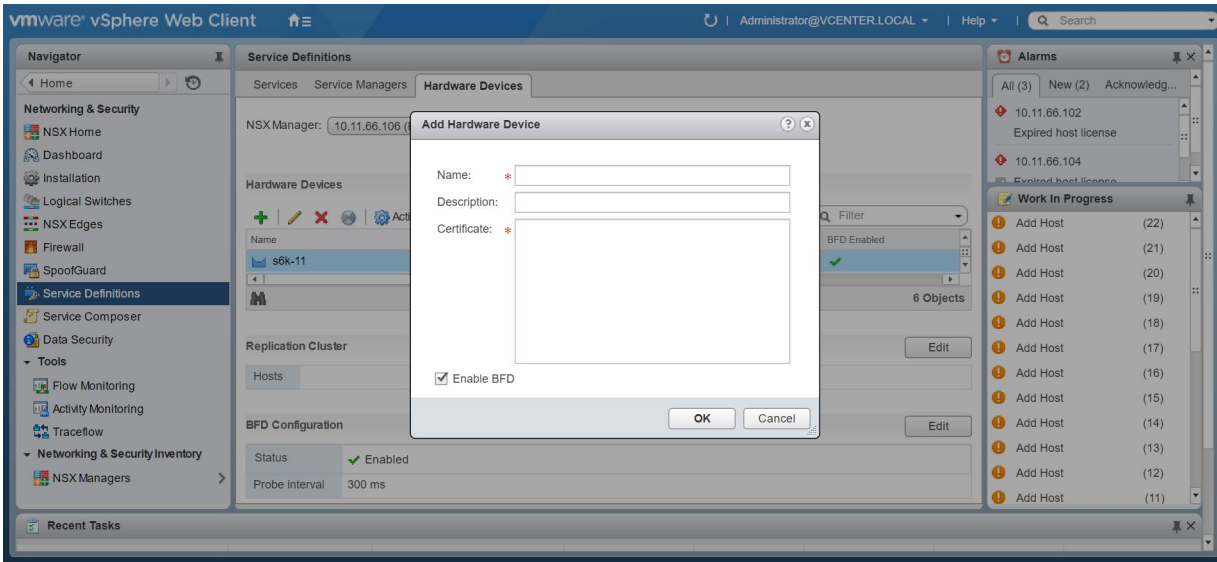
Verify or view the certificate using the `show nve controller ssl-certificate` command.

```
OS10# show nve controller ssl-certificate
-----BEGIN CERTIFICATE-----
MIIDgDCCAmgCAQMwDQYJKoZIhvcNAQENBQAwgYExCzAJBgNVBAYTAlVTMQswCQYD
VQQIDAJDQTEVMBMGA1UECgwMT3BlbiB2U3dpdGNoMREwDwYDVQQLEDAhZd210Y2hj
YTE7MDkGA1UEAwwyT1ZTIHN3aXRjaGNoIENBIENlcnRpZmljYXRlICgyMDE4IFNl
cCAyMyAwMzo0NzoyMCKwHhcNMjgwOTI0MTYzMDUyWhcNMjgwOTI0MTYzMDUyWjCB
iTELMakGA1UEBhMCVVMxMzA2ZjBGNVBAgMAkNBMRUwEwYDVQQKDAxPcGVuIHZTd210
Y2gXZAdBgNVBASMFk9wZW4gd1N3aXRjaCBjZXJ0aWZpZjYwZjYwZjYwZjYwZjYw
bGwgaWQ6MGVlZmUwYWMtNGJjOC00MmVmLTkzOTEtN2RlMmMwY2JmMTJjMIIBIjAN
BgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsMlD4c4fWwy+5t6VScjizlkFsNzE
BOK5PJyI3B6ReRK/J14Fdxio1YmzG0YobjxiwjpUYEsqPL3Nvh0f10KMqwqJVBdf
6sXWHUVw+9A7cIfRh0aRI+HIYyUC4YD48GlnVnaCqhxYaA0tcMzJm4r2k7AjwJUL
0pDXiqS3uJwGmfxlhvmFio8EeHM/Z79DkBRD6FUMwacAnb3yCIKZH50AWq7qRmmG
NZOgYUT+8oaj5tO/hEQfDYuv32E5z4d3FhiBJMFT86T4YvpJYyJkiKmaQWInkthL
V3VxEMXI5vJQclMhwYbKfPB4hh3+qdS5o+uVco76CVrcWi7r03XmsBkbnQIDAQAB
MA0GCSqGSIb3DQEBDQUAA4IBAQAUFVD20GcHD8zdpYf0YaP4b6TuonUzF0jwoV+
Qr9b4kOjEBGuoPdevX3AeV/dvAa2Q6o1iOBM5z74NgHizhr067pFP841Nv7DAVb7
cPHHSSTTSeeJjIVMh0kv0KkVefsYuI4r1jqJxu0GZgBinqehXxVKlceouLvwbbh1
MFYXN3lcE2AXR746q1VIC6stNkxf3nrlOpSDz3P4VOnbAnIrY+SvUVmAT0tdrowH
99y2AzoAxUH0dWsh8EjCFch7VilmCVVhyghXdfyl6lv/F6vMRwj343BpBW3QsGj
68ROX0ILrtOz/2q5oUb/rpJd15KFFN3itT/xYBfZ1ZdLYd5F
-----END CERTIFICATE-----
```

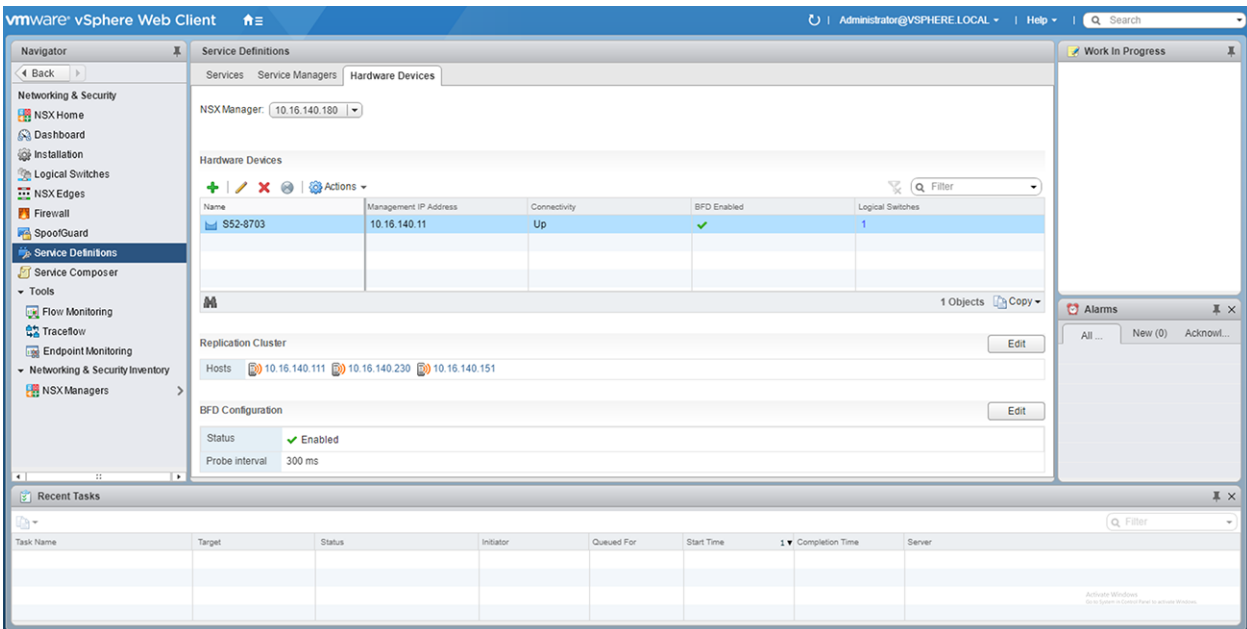
2. Create a VXLAN gateway in VMware vCenter console.

This following steps configure the VXLAN gateway:

- a. Open a browser window, enter the vCenter IP address, and log in to VMware vCenter.
- b. Click **Service Definitions** from the left navigation pane.
- c. Click the **Hardware Devices** tab.
- d. Click the green **+** icon under **Hardware Devices** to add a device. The **Add Hardware Device** dialog window opens.
- e. Enter a name for the device in the **Name** box and copy the certificate generated in the OS10 switch and paste it in the **Certificate** box and click **OK**.



If successfully establishing connectivity between the VTEP and the NSX controller, the console displays the current connection status between the controller and the management IP address of the VTEP.

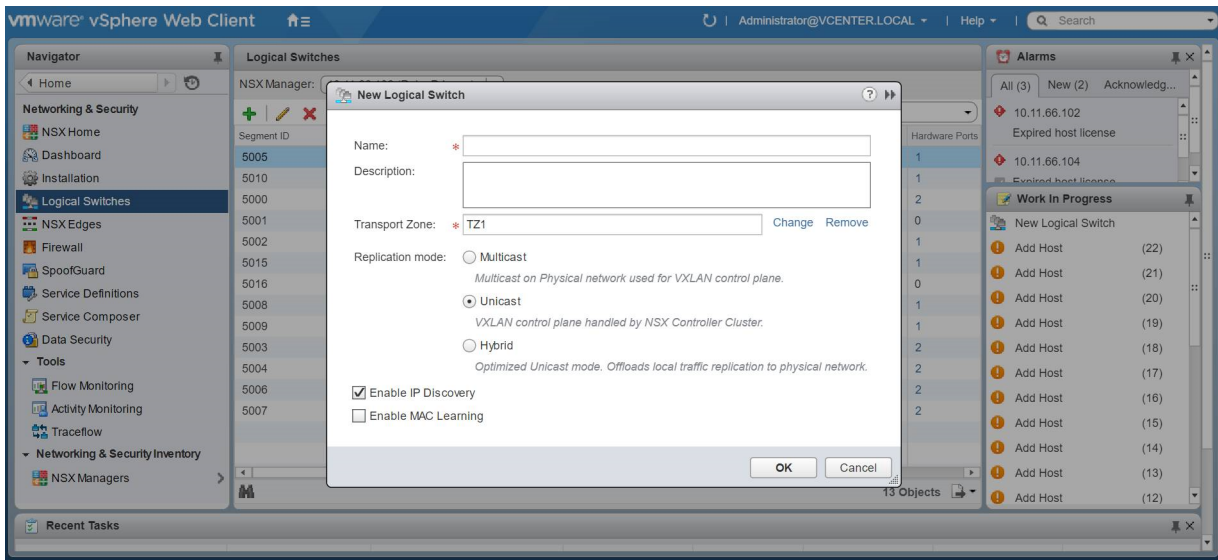


### 3. Create a logical switch.

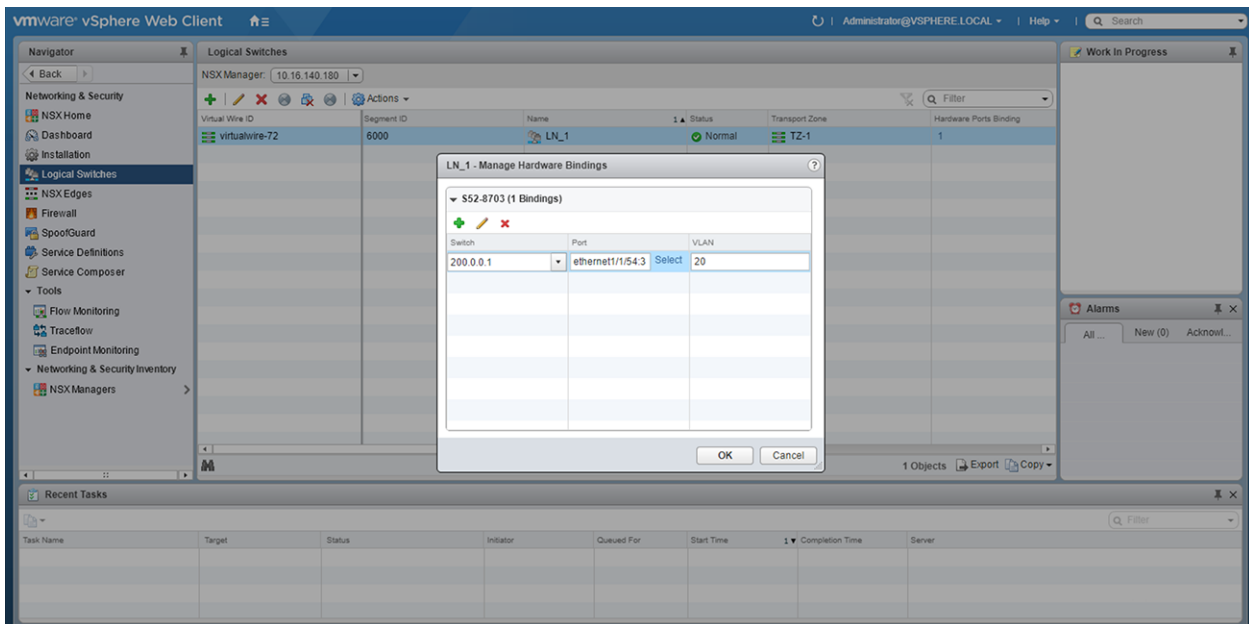
You can create a logical network that acts as the forwarding domain for virtualized and nonvirtualized server workloads on the physical and virtual infrastructure.

The following steps configure the logical switch for NSX controller management.

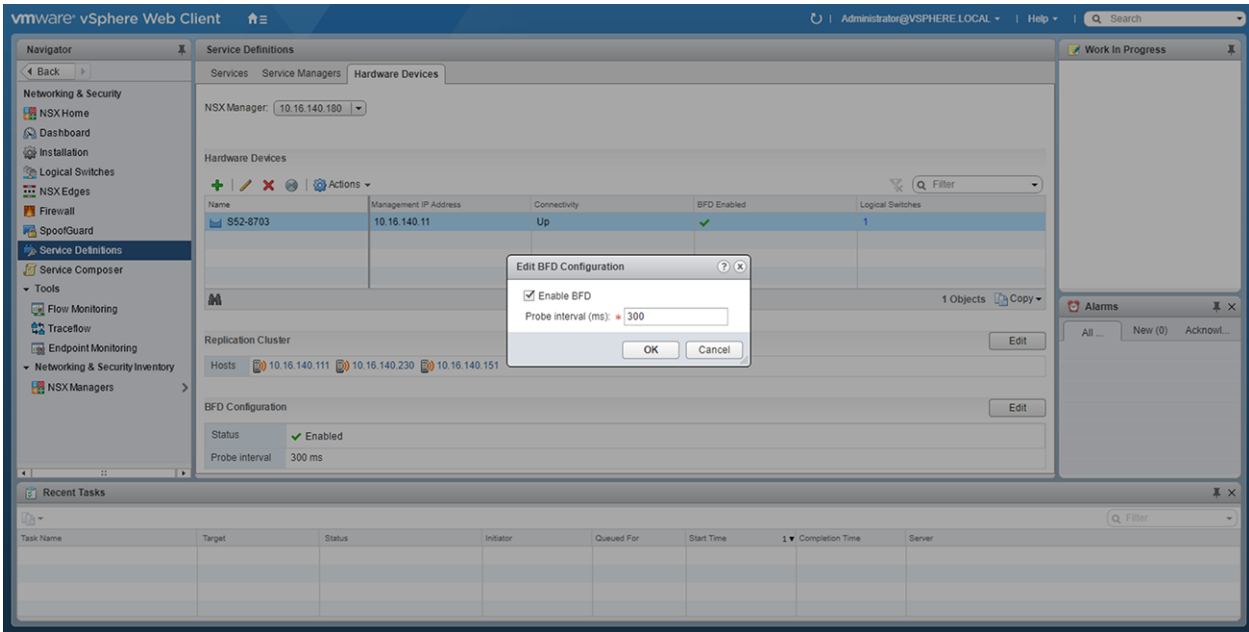
- a. Click **Logical Switches** from the left navigation pane.
- b. Click the green **+** icon under **Logical Switches**. The **New Logical Switch** dialog window opens.
- c. Enter a name and select **Unicast** as the replicate mode and click **OK**



4. Create a logical switch port that provides a logical connection point for a VM interface (VIF) and a L2 gateway connection to an external network.



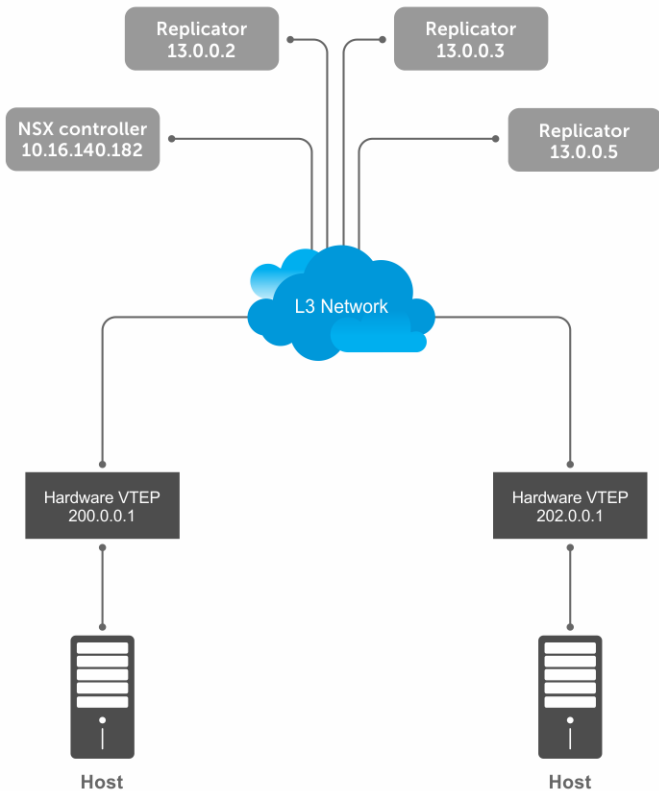
5. (Optional) Enable or disable BFD globally.  
The following steps enable or disable BFD configuration in the controller.
  - a. Click **Service Definitions** from the left navigation pane.
  - b. Click the **Hardware Devices** tab.
  - c. Click the **Edit** button in the **BFD Configuration**.
  - d. Check or clear the **Enable BFD** check box and provide the **Probe interval**, in milliseconds, if required.



After you configure a VMware NSX controller on a server VM, connect to the controller from the VXLAN gateway switch. For more information about the NSX controller configuration in the VTEP, see [Configure a connection to an OVSDB controller](#). For more information about NSX controller configuration, see the *NSX User Guide* from VMware.

## Example: VXLAN with a controller configuration

This example shows a simple NSX controller and an hardware OS10 VTEP deployed in VXLAN environment.



To configure an NSX controller-provisioned VXLAN:

- Configure the controller and the interfaces to be managed by the controller, in the OS10 VTEPs

- Configure the NSX controller in VMware vCenter. For more information about configuring the NSX controller using the GUI, see the [Configure and control VXLAN from the VMware vCenter](#).

You must configure an OS10 VTEP with the controller configuration so that the VTEP can communicate with the NSX controller. The NSX controller handles configurations and control plane operations in the VXLAN environment.

## VTEP 1

1. Configure the OSPF protocol in the underlay.

```
OS10# configure terminal
OS10(config)# router ospf 1
OS10(config)# exit
OS10(config)# interface ethernet 1/1/55:1
OS10(config-if-eth1/1/55:1)# no switchport
OS10(config-if-eth1/1/55:1)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/55:1)# exit
```

2. Configure a Loopback interface.

```
OS10(config)# interface loopback 1
OS10(config-if-lo-1)# no shutdown
OS10(config-if-lo-1)# ip address 200.0.0.1/32
OS10(config-if-lo-1)# exit
```

3. Create an NVE instance and configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback 1
```

4. Specify the NSX controller reachability information.

```
OS10(config-nve)# controller ovssdb
OS10(config-nve-ovssdb)# ip 10.16.140.182 port 6640 ssl
OS10(config-nve-ovssdb)# max-backoff 10000
OS10(config-nve-ovssdb)# exit
```

5. Assign interfaces to be managed by the controller.

```
OS10(config)# interface ethernet 1/1/54:3
OS10(config-if-eth1/1/54:3)# switchport mode trunk
OS10(config-if-eth1/1/54:3)# no switchport access vlan
OS10(config-if-eth1/1/54:3)# nve-controller
```

6. (Optional) Enable BFD.

```
OS10(config)# bfd enable
```

## VTEP 2

1. Configure the OSPF protocol in the underlay.

```
OS10# configure terminal
OS10(config)# router ospf 1
OS10(config)# exit
OS10(config)# interface ethernet 1/1/23:1
OS10(config-if-eth1/1/23:1)# no switchport
OS10(config-if-eth1/1/23:1)# ip ospf 1 area 0.0.0.0
OS10(config-if-eth1/1/23:1)# exit
```

2. Configure a Loopback interface.

```
OS10(config)# interface loopback 1
OS10(config-if-lo-1)# no shutdown
OS10(config-if-lo-1)# ip address 202.0.0.1/32
OS10(config-if-lo-1)# exit
```

3. Create an NVE instance and configure a Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback 1
```

4. Specify the NSX controller reachability information.

```
OS10(config-nve)# controller ovssdb
OS10(config-nve-ovssdb)# ip 10.16.140.182 port 6640 ssl
OS10(config-nve-ovssdb)# max-backoff 10000
OS10(config-nve-ovssdb)# exit
```

5. Assign interfaces to be managed by the controller.

```
OS10(config)# interface ethernet 1/1/25:3
OS10(config-if-eth1/1/25:3)# switchport mode trunk
OS10(config-if-eth1/1/25:3)# no switchport access vlan
OS10(config-if-eth1/1/25:3)# nve-controller
```

6. (Optional) Enable BFD.

```
OS10(config)# bfd enable
```

## Verify the controller configuration

### VTEP 1

To view controller-based information on the VTEP 1, use the `show nve controller` command.

```
OS10# show nve controller

Management IP           : 10.16.140.11/16
Gateway IP              : 200.0.0.1
Max Backoff             : 10000
Configured Controller   : 10.16.140.181:6640 ssl (connected)

Controller Cluster
IP           Port      Protocol  Connected  State      Max-Backoff
10.16.140.182 6640    ssl      true       ACTIVE    10000
10.16.140.183 6640    ssl      true       ACTIVE    10000
10.16.140.181 6640    ssl      true       ACTIVE    10000

NVE Controller Ports
ethernet1/1/54:3
```

To display the VNID, port members, source interface, and remote VTEPs of the VXLAN, use the `show virtual-network` command.

```
OS10# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
Virtual Network: 0
Members:

Virtual Network: 6000
Members:
  VLAN 20: ethernet1/1/54:3
VxLAN Virtual Network Identifier: 6000
Source Interface: loopback1(200.0.0.1)
Remote-VTEPs (flood-list): 13.0.0.5(CP)
```

To view all the replicators and their status in the VXLAN, use the `show nve replicators` command.

```
OS10# show nve replicators
Codes: * - Active Replicator

BFD Status:Disabled
Replicators      State
```

```
-----
13.0.0.5      Up
13.0.0.3      Up
13.0.0.2      Up
```

To view the remote VTEP status, use the `show nve remote-vtep` command.

```
OS10# show nve remote-vtep
IP Address: 13.0.0.2, State: up, Encap: VxLAN
VNI list: ,6000
IP Address: 13.0.0.3, State: up, Encap: VxLAN
VNI list: ,6000
IP Address: 13.0.0.5, State: up, Encap: VxLAN
VNI list: ,6000
IP Address: 202.0.0.1, State: up, Encap: Vxlan
VNI list: 6000
```

## VTEP 2

```
OS10# show nve controller

Management IP      : 10.16.140.13/16
Gateway IP        : 202.0.0.1
Max Backoff       : 10000
Configured Controller : 10.16.140.181:6640 ssl (connected)

Controller Cluster
IP                Port      Protocol  Connected  State      Max-Backoff
10.16.140.182    6640    ssl       true       ACTIVE    10000
10.16.140.183    6640    ssl       true       ACTIVE    10000
10.16.140.181    6640    ssl       true       ACTIVE    10000

NVE Controller Ports
ethernet1/1/25:3
```

To display the VNID, port members, source interface, and remote VTEPs of the VXLAN, use the `show virtual-network` command.

```
OS10# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
Virtual Network: 0
Members:

Virtual Network: 6000
Members:
  VLAN 20: ethernet1/1/25:3
VxLAN Virtual Network Identifier: 6000
Source Interface: loopback1(202.0.0.1)
Remote-VTEPs (flood-list): 13.0.0.5(CP)
```

To view all the replicators and their status in the VXLAN, use the `show nve replicators` command.

```
OS10# show nve replicators
Codes: * - Active Replicator

BFD Status:Disabled
Replicators      State
-----
13.0.0.5        Up
13.0.0.3        Up
13.0.0.2        Up
```

To view the remote VTEP status, use the `show nve remote-vtep` command.

```
OS10# show nve remote-vtep
IP Address: 13.0.0.2, State: up, Encap: VxLAN
VNI list: ,6000
IP Address: 13.0.0.3, State: up, Encap: VxLAN
```

```
VNI list: ,6000
IP Address: 13.0.0.5, State: up, Encap: VxLAN
VNI list: ,6000
IP Address: 200.0.0.1, State: up, Encap: Vxlan
VNI list: 6000
```

## VXLAN Controller commands

### controller ovssdb

Changes the mode to CONFIGURATION-NVE-OVSDB from where you can configure the controller parameters.

**Syntax** `controller ovssdb`

**Parameters** None

**Default** None

**Command mode** CONFIGURATION-NVE

**Usage information** The controller configuration initiates the OVSDB service on the OS10 switch.

The `no` version of this command stops the OVSDB service. The `no` version command fails if any ports are configured as controller-managed ports or IP address configuration.

**NOTE:** Before removing the controller configuration from the device, you must delete all controller-managed ports and IP address configuration.

#### Example

```
OS10(config)# nve
OS10(config-nve)# controller ovssdb
```

**Supported releases** 10.4.3.0 or later

### ip port ssl

Configures the OVSDB controller reachability information such as IP address, port number, and the connection type of session, in the switch.

**Syntax** `ip ip-address port port-number ssl`

**Parameters**

- `ip-address` — Specify the IP address of the OVSDB controller to connect with.
- `port-number` — Specify the port number through which the connection to the OVSDB controller is made.

**Default** For an OVSDB-based controller, configure the following:

- Port number as 6640
- Connection type as SSL

**Command mode** CONFIGURATION-NVE-OVSDB

**Usage information** Currently, the only supported OVSDB controller is the NSX controller. `no` version of this command removes the connection to the OVSDB controller.

#### Example

```
OS10(config)# nve
OS10(config-nve)# controller ovssdb
OS10(config-nve-ovssdb)# ip 10.11.66.110 port 6640 ssl
```

**Supported releases** 10.4.3.0 or later

## max-backoff

Configures a time interval, in milliseconds (ms). This is the duration the switch waits between the connection attempts to the controller.

**Syntax** `max-backoff interval`

**Parameters** `interval`—Enter the amount of time, in ms. This is the duration the switch waits between the connection attempts to the controller, from 1000 to 180000 ms.

**Default** 8000 ms

**Command Mode** CONFIGURATION-NVE-OVSDB

**Usage Information** The `no` version of this command replaces the default maximum wait time configuration in the switch.

### Example

```
OS10(config)# nve
OS10(config-nve)# controller ovsdb
OS10(config-nve-ovsdb)# max-backoff 40000
```

**Supported Releases** 10.4.3.0 or later

## nve-controller

Assigns the interfaces to be managed by the controller.

**Syntax** `nve-controller`

**Parameters** None

**Default** None

**Command mode** INTERFACE

**Usage information** The interface must be in Switchport Trunk mode when adding the interface to the controller. If the interface is not in the Switchport Trunk mode, the system displays the following error message:

```
% Error: Interface ethernet1/1/1, must be in switchport trunk for controller mode.
```

**NOTE:** If the interface has active port-scoped VLAN (Port,VLAN) pairs configured by the controller, you cannot remove an interface from the controller.

The `no` version of this command removes the interface from the controller and removes any VXLAN binding associated with the interface.

### Example

```
OS10(config)# interface ethernet 1/1/1
OS10(config-if-eth1/1/1)# nve-controller
```

**Supported releases** 10.4.3.0 or later

## nve controller ssl-key-generate

Generates the SSL certificate for the OVSDB server to setup the SSL connection with the controller.

**Syntax** `nve controller ssl-key-generate`

**Parameters** None

**Default** None

**Command mode** EXEC



```
bGwgaWQ6MGV1ZmUwYWMtNGJjOC00MmVmLTkzOTEtN2RlMmMwY2JmMTJjMIIBIjAN
BgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsM1D4c4fWwy+5t6VScjiZlkFsNzE
BOK5PJyI3B6ReRK/J14Fdxio1YmzG0YObjxiwjpUYEsqPL3Nvh0f10KMqwqJVbdf
6sXWHUVw+9A7cIfRh0aRI+HIYyUC4YD48GlnVnaCqhxYaa0tcMzJm4r2k7AjjwJU1
0pDXiqS3uJwGmfxlhvmFio8EeHM/Z79DkBRD6FUMwacAnb3yCIKZH50AWq7qRmmG
NZOgYUT+8oaj5tO/hEQfDYuv32E5z4d3FhiBJMFT86T4YvpJYyJkiKmaQWInkthL
V3VxEMXI5vJQc1MhwYbKfPB4hh3+qdS5o+uVco76CVrcWi7r03XmsBkbnQIDAQAB
MA0GCSqGSIsb3DQEEDQUAA4IBAQATuFVD20GcHD8zdpYf0YaP4b6TuonUzF0jwoV+
Qr9b4k0jEBGuoPdevX3AeV/dvAa2Q6o1iOBM5z74NgHizhr067pFP841Nv7DAVb7
cPHHSSTTSeeJjIVMh0kv0KkVefsYuI4r1jqJxu0GZgBinqehXxVKlceouLvwbbh1
MFYXN3lcE2AXR746q1VIc6stNkxf3nr1OpSDz3P4V0nbAnIrY+SvUVMAT0tdrowH
99y2AzoAxUHOdWsh8EjCFch7VilmCVVhyghXdfyl6lv/F6vMRwjc343BpBW3QsGj
68ROX0ILrtOz/2q5oUb/rpJd15KFFN3itT/xYBfZ1ZdLYd5F
-----END CERTIFICATE-----
```

**Supported releases** 10.4.3.0 or later

## show nve replicators

Displays all the replicators and their states.

**Syntax** show nve replicators [vnid vnid]

**Parameters** None

**Default** None

**Command mode** EXEC

**Usage information** When you specify the VNID, the output displays details about the service nodes available for the VNID.

**Example (without VNID)**

```
OS10# show nve replicators
Codes: * - Active Replicator

BFD Status:Enabled
Replicators      State
-----
2.2.2.3          Up
2.2.2.2          Up

OS10# show nve replicators
```

**Example (with VNID)**

```
OS10# show nve replicators vnid 10009
Codes: * - Active Replicator

BFD Status:Enabled
Replicators      State
-----
2.2.2.3          Up
2.2.2.2*         Up
```

\* — indicates service node to which the VTEP sends BUM traffic for the specific VNID.

**Supported releases** 10.4.3.0 or later

## show ovssdb-tables mac-local-ucast

Displays information about local MAC address entries including each MAC address, IP address, local switch name, and VNID.

**Syntax** show ovssdb-tables mac-local-ucast

**Parameters** None

**Default** None

**Command mode** EXEC

**Usage information** This command is available only for netadmin, sysadmin, and secadmin roles.

**Example**

```
OS10# show ovssdb-tables mac-local-ucast
Count : 1356
Ucast_Macs_Local table
MAC          _uuid          ipaddr locator
logical_switch
-----
"00:00:09:00:00:00" 948d2357-9a68-49b2-b5b2-a6a9beaec17a "" bb43d2ec-1e60-4367-9840-648a8cc8acff
f8994210-e29d-4ad4-90fb-557c30f83769
"00:00:09:00:00:01" 4e620093-311a-420e-957f-fbd2bb63f20a "" bb43d2ec-1e60-4367-9840-648a8cc8acff
f8994210-e29d-4ad4-90fb-557c30f83769
"00:00:09:00:00:02" 3846973c-2b29-4c84-af39-dfe7513cdb3d "" bb43d2ec-1e60-4367-9840-648a8cc8acff
f8994210-e29d-4ad4-90fb-557c30f83769
```

**Supported releases** 10.4.3.0 or later

## show ovssdb-tables mac-remote-ucast

Displays information about remote MAC address entries including each MAC address, IP address, local switch name, and VNID.

**Syntax** show ovssdb-tables mac-remote-ucast

**Parameters** None

**Default** None

**Command mode** EXEC

**Usage information** This command is available only for netadmin, sysadmin, and secadmin roles.

**Example**

```
OS10# show ovssdb-tables mac-remote-ucast
Count : 1
Ucast_Macs_Remote table
MAC          _uuid          ipaddr locator
logical_switch
-----
"00:50:56:8a:b4:c8" 61fa240b-e6a3-4d8e-a693-dd2468e6f308 "" 3105e34b-a273-4193-a60f-51d9cee91403
6932fc02-fb12-4a22-9ec2-f0e2b20df476
```

**Supported releases** 10.4.3.0 or later

## show ovssdb-tables manager

Displays information about the list of controllers and the respective controller connection details.

**Syntax** show ovssdb-tables manager

**Parameters** None

**Default** None

**Command mode** EXEC

**Usage information** This command is available only for netadmin, sysadmin, and secadmin roles.

**Example**

```
OS10# show ovssdb-tables manager
Count : 3
Manager table
_uuid          inactivity_probe is_connected max_backoff
other_config status target
-----
```

```
-----
478ec8ca-9c5a-4d29-9069-633af6c48002 [] false 1000 {} {state=BACKOFF}

"ssl:10.16.140.171:6640"
52f2b491-6372-43e0-98ed-5c4ab0ca8542 [] true 1000 {}
{sec_since_connect="37831", sec_since_disconnect="37832", state=ACTIVE}
"ssl:10.16.140.173:6640"
7b8a7e36-6221-4297-b85e-51f910abcb5c [] true 1000 {}
{sec_since_connect="87", sec_since_disconnect="99", state=ACTIVE}
"ssl:10.16.140.172:6640"
OS10#
```

**Supported releases** 10.4.3.0 or later

## show ovssdb-tables tunnel

Displays information about the tunnels created by the physical switch to the service nodes.

**Syntax** show ovssdb-tables tunnel

**Parameters** None

**Default** None

**Command mode** EXEC

**Usage information** This command is available only for netadmin, sysadmin, and secadmin roles.

### Example

```
OS10# show ovssdb-tables tunnel
Count : 2
Tunnel table
_uuid          bfd_params          bfd_config_local          bfd_config_remote          bfd_status
-----
                local                remote
8025d953-acf5-4091-9fa2-75d41953b397 {bfd_dst_ip="55.55.5.5", bfd_dst_mac="00:23:20:00:00:01"} {bfd_dst_ip="2.2.2.2",
bfd_dst_mac="00:50:56:65:b2:3c"} {enable="true", forwarding_if_rx="true", min_rx="1000"} {diagnostic="No
Diagnostic", enabled="true", forwarding="true", remote_state=up, state=up} bb43d2ec-1e60-4367-9840-648a8cc8acff
2d8963da-24d0-4fbd-81e2-fb1a7bba88fd
9853f77a-9db7-47f5-8203-b5b8895d15bd {bfd_dst_ip="55.55.5.5", bfd_dst_mac="00:23:20:00:00:01"} {bfd_dst_ip="2.2.2.3",
bfd_dst_mac="00:50:56:6e:56:9b"} {enable="true", forwarding_if_rx="true", min_rx="1000"} {diagnostic="No Diagnostic",
enabled="true", forwarding="true", remote_state=up, state=up} bb43d2ec-1e60-4367-9840-648a8cc8acff 5eee586b-
e0aa-442b-83ea-16633ec41230
```

**Supported releases** 10.4.3.0 or later

## BGP EVPN for VXLAN

Ethernet Virtual Private Network (EVPN) is a control plane for VXLAN that reduces flooding in the network and resolves scalability concerns. EVPN uses MP-BGP to exchange information between VTEPs. EVPN was introduced in RFC 7432 and is based on BGP MPLS-based VPNs. RFC 8365 describes VXLAN-based EVPN.

The MP-BGP EVPN control plane provides protocol-based remote VTEP discovery, and MAC and ARP learning. This configuration reduces flooding related to L2 unknown unicast traffic. The distribution of host MAC and IP reachability information supports virtual machine (VM) mobility and scalable VXLAN overlay network designs.

The BGP EVPN protocol groups MAC addresses and ARP/neighbor addresses under EVPN instances (EVI) to exchange them between VTEPs. In OS10, each EVI is associated with a VXLAN VNI in 1:1 mapping.

### Benefits of a BGP EVPN-based VXLAN

- Eliminates the flood-and-learn method of VTEP discovery by enabling control-plane learning of end-host L2 and L3 reachability information.
- Minimizes network flooding of unknown unicast and broadcast traffic through EVPN-based MAC and IP route advertisements on local VTEPs.
- Provides support for host mobility.

 **NOTE:** This feature is not supported on the N3224F-ON platform.

## BGP EVPN compared to static VXLAN

OS10 supports two types of VXLAN NVO overlay networks:

- Static VXLAN
- BGP EVPN

Configure and operate static VXLANs and BGP EVPNs for VXLAN in the same way:

- Manually configure the overlay and underlay networks.
- Manually configure each virtual network and VNI.
- Manually configure access port membership in a virtual network.
- Existing routing protocols provision and learn underlay reachability to VTEP peers.

However, static VXLANs and BGP EVPNs for VXLAN differ as described:

**Table 6. Differences between Static VXLAN and VXLAN BGP EVPN**

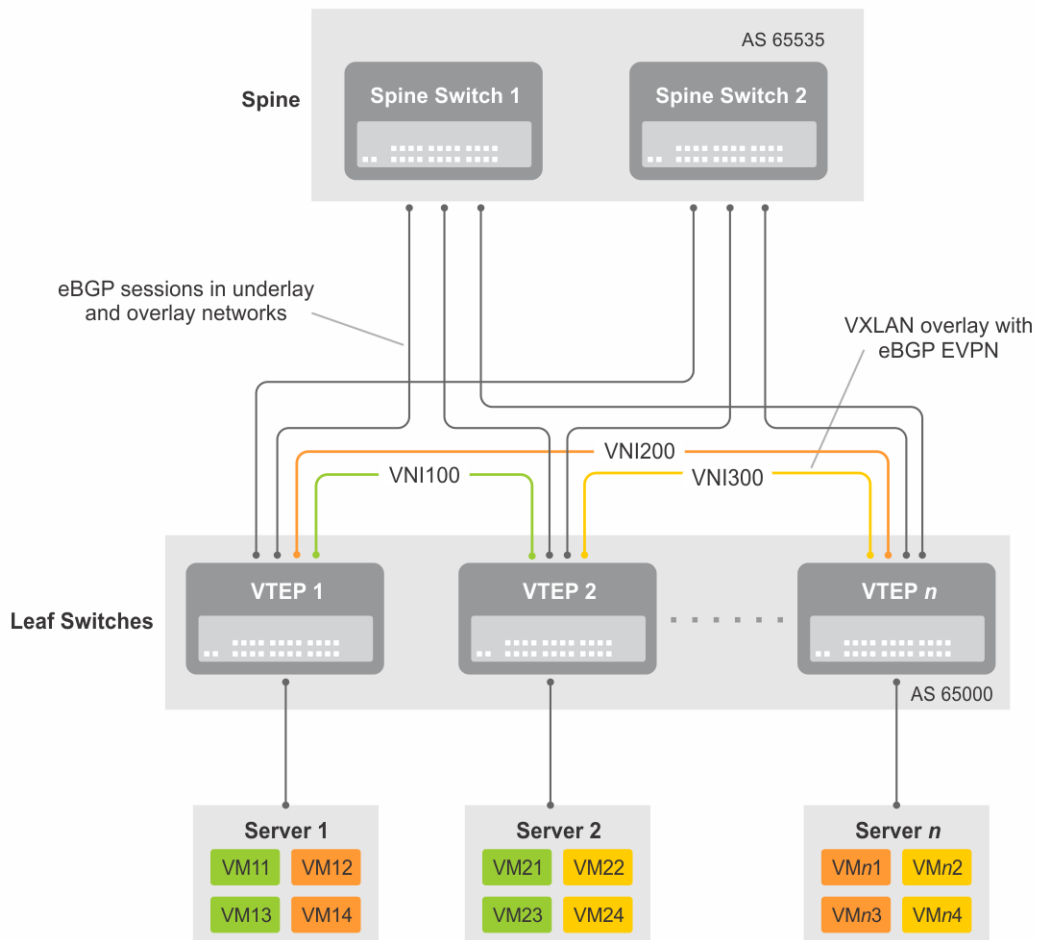
Static VXLAN	VXLAN BGP EVPN
To start sending and receiving virtual-network traffic to and from a remote VTEP, manually configure the VTEP as a member of the virtual network.	No manual configuration is required. Each remote VTEP is automatically learned as a member of a virtual network from the EVPN routes received from the remote VTEP. After a remote VTEP address is learned, VXLAN traffic is sent to, and received from, the VTEP.
Data packets learn remote hosts after decapsulation of the VXLAN header in the data plane.	Remote host MAC addresses are learned in the control plane using BGP EVPN Type 2 routes and MAC/IP advertisements.

## VXLAN BGP EVPN operation

The EVPN address family allows VXLAN to carry EVPN routes in External Border Gateway Protocol (eBGP) and Internal Border Gateway Protocol (iBGP) sessions. In a data center network, use eBGP or iBGP for route exchange in both the IP underlay network and EVPN.

The following sample BGP EVPN topology shows a leaf-spine data center network where eBGP exchanges IP routes in the IP underlay network, and exchanges EVPN routes in the VXLAN overlay network. All spine nodes are in one autonomous system—AS 65535. All leaf nodes are in another autonomous system—AS 65000.

To advertise underlay IP routes, eBGP peer sessions establish between the leaf and spine nodes using an interface IP address. To advertise EVPN routes, eBGP peer sessions between the leaf and spine nodes use a Loopback IP address.



**Figure 3. BGP EVPN topology**

### Leaf nodes

Leaf nodes are typically top-of-rack (ToR) switches in a data center network. They act as the VXLAN tunnel endpoints and perform VXLAN encapsulation and decapsulation. Leaf nodes also participate in the MP-BGP EVPN to support control plane and data plane functions.

Control plane functions include:

- Initiate and maintain route adjacencies using any routing protocol in the underlay network.
- Advertise locally learned routes to all MP-BGP EVPN peers.
- Process the routes that are received from remote MP-BGP EVPN peers and install them in the local forwarding plane.

Data plane functions include:

- Encapsulate server traffic with VXLAN headers and forward the packets in the underlay network.
- Decapsulate VXLAN packets received from remote VTEPs and forward the native packets to downstream hosts.
- Perform underlay route processing, including routing based on the outer IP address.

### Spine nodes

The role of a spine node changes based on its control plane and data plane functions. Spine nodes participate in underlay route processing to forward packets and in the overlay network to advertise EVPN routes to all MP-BGP peers.

Control plane functions include:

- Initiate BGP peering with all neighbor leaf nodes.
- Advertise BGP routes to all BGP peers.
- Initiate and maintain routing adjacencies with all leaf and spine nodes in the underlay network.

Data plane functions include:

- Perform only underlay route processing based on the outer header in VXLAN encapsulated packets.
- Does not perform VXLAN encapsulation or decapsulation.

The BGP EVPN running on each VTEP listens to the exchange of route information in the local overlay, encodes the learned routes as BGP EVPN routes, and injects them into BGP to advertise to the peers. Tunnel endpoints advertise as Type 3 EVPN routes. MAC/IP addresses advertise as Type 2 EVPN routes.

### EVPN instance

An EVPN instance (EVI) spans across the VTEPs that participate in an Ethernet VPN. Each virtual-network tenant segment, that is advertised using EVPN, must associate with an EVI. In OS10, configure EVIs in auto-EVI or manual configuration mode.

- Auto-EVI — After you configure a virtual network on a VTEP, auto-EVI mode automatically creates an EVPN instance. The route distinguisher (RD) and route target (RT) values automatically generate:
  - The EVI ID autogenerates with the same value as the virtual-network ID (VNID) configured on the VTEP and associates with the VXLAN network ID (VNI).
  - A Route Distinguisher autogenerates for each EVI ID. A Route Distinguisher maintains the uniqueness of an EVPN route between different EVPN instances.
  - A Route Target import and export value autogenerates for each EVI ID. A Route Target determines how EVPN routes distribute among EVPN instances.
- Manual EVI configuration — To specify the RD and RT values, manually configure EVPN instances and associate each EVI with the overlay virtual network using the VXLAN VNI. The EVI activates only when you configure the virtual network, RD, and RT values.

In manual EVI configuration, you can either manually configure the RD and RT or have them autoconfigured.

**NOTE:** Dell Technologies recommends using manual EVI for interoperability with network equipment vendors.

### Route distinguisher

The RD is an 8-byte identifier that uniquely identifies an EVI. Each EVPN route is prefixed with a unique RD and exchanged between BGP peers, making the tenant route unique across the network. In this way, overlapping address spaces among tenants are supported.

You can autogenerate or manually configure a RD for each EVI. In auto-EVI mode, the RD is autogenerated. In manual EVI configuration mode, you can autogenerate or manually configure the RD.

As specified in RFC 7432, a manually configured RD is encoded in the format: *4-octet-ipv4-address:2-octet-number*. An autogenerated RD has the format: *vtep-ip-address:evi*.

### Route target

While a RD maintains the uniqueness of an EVPN route among different EVIs, a RT controls the way the EVPN routes are distributed among EVIs. Each EVI is configured with an import and export RT value. BGP EVPN routes advertise for an EVI carry the export RT associated with the EVI. A receiving VTEP downloads information in the BGP EVPN route to EVIs that have a matching import RT value.

You can autogenerate or manually configure the RT import and export for each EVI. In auto-EVI mode, RT autogenerates. In manual EVI configuration mode, you can autogenerate or manually configure the RT.

The RT consists of a 2-octet *type* and a 6-octet *value*. If you autoconfigure a RT, the encoding format is different for a 2-byte and 4-byte AS number (ASN):

- For a 2-byte ASN, the RT *type* is set to 0200 (Type 0 in RFC 4364). The RT *value* is encoded in the format that is described in section 5.1.2.1 of RFC 8365: *2-octet-ASN: 4-octet-number*, where the following values are used in the *4-octet-number* field:
  - Type: 1
  - D-ID: 0
  - Service-ID: VNI
- For a 4-byte ASN, the RT *type* is set to 0202 (Type 2 in RFC 4364). The RT *value* is encoded in the format: *4-octet-ASN: 2-octet-number*, where the *2-octet-number* field contains the EVI ID. In auto-EVI mode, the EVI ID is the same as the virtual network ID (VNID). In 4-byte ASN deployment, OS10 supports RT autoconfiguration if the VNID-to-VNI mapping is the same on all VTEPs.

**NOTE:** Dell Technologies recommends using manual route-target for interoperability with network equipment vendors.

## Configure BGP EVPN for VXLAN

To set up BGP EVPN service in a VXLAN overlay network:

1. Configure the VXLAN overlay network. If you enable routing for VXLAN virtual networks, Integrated Routing and Bridging (IRB) for BGP EVPN is automatically enabled. For more information, see [Configure VXLAN](#).
2. Configure BGP to advertise EVPN routes.
3. Configure EVPN, including the VNI, RD, and RT values associated with the EVPN instance.
4. Verify the BGP EVPN configuration.

### Configuration

1. Configure BGP to advertise EVPN routes.

EVPN requires that you establish MP-BGP sessions between leaf and spine nodes in the underlay network. On each spine and leaf node, configure at least two BGP peering sessions:

- A directly connected BGP peer in the underlay network to advertise VTEP and Loopback IP addresses using the IPv4 unicast address family.
- A BGP peer in the overlay network to advertise overlay information using the EVPN address family. In BGP peer sessions in the overlay, activate only the EVPN address family.

For each BGP peer session in the underlay network:

- a. Create a BGP instance in CONFIGURATION mode. You enter router BGP configuration mode.

```
router bgp as-number
```

- b. Assign an IP address to the BGP instance in ROUTER-BGP mode.

```
router-id ip-address
```

- c. Enter IPv4 address-family configuration mode from ROUTER-BGP mode.

```
address-family ipv4 unicast
```

- d. Advertise the IPv4 prefix to BGP peers in the address family in ROUTER-BGP-ADDRESS-FAMILY mode.

```
network ip-address/mask
```

- e. Return to ROUTER-BGP mode.

```
exit
```

- f. Configure the BGP peer address in ROUTER-BGP mode.

```
neighbor ip-address
```

- g. Assign the BGP neighbor to an autonomous system in ROUTER-BGP-NEIGHBOR mode.

```
remote-as as-number
```

- h. Enable the peer session with the BGP neighbor in ROUTER-BGP-NEIGHBOR mode.

```
no shutdown
```

- i. Return to ROUTER-BGP mode.

```
exit
```

For each BGP peer session in the overlay network:

- a. Configure the BGP peer using its Loopback IP address on the VTEP in ROUTER-BGP mode.

```
neighbor loopback-ip-address
```

- b. Assign the BGP neighbor Loopback address to the autonomous system in ROUTER-BGP-NEIGHBOR mode. The neighbor Loopback IP address is the source interface on the remote VTEP.

```
remote-as as-number
```

- c. Use the local Loopback address as the source address in BGP packets sent to the neighbor in ROUTER-BGP-NEIGHBOR mode.

```
update-source loopback0
```

- d. Send an extended community attribute to the BGP neighbor in ROUTER-BGP-NEIGHBOR mode.

```
send-community extended
```

- e. Enable the peer session with the BGP neighbor in ROUTER-BGP-NEIGHBOR mode.

```
no shutdown
```

- f. Configure the L2 VPN EVPN address family for VXLAN host-based routing to the BGP peer in ROUTER-BGP-NEIGHBOR mode.

```
address-family l2vpn evpn
```

- g. Enable the exchange of L2VPN EVPN addresses with the BGP peer in ROUTER-BGP-NEIGHBOR mode.

```
activate
```

- h. Return to ROUTER-BGP mode.

```
exit
```

- i. Enter IPv4 address-family configuration mode from ROUTER-BGP mode.

```
address-family ipv4 unicast
```

- j. Disable the exchange of IPv4 addresses with BGP peers in ROUTER-BGP mode.

```
no activate
```

- k. Return to ROUTER-BGP-NEIGHBOR mode.

```
exit
```

- l. (Optional) If all the leaf switches are configured in the same ASN:

- On each leaf switch, enter L2VPN EVPN address-family configuration mode from ROUTER-BGP-NEIGHBOR mode. Activate the exchange of L2VPN EVPN addresses with BGP peers. Configure the switch to accept a route with the local AS number in updates received from a peer in ROUTER-BGP-NEIGHBOR-AF mode.

```
OS10(config-router-bgp-neighbor)# address-family l2vpn evpn
OS10(config-router-neighbor-af)# activate
OS10(config-router-neighbor-af)# allowas-in 1
OS10(config-router-neighbor-af)# exit
OS10(config-router-bgp-neighbor)# exit
```

- On each spine switch, disable sender-side loop detection to leaf switch neighbors in ROUTER-BGP-NEIGHBOR-AF mode.

```
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
```

- m. (Optional) In a VLT deployment, on each leaf switch, configure the number of multi-hop peer routes in ROUTER-BGP-NEIGHBOR mode to ensure that the BGP EVPN peer session establishes over the VLT VTEP peer if all local links to spine switches are down.

```
OS10(conf-router-neighbor)# ebgp-multihop 1
```

## 2. Configure EVPN.

An EVPN instance (EVI) spans across the VTEPs that participate in the EVPN. In OS10, configure an EVI in auto-EVI or manual configuration mode.

### • Auto-EVI mode

- a. Enable the EVPN control plane in CONFIGURATION mode.

```
evpn
```

- b. Enable auto-EVI creation for overlay virtual networks in EVPN mode. Auto-EVI creation is supported only if BGP EVPN is used with 2-byte AS numbers and if at least one BGP instance is enabled with the EVPN address family. No further manual configuration is allowed in auto-EVI mode.

```
auto-evi
```

- **Manual EVI configuration mode**

- a. Enable the EVPN control plane in CONFIGURATION mode.

```
evpn
```

- b. Manually create an EVPN instance in EVPN mode. The range is from 1 to 65535.

```
evi id
```

- c. Configure the Route Distinguisher in EVPN EVI mode.

```
rd {A.B.C.D:[1-65535] | auto}
```

Where:

- o rd A.B.C.D:[1-65535] configures the RD with a 4-octet IPv4 address then a 2-octet-number.
- o rd auto automatically generates the RD.

- d. Configure the RT values in EVPN EVI mode.

```
route-target {auto | value [asn4] {import | export | both}}
```

Where:

- o route-target auto auto-configures an import and export value for EVPN routes.
- o route-target value [asn4]{import | export | both} configures an import or export value for EVPN routes in the format *2-octet-ASN: 4-octet-number* or *4-octet-ASN: 2-octet-number*.
  - The *2-octet* ASN number is 1 to 65535.
  - The *4-octet* ASN number is 1 to 4294967295.

To configure the same value for the RT import and export values, use the *both* option. *asn4* advertises a 2-byte AS number as a 4-byte route target value. If you specify the *asn4* option, configure the VXLAN network ID associated with the EVPN instance in EVPN EVI mode, from 1 to 16,777,215. Configure the same VNI value that you configure for the VXLAN virtual network. For more information, see [Configure VXLAN](#).

```
vni vni
```

- 3. Verify the BGP EVPN configuration.

### Display the EVPN instance configuration

```
OS10# show evpn evi 1
EVI : 65447, State : up
  Bridge-Domain      : (Virtual-Network)100, (VNI)100
  Route-Distinguisher : 1:110.111.170.102:65447(auto)
  Route-Targets      : 0:101:268435556(auto) both
  Inclusive Multicast : 110.111.170.107
```

### Display the VXLAN overlay for the EVPN instance

```
OS10# show evpn vxlan-vni
VXLAN-VNI  EVI  Virtual-Network-Instance
100001     1    1
100010     2    2
```

### Display the BGP neighbors in the EVPN instances

```
OS10# show ip bgp neighbors 110.111.170.102
BGP neighbor is 110.111.170.102, remote AS 100, local AS 100 internal link
BGP version 4, remote router ID 110.111.170.102
BGP state ESTABLISHED, in this state for 04:02:59
Last read 00:21:21 seconds
Hold time is 180, keepalive interval is 60 seconds
Configured hold time is 180, keepalive interval is 60 seconds
Fall-over disabled
```

```

Received 311 messages
  2 opens, 2 notifications, 3 updates
  304 keepalives, 0 route refresh requests
Sent 307 messages
  4 opens, 0 notifications, 2 updates
  301 keepalives, 0 route refresh requests
Minimum time between advertisement runs is 30 seconds
Minimum time before advertisements start is 0 seconds
Capabilities received from neighbor for IPv4 Unicast:
MULTIPROTO_EXT(1)
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
4 OCTET_AS(65)
MP_L2VPN_EVPN
Capabilities advertised to neighbor for IPv4 Unicast:
MULTIPROTO_EXT(1)
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
4 OCTET_AS(65)
MP_L2VPN_EVPN
Prefixes accepted 1, Prefixes advertised 1
Connections established 2; dropped 0
Last reset never
Prefixes ignored due to:
  Martian address 0, Our own AS in AS-PATH 0
  Invalid Nexthop 0, Invalid AS-PATH length 0
  Wellknown community 0, Locally originated 0

Local host: 110.111.180.195, Local port: 43081
Foreign host: 110.111.170.102, Foreign port: 179

```

#### Display the BGP L2VPN EVPN address family

```

OS10# show ip bgp l2vpn evpn
BGP local RIB : Routes to be Added , Replaced , Withdrawn
BGP local router ID is 110.111.170.102
Status codes: s suppressed, S stale, d dampened, h history, * valid, > best
Path source: I - internal, a - aggregate, c - confed-external,
r - redistributed/network, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

```

Network	Next Hop	Metric	LocPrf	Weight	Path
*>r Route distinguisher: 110.111.170.102:65447					
[3]:[0]:[32]:[110.111.170.102]/152	110.111.170.102	0	100	32768	?
*> Route distinguisher: 110.111.170.107:64536					
[3]:[0]:[32]:[110.111.170.107]/152	110.111.170.107	0	100	0	100 101 ?

#### Display the EVPN routes for host MAC addresses

```

OS10# show evpn mac
Type -(lcl): Local (rmt): remote

```

EVI	Mac-Address	Type	Seq-No	Interface/Next-Hop
50	00:00:00:aa:aa:aa	rmt	0	55.1.1.3
50	00:00:00:cc:cc:cc	lcl	0	ethernet1/1/8:1

```

OS10# show evpn mac evi 50
Type -(lcl): Local (rmt): remote

```

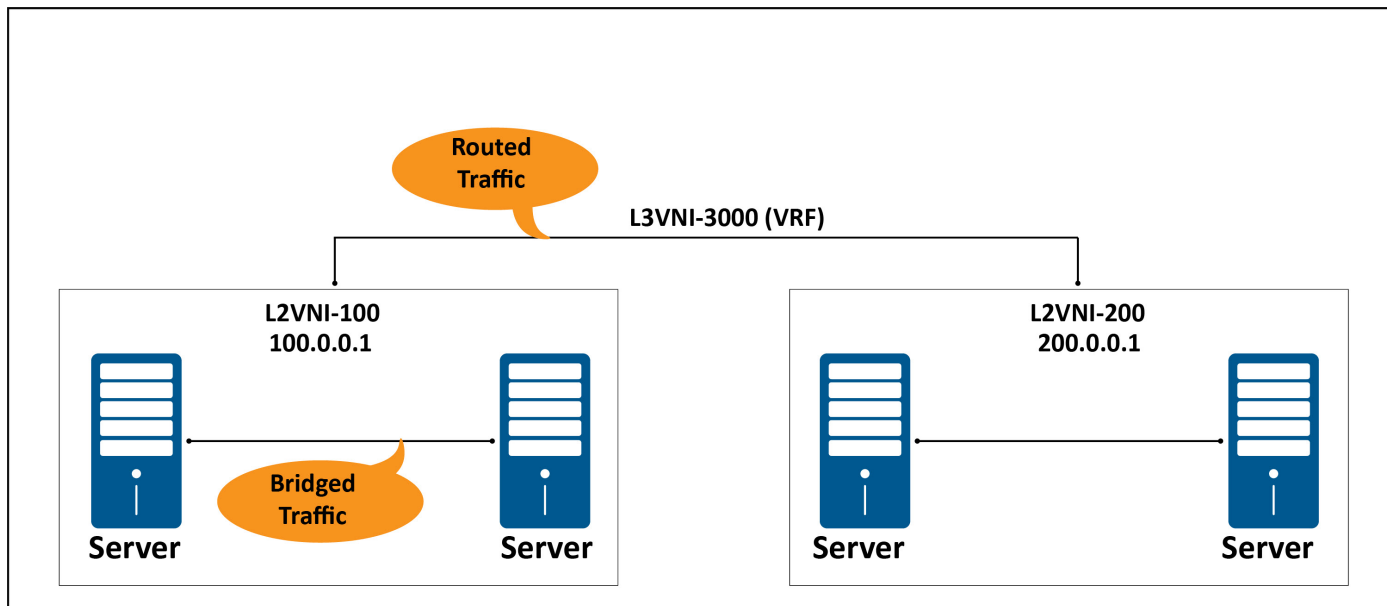
EVI	Mac-Address	Type	Seq-No	Interface/Next-Hop
50	00:00:00:aa:aa:aa	rmt	0	55.1.1.3
50	00:00:00:cc:cc:cc	lcl	0	ethernet1/1/8:1

## BGP EVPN with VXLAN overlay - Multi tenancy

In BGP-EVPN with VXLAN overlay, multitenancy is achieved by logically isolating the traffic from different entities of a tenant in a shared VXLAN network. The tenant can be a group of hosts or servers and tenant entities can be group of VMs in a server or applications within a VM. Each such tenant entity can be mapped to a VLAN or VRF depending on the type of network segments (Layer 2 or Layer 3) they belong to.

Each tenant that belongs to the same Layer 2 network is mapped to a VLAN and in turn each such VLAN is mapped to the same Layer 2 VNI. So, traffic between the tenant entities belonging to the same Layer 2 virtual network is tagged with a unique Layer 2 VNI and this traffic is bridged across the VXLAN network. The traffic between different Layer 2 virtual networks is routed. This routing is achieved by grouping Layer 2 virtual networks into a VRF. Each Layer 2 virtual network is assigned with an IP address.

The following figures captures the logical separation of tenant traffic based on the network segment they belong to:



## Layer 2 Multi tenancy

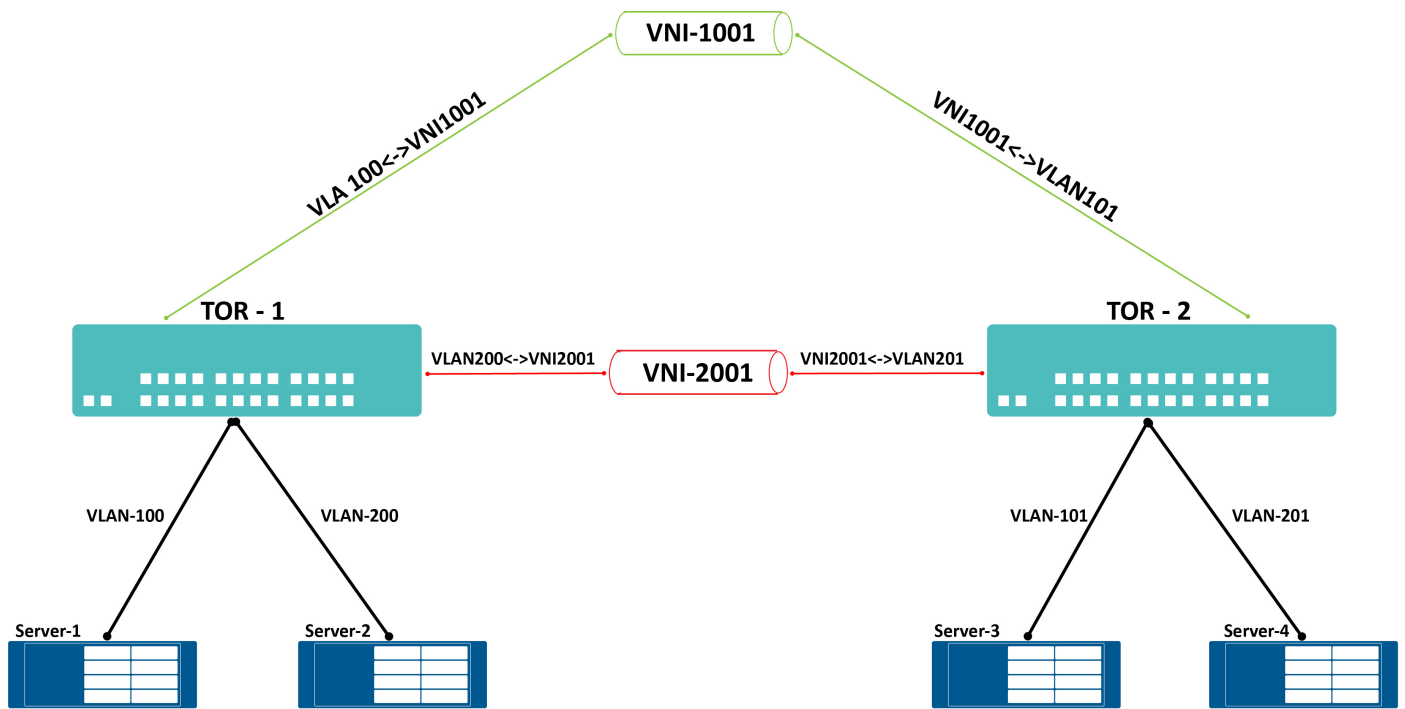
In SmartFabric DellEMC OS10, Layer 2 multitenancy is achieved in the following two ways: VLAN attach Mode and Port-scoped VLAN Mode.

### VLAN attach mode

In this mode, each tenant entity is mapped with one VLAN and each such VLAN is mapped 1:1 with one Layer 2 VNI.

In the following example, Server-1 and Server-3 belong to the same tenant.

Traffic from Server-1 is tagged with VLAN-100 and it is mapped to VNI1001. Similarly, traffic from Server-3 is tagged with VLAN-101 and mapped to VNI1001.



### Port-scoped VLAN mode

In the port-scoped VLAN to VNI mapping mode, instead of attaching a VLAN to the virtual network, packets tagged with a specific vlan-id on a specific port are assigned to a virtual network for bridging.

To assign a packet tagged with a specific vlan-id on a specific port to a virtual network, perform the following steps:

1. Create a virtual network for VLAN tunnelling.

```
OS10(config)# virtual-network 10000
```

2. Assign a member interface corresponding to a specific vlan-id on a specific port to the virtual network:

```
OS10(config-vn)# member interface ethernet1/1/1 vlan 10
```

Each vlan-id on a port cannot be a part of more than one virtual network bridge.

For example, if you apply the following configuration, an error message appears indicating that the operation is not allowed:

```
(config)# virtual-network 10
(config-vn)# member-interface ethernet 1/1/1 vlan 6
(config)# virtual-network 11
(config-vn)# member-interface ethernet 1/1/1 vlan 6
```

The following error message appears:

```
% Error: Operation not allowed. Reason:Only one VLAN can be mapped to an interface in a
particular Virtual-Network.
```

To reserve vlan-id to be used internally:

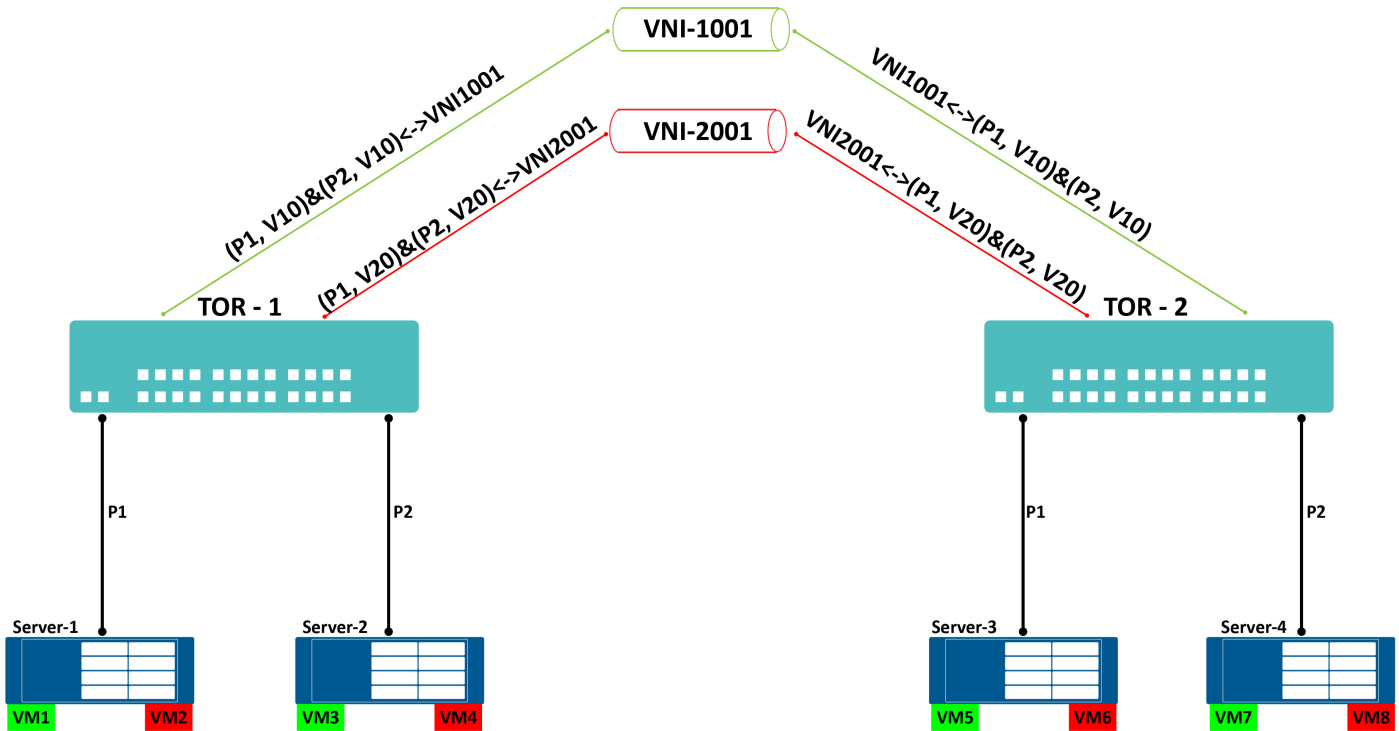
- OS10(config)# virtual-network untagged-vlan 4001

**NOTE:** Due to hardware limitations, you cannot use the default VLAN as the reserved VLAN for untagged virtual network members. The hardware expects a port configured as an untagged member of the virtual network to also be an untagged member of the reserved VLAN used (unlike the case of the tagged PV membership to virtual network). However, an untagged virtual network member port cannot be an untagged member of the default VLAN; because it results in traffic leaking. Therefore, a dedicated VLAN that does not have any other ports to be reserved as the untagged VLAN is required.

To add untagged ports using the port-scoped method, first reserve a vlan-id that you want to use internally. You need to reserve only one such VLAN for the entire switch; this VLAN is used internally for all untagged ports in all virtual networks.

Use the following commands to add the untagged port to the virtual network:

```
OS10(config)# virtual-network 10000
OS10(config-vn)# member interface ethernet1/1/1 untagged
```

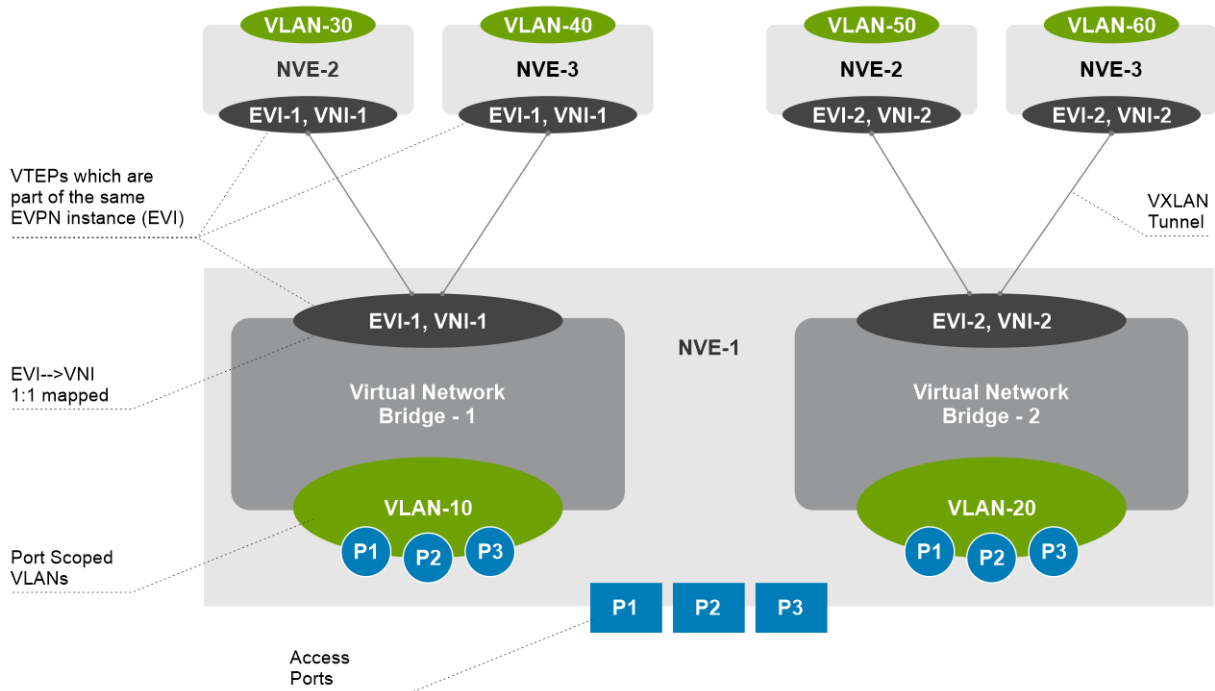


## Enabling EVPN services

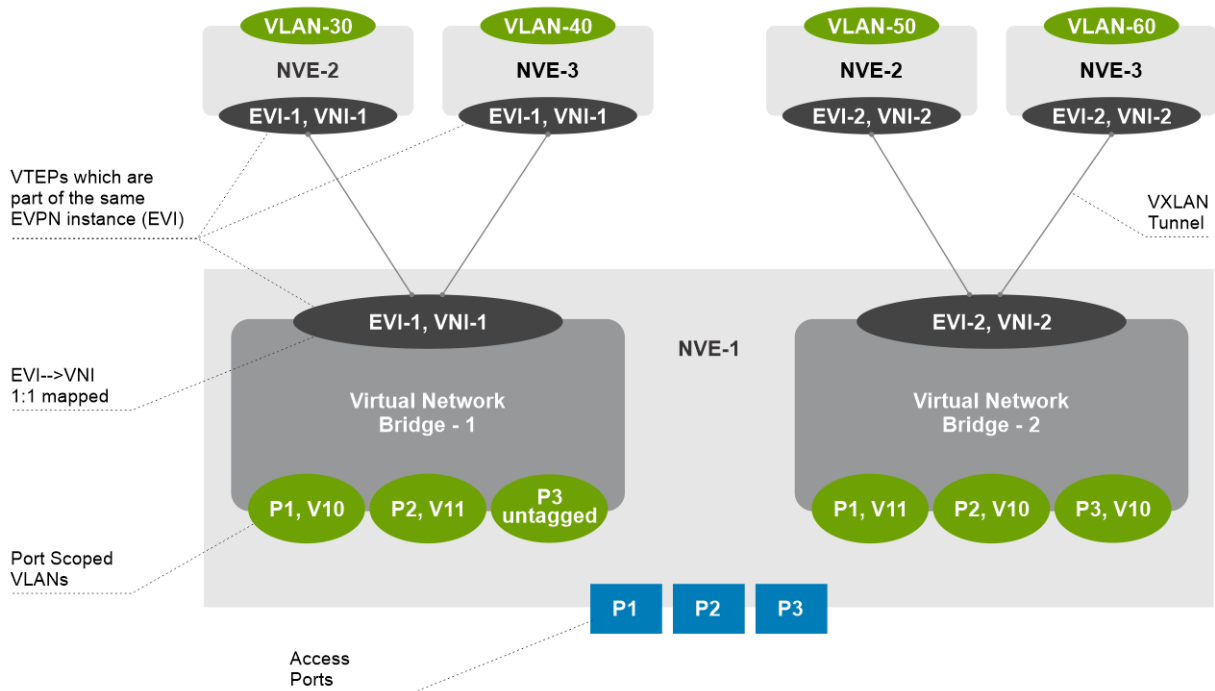
Enabling MP-BGP EVPN service involves the following sequence:

1. Setting up MP-BGP control plane to advertise L2VPN or EVPN address family in BGP control plane.
2. Enabling EVPN.
3. Setting up the EVPN Instance (EVI), which spans across multiple PEs or VTEPs participating in EVPN services.
4. Associating a virtual network (VNI) with EVI setting up the forwarding path, to perform VXLAN encapsulation and decapsulation.

The following diagram depicts an NVE (NVE-1) participating in two EVPN instances (EVI-1, EVI-2), in which two other NVEs (NVE-2, NVE3) are also participating:



The following diagram depicts an NVE (NVE1) with P1, P2, P3, V10, and V11:



## Setting up MP-BGP control plane

Each NVE participating in EVPN should peer up with each other NVE through MP-BGP control plane. They also advertise their EVPN processing capability. In SmartFabric DellEMC OS10, this feature is enabled using the following commands:

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 45.0.0.1
```

```
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
```

- You must enable EVPN address family on the BGP peers to carry EVPN routes.

## Enable EVPN

The following sequence depicts the enabling of EVPN service in SmartFabric DellEMC OS10:

1. `switch(config)# evpn`

Enabling EVPN triggers BGP to advertise EVPN capability (with AFI=25 and SAFI=70) to all the BGP peers in a given AS.

2. `switch(config)# no evpn`

Disabling EVPN triggers BGP to withdraw the EVPN capability.

## Configuring an EVPN Instance (EVI)

### Auto EVI Configuration mode

In this mode, an EVI is auto-allocated for each virtual network configured in the NVE.

The EVI Index is the same as the virtual network identifier.

The Route Distinguisher(RD) and the Route Target(RT) are also automatically generated as follows:

- The Route Distinguisher is Type 1 - derived from the overlay network source IP and the EVI index (Virtual-Network ID).
- The Route Target varies depending on the AS:
  - 2 byte AS: RT will be Type 0 derived from the 2 byte AS and the 3 byte VNI. (Type Encoded as 0x0002).
  - 4 byte AS: RT will be Type 2 derived from the 4 byte AS and the 2 byte EVI which is same as the virtual network ID that you have configured. (Type encoded as 0x0202)

This command is rejected if there are no BGP instances with EVPN address family enabled.

The Route target used for any given VNI needs to be identical on all VTEPs in order for EVPN to synchronize MAC addresses on all VTEPs.

In the case of 2 byte AS this is true for auto-derivation, since all the VTEPs are in the same AS and use the same VNI. In the case of 4 byte AS, the EVPN RFC recommends configuring manual Route target. However, SmartFabric DellEMC OS10 switches support auto-derived Route targets even in case of 4 byte AS as follows:

- The Route target requires a 2 byte representation for each VNI.
- On SmartFabric DellEMC OS10, the 2 byte representation for a VNI is its corresponding virtual network identifier.
- As long as the virtual network ID to VNI mapping is configured identical on all VTEPs, they will be able to use auto-derived Route targets even with 4 byte AS.

```
switch(config)# evpn
switch(config-evpn)# auto-evi
```

### Explicit EVI Configuration mode

In this mode, an EVI is configured explicitly.

This mode allows User to configure Route Distinguisher and Route Targets explicitly.

The EVPN RFC recommends configuring manual Route Target in case of 4 byte AS.

```
switch(config)# evpn
switch(config-evpn)# evi 100
```

### Auto configuration of RD and RT

The Route distinguisher is Type 1 - derived from the overlay network source IP and the user supplied EVI index.

The Route target is Type 2 - derived from the 4 byte AS and the user supplied EVI Index. (Type Encoded as 0x0202).

```
switch(config)# evpn
switch(config-evpn)# evi 10
switch(config-evpn-evi-10)# rd
A.B.C.D:[1..65535] 4-octet-ipv4addr:2-octet-number
auto Enable auto rd mode
switch(config-evpn-evi-10)# rd auto

switch(config)# evpn
switch(config-evpn)# evi 10
switch(config-evpn-evi-10)# rd auto
switch(config-evpn-evi-10)# route-target
{import | export | both} {<value>|auto}
switch(config-evpn-evi-10)# route-target both auto
```

## Explicit configuration of RD and RT

The following command sequence depicts the explicit configuration of RD and RT:

```
switch(config)# evpn
switch(config-evpn)# evi 10
switch(config-evpn-evi)# rd
A.B.C.D:[1..65535] 4-octet-ipv4addr:2-octet-number
auto Enable auto rd mode
switch(config-evpn-evi)# rd 111.111.111.111:65536

switch(config)# evpn
switch(config-evpn)# evi 10
switch(config-evpn-evi-10)# rd 111.111.111.111:65535
switch(config-evpn-evi-10)# route-target
{import | export | both} {<value>|auto}
switch(config-evpn-evi-10)# route-target import
1..65535:1..4294967295 2-octet-asn:4-octet-number
1..4294967295:1..65535 4-octet-asn:2-octet-number
auto Enable auto route target mode
switch(config-evpn-evi-10)# route-target import 1:2
switch(config-evpn-evi-10)# route-target import 1:3
asn4 force 4-octet-asn:2-octet-number
switch(config-evpn-evi-10)# route-target import 1:3 asn4
switch(config-evpn-evi-10)# route-target export 1:4
```

### NOTE:

1. If you want to configure both import and export RT with the same value, SmartFabric DellEMC OS10 recommends to use "route-target both <value>" or "route-target both auto" commands.
2. If you explicitly configure, RT import and export with the same value, the last configured <type, value> overwrites the old configured <type, value>

```
Example :
route-target import 1:2
route-target export 1:2 <--- This overwrites/deletes RT import value.
route-target both 1:2 <--- Will configure RT export/import values with 1:2
```

```
LVTEP1(config-evpn-evi-400)# route-target 1:1
asn4 Force 4-octet-asn:2-octet-number
<import/export/both> Set route target type
```

3. By default, Route target values are configured with 2Byte ASN value. If you want to explicitly advertise a 4Byte ASN, use the following configuration:

```
route-target 1:2 asn4 <--- This enables, 4byte ASN to be encoded in the Route Target
```

## Associate VNI with EVI (Setting up EVPN Control Plane)

A VNI has to be associated with an EVI, to advertise, VNI association with EVI in the MP-BGP control plane.

```
switch(config)# evpn
switch(config-evpn)# evi 10
switch(config-evpn-evi-10)# vni 100
```

## Setting up VXLAN forwarding

To setup the VXLAN forwarding, for the VNI associated with an EVI, you must configure a virtual network with the given VNI.

```
switch(config)# evpn
switch(config-evpn)# evi 10
switch(config-evpn-evi-10)# vni 100

switch(config)# virtual-network bridge-1
switch(config-vn)# vxlan-vni 100
switch(config-vn)# member interface ethernet1/1/1 vlan 10
```

### **NOTE:**

- With this configuration, BGP notifies all the BGP peers with inclusive multicast route.
- On Local NVE, VXLAN forwarding path is enabled.
- On the attached member interfaces, MAC learning is enabled.

## EVPN configuration events and system behavior

In SmartFabric DellEMC OS10, configuring EVPN instances and attaching virtual networks to EVPN instances, can happen in any order. The following section summarizes, configuration sequence and SmartFabric DellEMC OS10 behavior in each such event:

### Prerequisites:

- Each Leaf and Spine in the network needs to be configured with eBGP as IGP protocol and neighborhood should be enabled with both "address family ipv4 unicast" and "address family I2vpn evpn".
- In this example, numbered interfaces (connected interfaces) are used for the underlay BGP peering. These sessions distribute the peer reachability. They only distribute the IPV4 address family routes.
- Loopback addresses are used for the overlay sessions. They are enabled with "I2vpn evpn" address family and distribute only the EVPN routes.

The following table captures the EVPN instance configuration sequence:

**Table 7. EVPN instance configuration sequence**

Spine	Leaf
<pre>OS10# show running-configuration bgp ! router bgp 100 router-id 110.111.170.195 ! address-family ipv4 unicast redistribute connected ! neighbor 11.1.1.2 remote-as 101 no shutdown ! address-family ipv4 unicast no sender-side-loop-detection ! neighbor 110.111.170.102 ebgp-multihop 2 remote-as 101 send-community extended</pre>	<pre>OS10# show running-configuration bgp ! router bgp 101 router-id 110.111.170.102 ! address-family ipv4 unicast redistribute connected ! neighbor 11.1.1.1 remote-as 100 no shutdown ! address-family ipv4 unicast allowas-in 1 ! neighbor 110.111.170.195 ebgp-multihop 2 remote-as 100 send-community extended</pre>

**Table 7. EVPN instance configuration sequence**

Spine	Leaf
<pre> update-source loopback1 no shutdown ! address-family ipv4 unicast no activate ! address-family l2vpn evpn activate !                     </pre>	<pre> update-source loopback1 no shutdown ! address-family ipv4 unicast no activate ! address-family l2vpn evpn activate                     </pre>

```

OS10# show ip bgp l2vpn evpn summary
BGP router identifier 110.111.170.102 local AS number 101
Neighbor          AS              MsgRcvd      MsgSent
Up/Down          State/Pfx
110.111.170.195  100          1687         1685
1d:00:24:44
OS10# show ip bgp l2vpn evpn neighbors
BGP neighbor is 110.111.170.195, remote AS 100, local AS 101 external link
BGP version 4, remote router ID 110.111.170.195
BGP state ESTABLISHED, in this state for 2 days 17:50:18
Last read 00:30:35 seconds
Hold time is 180, keepalive interval is 60 seconds
Configured hold time is 180, keepalive interval is 60 seconds
Fall-over disabled
EBGP multihop enabled, multihop TTL set to 2
Received 4542 messages
1 opens, 0 notifications, 2 updates
4539 keepalives, 0 route refresh requests
Sent 4542 messages
1 opens, 0 notifications, 2 updates
4539 keepalives, 0 route refresh requests
Minimum time between advertisements runs is 30 seconds
Minimum time before advertisements start is 0 seconds
Capabilities received from neighbor for IPv4 Unicast:
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
4_OCTET_AS(65)
MP_L2VPN_EVPN(1)
Capabilities advertised to neighbor for IPv4 Unicast:
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
4_OCTET_AS(65)
MP_L2VPN_EVPN(1)
Prefixes accepted 2, Prefixes advertised 2
Connections established 1; dropped 0
Last reset never
Local host: 110.111.170.102, Local port: 179
Foreign host: 110.111.170.195, Foreign port: 44115
                    
```

The following table captures the configuration events in the Explicit EVI configuration mode:

**Table 8. Explicit EVI configuration events**

<pre> Current system state ---&gt; Current configuration event                       v                     </pre>	<pre> Start up 1. evpn disabled                     </pre>	<pre> 2. evi 10       vni 100       rd &lt;x:y&gt;       rt &lt;x:y&gt;                     </pre>
<pre> 1. evpn                     </pre>	<pre> BGP peer is notified with EVPN capability                     </pre>	<pre> N/A                     </pre>

**Table 8. Explicit EVI configuration events (continued)**

Current system state ---> Current configuration event     v	Start up 1. evpn disabled	2. evi 10 vni 100 rd <x:y> rt <x:y>
2. virtual-network 10 vni 100 vxlan-	N/A	<ul style="list-style-type: none"> <li>• BGP Peer is notified with inclusive multicast route update.</li> <li>• Forwarding path is enabled with VXLAN overlay.</li> <li>• Remote MAC learning is disabled.</li> </ul>
3. evi deleted	N/A	<ul style="list-style-type: none"> <li>• BGP Peer is notified with inclusive multicast route withdrawal.</li> <li>• All the advertised local MAC routes are withdrawn.</li> </ul>
4. virtual-network deleted	N/A	<ul style="list-style-type: none"> <li>• BGP peer is notified that inclusive multicast route is withdraw</li> <li>• Forwarding path is disabled for VXLAN overlay.</li> <li>• All the advertised local MAC routes are withdrawn.</li> </ul>

The following table captures the configuration events in Auto EVI configuration mode:

**Table 9. Auto EVI configuration events**

Current system state ---> Current configuration event     v	Start up 1. evpn disabled	2. auto-evi
1. evpn	BGP peer is notified with EVPN capability	N/A
2. virtual-network 10 vni 100 vxlan-	N/A	<ul style="list-style-type: none"> <li>• A new EVI with VNI 100 is created automatically.</li> <li>• RD and RT are generated automatically.</li> <li>• BGP Peer is notified with inclusive multicast route update.</li> <li>• Forwarding path is enabled with VXLAN overlay.</li> <li>• Remote MAC learning is disabled.</li> </ul>
4. virtual-network deleted	N/A	<ul style="list-style-type: none"> <li>• EVI with VNI 100 is deleted.</li> <li>• BGP Peer is notified with inclusive multicast route withdrawal notification.</li> <li>• Forwarding path is disabled for VXLAN overlay.</li> <li>• All the advertised local MAC routes are withdrawn.</li> </ul>

## EVPN constructs

EVPN provides constructs that are required to enable VXLAN overlay in Data Centers using MP-BGP control plane.

The following table captures the features that are supported to enable the VXLAN Overlay, these features form the basic constructs for EVPN:

**Table 10. Features supported to enable VXLAN overlay**

Feature name	SmartFabric DellEMC OS10 support
Advertising the BGP capability supporting the AFI of 25 (L2VPN) and a SAFI of 70 (EVPN).	10.4.2
Exchange and understand the EVPN NLRI from the MP-BGP peers.	10.4.2
Supporting the following EVPN Route types: <ul style="list-style-type: none"> <li>Type 1 - Ethernet Auto-Discovery (A-D) route</li> <li>Type 2 - MAC/IP Advertisement route</li> <li>Type 3 - Inclusive Multicast Ethernet Tag route</li> <li>Type 4 - Ethernet Segment route</li> </ul>	10.4.2 (Only Type-2 and Type-3 routes)
Support following EVPN service interface types: <ul style="list-style-type: none"> <li>VLAN-based service interfaces.</li> </ul>	10.4.2
Control plane MAC learning using MAC or IP advertisement route.	10.4.2
Auto-derived route targets.	10.4.2
MAC mobility procedures.	10.4.2
Ingress replication for BUM traffic	10.4.2

## EVPN service types

This section describes various EVPN service types.

The following table captures the various EVPN service types:

**Table 11. EVPN service types**

Service type	Steps	Supported in SmartFabric DellEMC OS10	Description
VLAN based	<ol style="list-style-type: none"> <li>VLAN --&gt; VNI: 1:1</li> <li>VNI --&gt; EVI: 1:1</li> <li>Single FDB per VLAN</li> <li>VLAN is stripped at encapsulation</li> <li>VLAN translation: Yes</li> </ol>	Yes	
VLAN bundle	<ol style="list-style-type: none"> <li>VLAN --&gt; VNI: All:1</li> <li>VNI --&gt; EVI: 1:1</li> <li>Single FDB for all VLANs</li> <li>VLAN is carried in the packet</li> <li>VLAN translation: No</li> </ol>	No	
Port based	<ol style="list-style-type: none"> <li>VLAN --&gt; VNI: N:1</li> <li>VNI --&gt; EVI: 1:1</li> <li>Single FDB: Yes</li> <li>VLAN is carried in the packet</li> <li>VLAN translation: No</li> </ol>	No	Special case of VLAN bundle

**Table 11. EVPN service types (continued)**

Service type	Steps	Supported in SmartFabric DellEMC OS10	Description
VLAN aware bundle	<ol style="list-style-type: none"> <li>1. VLAN --&gt; VNI: 1:1</li> <li>2. VNI --&gt; EVI: N:1</li> <li>3. Single FDB per VLAN</li> <li>4. VLAN is stripped at encapsulation</li> <li>5. VLAN translation: Yes</li> </ol>	No	
Port based VLAN aware	<ol style="list-style-type: none"> <li>1. VLAN --&gt; VNI: 1:1</li> <li>2. VNI --&gt; EVI: All:1</li> <li>3. Single FDB per VLAN</li> <li>4. VLAN is stripped at encapsulation</li> <li>5. VLAN translation: No</li> </ol>	No	Special case of VLAN aware bundle

## EVPN route types

The EVPN NLRI is carried in MP-BGP advertisements with an Address Family Identifier (AFI) of 25 (L2VPN) and a Subsequent Address Family Identifier (SAFI) of 70 (EVPN).

The following table captures the various EVPN route types:

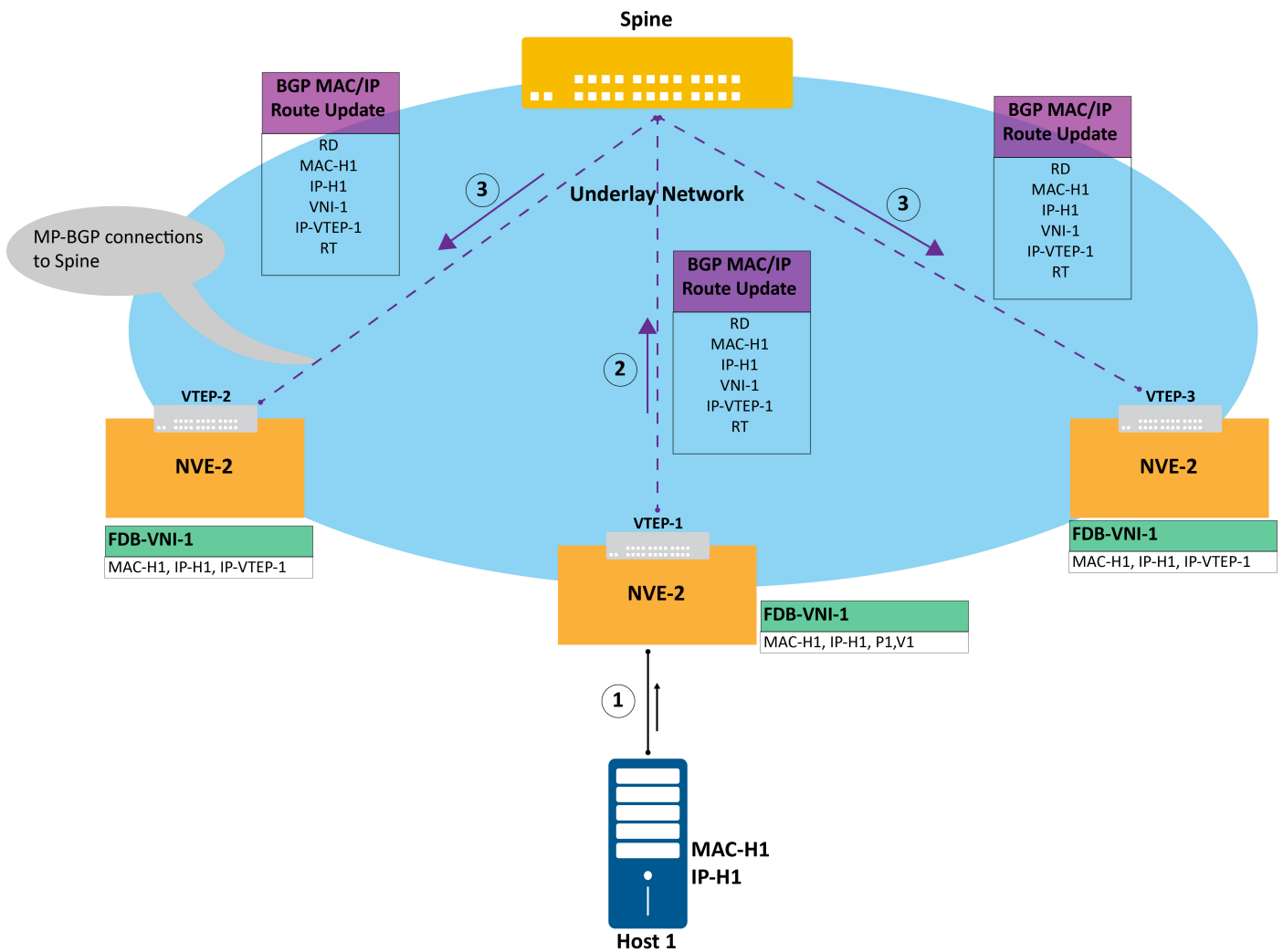
**Table 12. EVPN route types**

Route type	Description	Usage	Optional attributes carried in each	Supported release
2	MAC or IP -VNI Advertisement Route	Unicast forwarding , Proxy ARP Default gateway	MAC mobility extended community, Default gateway	10.4.2
3	Inclusive Multicast Route	BUM Traffic	PMSI tunnel attributes	10.4.2
5	IP Prefix Route	IP Route Advertisement	N/A	10.5.x.x

## MAC learning using EVPN control plane

This section describes MAC learning using EVPN control plane.

The following figure illustrates local and remote MAC learning:



In this topology diagram, NVE-1, NVE-2, and NVE-3 are peering with the Spine switch. VTEP-1 learns the MAC and IP addresses of the connected Host H1, when H1 either sends an ARP request for remote Host or Gratuitous ARP. VTEP-1 also learns about Host1 being part of VNI-1 as it is connected by Port P1, VLAN V1. (P1,V1) is configured to be part of the bridge domain VNI-1. VTEP-1 advertises this information about H1 to all the remote VTEPs using MP-BGP EVPN control plane.

The following captures the route updates consisting of L2VPN AFI and EVPN SAFI values along with the NLRI attributes:

**Table 13. NLRI attributes**

NLRI attributes	Description
Route Distinguisher RD	Encoded as 8 Byte value <Type+IP address of VTEP-1+EVI>
MAC-H1	MAC address of the Host attached to the VTEP-1
IP-H1	IP Address of the Host attached to the VTEP-1
VNI-1	VNID of the Bridge domain that Host is part of
IP-VTEP-1	IP Address of the VTEP-1 itself as the Next Hop to reach the Host H1
RT - Export	Export value of the Route Target

When the remote VTEPs receive this MAC or IP route update, they compare the RT export attribute with the local import attributes. If they match, the route is downloaded and installed in the respective FDB.

## MAC events and SmartFabric DellEMC OS10 behavior

The following table summarizes the behavior of SmartFabric DellEMC OS10 for various events that trigger the MAC processing:

Prerequisites:

- Underlay IGP neighborhood is established with BGP.
- BGP Peering with L2VPN is established.
- EVPN service is enabled.
- EVI is configured with VNI in the control plane.
- Virtual network configured with VNI and VXLAN overlay is enabled in the forwarding path.

**Table 14. Behavior of SmartFabric DellEMC OS10 for various events - Local MACs**

<b>Local MACs Events</b>	<b>System behavior</b>
VLAN is attached to Virtual Network.	<p>Expectation: MAC Learning is enabled on all the VLAN member interfaces. MACs learnt on these member interfaces are advertised to the BGP Peers.</p> <ul style="list-style-type: none"> <li>• MACs learnt on the VLAN prior to attachment with virtual-network are all pulled by the virtual-network and are published one at a time.</li> <li>• EVPN control plane advertises the MAC or IP route to the BGP peer (in this case to the Spine), which in turn advertises it to all the NVE peers in the EVPN instance</li> </ul>
A new MAC learnt from one of the VLAN Sublfs	<p>Expectation: The learnt MAC is advertised to the BGP peers.</p> <ul style="list-style-type: none"> <li>• MAC is programmed in the local FDB of the virtual-network.</li> <li>• Virtual-network published the MAC.</li> <li>• EVPN control plane advertises the MAC or IP route to the BGP peer (in this case to the Spine), which in turn advertises it to all the NVE peers in the EVPN instance</li> </ul>
New interface added to the attached VLAN	<p>Expectation: MAC learning is enabled on the VLAN member interfaces. MACs learnt on these member interfaces are advertised to the BGP peers.</p> <ul style="list-style-type: none"> <li>• MACs learnt on the interface prior to the addition to VLAN are all pulled by virtual-network and are published one at a time.</li> <li>• EVPN control plane advertises the MAC or IP route to the BGP peer (in this case to the Spine), which in turn advertises it to all the NVE peers in the EVPN instance</li> </ul>
VLAN SubIntf operationally goes down or Interface is deleted from the VLAN	<p>Expectation: MACs learnt on this interface are withdrawn. BGP sends MAC withdraw notification to all the BGP peers.</p> <ul style="list-style-type: none"> <li>• MAC deletion on a virtual-network is published to EVPN control plane.</li> <li>• MACs are deleted from the local FDB.</li> <li>• EVPN control plane advertises the MAC or IP route to the BGP peer (in this case to the Spine), which in turn advertises it to all the NVE peers in the EVPN instance</li> </ul>
VLAN deleted from virtual-network	<p>Expectation: MACs learnt on all the VLAN member interfaces are withdrawn. BGP sends MAC withdraw notification to all the BGP peers.</p> <ul style="list-style-type: none"> <li>• Virtual-network publishes the MAC delete one-by-one to the EVPN control plane.</li> <li>• MACs are deleted from the local FDB.</li> <li>• EVPN control plane advertises the MAC or IP route to the BGP peer (in this case to the Spine), which in turn advertises it to all the NVE peers in the EVPN instance</li> </ul>
Virtual-network delete or VNI delete from virtual-network	<p>Expectation: All the Local MACs on the virtual-network are withdrawn. BGP sends MAC withdraw notification to all the BGP peers.</p>

**Table 14. Behavior of SmartFabric DellEMC OS10 for various events - Local MACs (continued)**

Local MACs Events	System behavior
	<ul style="list-style-type: none"> <li>• Virtual-network publishes the MAC delete one-by-one to the EVPN control plane.</li> <li>• MACs are deleted from the local FDB.</li> <li>• EVPN control plane advertises the MAC or IP route to the BGP peer (in this case to the Spine), which in turn advertises it to all the NVE peers in the EVPN instance</li> <li>• EVPN control plane is withdraw along with the inclusive multicast route (with PMSI) attribute from all the NVE Peers.</li> </ul>
Clear all local MACs	<p>Expectation: All the Local MACs on the virtual-network are withdrawn. BGP sends MAC withdraw notification to all the BGP peers.</p> <ul style="list-style-type: none"> <li>• Virtual-network flushes all MACs in the FDB.</li> <li>• Virtual-network publishes the MAC delete one-by-one to the EVPN control plane.</li> <li>• EVPN control plane is withdraw along with the inclusive multicast route (with PMSI) attribute from all the NVE Peers.</li> </ul>
Delete EVI or Delete VNI from EVI	<p>Expectation: All the Local MACs on the virtual-network are withdrawn. BGP sends MAC withdraw notification to all the BGP peers.</p> <ul style="list-style-type: none"> <li>• EVPN control plane advertises MAC or IP withdraw notification to all the NVE peers.</li> <li>• EVPN control plane deletes its MAC database.</li> <li>• EVPN control plane deletes all the remote endpoints from the corresponding virtual-network.</li> </ul>

The following table summarizes the behavior of SmartFabric DellEMC OS10 for various events that trigger the MAC processing:

**Table 15. Behavior of SmartFabric DellEMC OS10 for various events - Remote MACs**

Remote MACs Events	System behavior
MAC learnt from remote peer	<p>Expectation: Remote MACs learnt are programmed in the local FDB.</p> <ul style="list-style-type: none"> <li>• EVPN control plane populates the virtual-network with remote MAC.</li> <li>• Virtual-network programs the MAC in FDB.</li> <li>• Additionally, if this is the first ever MAC learnt from the remote peer, EVPN control plane configures this remote-endpoint in the virtual-network. The remote-mac learnt on the network port, from this remote-endpoint is disabled.</li> </ul>
MAC withdraw from the remote peer	<p>Expectation: Remote MAC is deleted from the Local FDB.</p> <ul style="list-style-type: none"> <li>• EVPN control plane deletes the MAC from virtual-network.</li> <li>• Virtual-network deletes the MAC entry from the FDB.</li> <li>• Additionally, if this is the last MAC being withdrawn from the remote peer, EVPN control plane deletes this remote-endpoint in the virtual-network.</li> </ul>
BGP peer down	<p>Expectation: All the remote MACs learnt from the peer are deleted from the local FDB.</p> <ul style="list-style-type: none"> <li>• EVPN control plane deletes all the MACs learnt from remote BGP peer from the virtual-network.</li> </ul>

**Table 15. Behavior of SmartFabric DellEMC OS10 for various events - Remote MACs (continued)**

Remote MACs Events	System behavior
	<ul style="list-style-type: none"> <li>Virtual-network deletes all the remote MAC entries from the FDB.</li> <li>EVPN control plane removes the remote-endpoints from the virtual-network after the last remote MAC is removed from the virtual-network.</li> </ul>
Clear all remote MACs	<p>Expectation: All the remote MACs learnt from the peer are deleted from the local FDB.</p> <ul style="list-style-type: none"> <li>Virtual-network deletes all the remote MACs from the FDB.</li> <li>Virtual-network publishes the MAC delete notification one-by-one to the EVPN control plane.</li> <li>EVPN control plane re-adds each remote MAC to virtual-network.</li> <li>Virtual-network programs the MAC in FDB.</li> </ul>
Virtual-network delete or VNI delete from virtual-network	<p>Expectation: All the Local and remote MACs learnt from the peer are deleted from the local FDB.</p> <ul style="list-style-type: none"> <li>Virtual-network publishes the VN delete.</li> <li>EVPN control plane deletes the EVPN bridge domain by withdrawing the inclusive multicast route (with PMSI) attribute from all the NVE peers.</li> <li>Virtual-network publishes the MAC delete notification one-by-one to the EVPN control plane.</li> <li>EVPN control plane deletes from its local database.</li> </ul>
Delete EVI or Delete VNI from EVI	<p>Expectation: All the remote MACs learnt from the peer are deleted from the local FDB.</p> <ul style="list-style-type: none"> <li>EVPN control plane deletes all the remote-MACs from the corresponding virtual-network.</li> <li>EVPN control plane deletes all the remote endpoints from the corresponding virtual-network.</li> </ul>

## MAC mobility

In a typical Data Center environment, VM's hosted in one server attached to a VTEP, can be moved to a different server attached to another VTEP, to facilitate the Workload mobility. This situation creates a unique problem of the same VM MAC being advertised by two VTEPs till the MAC entry is aged out from the VTEP, which is initially advertised. It also leads to traffic block-holing for a brief period of time. EVPN provides a mechanism to avoid this problem, by attaching a sequence number for each MAC advertised by a VTEP in the EVPN control plane. This sequence number is part of MAC mobility extended community, that is attached to each MAC advertisement.

When a VTEP learns a new MAC address from its attached hosts or servers and if that MAC address was previously advertised by another VTEP, it includes the MAC mobility extended community with an incremented sequence number in the MAC advertisement. After receiving this advertisement, the VTEP, which advertised the old MAC route, will initiate the MAC withdrawal, thus eliminating the traffic block-holing.

In this topology diagram, NVE-1, NVE-2, and NVE-3 are peering with the Spine switch. VTEP-1 learns the MAC and IP addresses of the connected Host H1, when H1 either sends an ARP request for remote Host or Gratuitous ARP. VTEP-1 also learns about Host1 being part of VNI-1 as it is connected by Port P1, VLAN V1. (P1,V1) is configured to be part of the bridge domain VNI-1. VTEP-1 advertises this information about H1 to all the remote VTEPs using MP-BGP EVPN control plane.

The following table captures the static MAC events and the SmartFabric DellEMC OS10 behavior:

**Table 16. Static MAC events and SmartFabric DellEMC OS10 behavior**

Static MAC event	SmartFabric DellEMC OS10 behavior
Static local MAC configured on a Virtual-network	<ul style="list-style-type: none"> <li>Virtual-network populates the static MAC in the local FDB.</li> </ul>

**Table 16. Static MAC events and SmartFabric DellEMC OS10 behavior (continued)**

Static MAC event	SmartFabric DellEMC OS10 behavior
	<ul style="list-style-type: none"><li>• Virtual-network publishes the MAC to EVPN control plane.</li></ul>
Received a Static MAC from the Remote NVE	<ul style="list-style-type: none"><li>• EVPN control plane adds the static MAC to the virtual-network.</li><li>• Virtual-network programs the MAC as static in its FDB.</li></ul>

## BUM traffic handling

This section describes BUM traffic handling.

EVPN provides mechanisms in which each PE or VTEP that is destined to receive BUM traffic, advertises an inclusive multicast ethernet tag EVPN route (Type-3) for a given EVPN bridge domain or EVI.

This behavior is accompanied by provider multicast service interface (PMSI) tunnel attribute, which specifies the Tunnel Type (for example, Ingress Replication), Encapsulation type (VXLAN), to be used to forward the traffic to the VTEP or PE, and the routable address of the PE or VTEP.

SmartFabric DellEMC OS10 supports ingress replication of the BUM traffic to all the remote VTEPs that advertised the inclusive multicast EVPN route in a given EVI.

## ARP suppression

This feature provides support to configure ARP-suppression on the switch.

Network Virtualization Overlay (NVO) is a solution in which an overlay network is used to extend L2 connectivity among VMs belonging to a tenant segment (or virtual network) over an underlay IP network.

This feature encapsulates the payload tenant within an IP packet at the originating end-point (ingress VTEP) and strips the encapsulated packets to access the payload at the destination end-point (egress VTEP). VXLAN is an example of NVO encapsulation.

Ethernet Virtual Private Network (EVPN) is a standards-based technology that is used to exchange control-plane information between the VTEPs. This control-plane information is exchanged using BGP, instead of manual configuration or flooding and learning in hardware. EVPN supports exchange of tenant IP-MAC binding between all VTEPs as part of its Type-2 route.

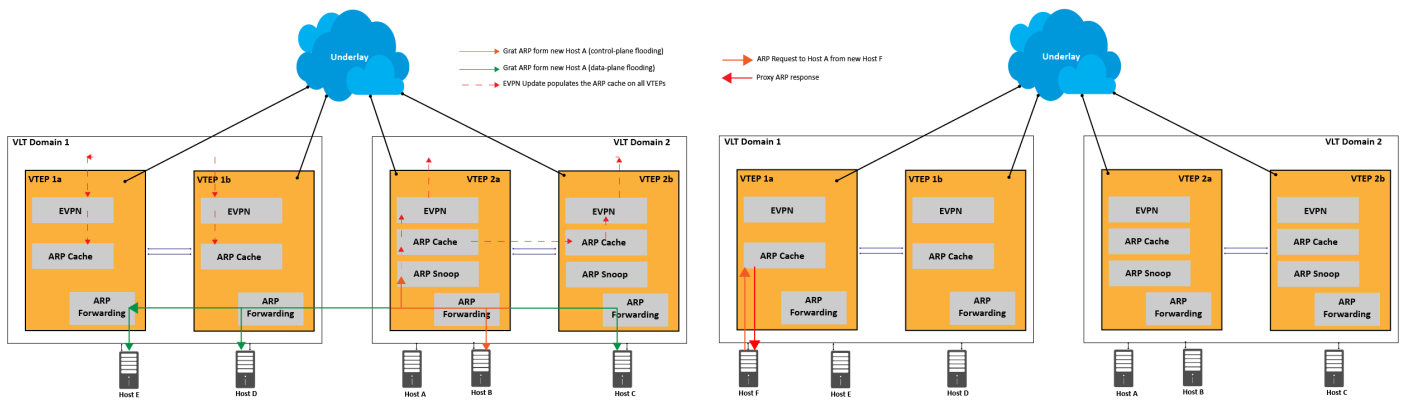
ARP suppression provides an option to minimize the flooding of tenant ARP or NS or NA packets in the underlay IP fabric to all the remote VTEPs saving both underlay bandwidth and CPU cycles on the end hosts. It requires each VTEP to maintain a cache of all the remote tenant IP-MAC bindings, so that when an ARP-request or NS is received for a remote tenant host within the tenant IP subnet (within virtual network), the ingress VTEP can retrieve the remote IP-MAC binding from its cache and responds on behalf of the remote host instead of flooding the ARP-requests or NS. This optimization is called ARP-suppression.

ARP flooding can occur for the initial ARP request to a silent host in the network. The VTEPs in the network do not see any traffic from the silent host until another host sends an ARP request for its IP address and an ARP response is sent back.

After the local VTEP learns about the MAC and IP addresses of the silent host, the information is distributed through BGP-EVPN control-plane to all other VTEPs. Any subsequent ARP requests do not must be flooded. Most end hosts send GARP, or RARP requests to announce themselves to the network immediately after they come online.

The local VTEP immediately has the opportunity to learn their MAC and IP addresses and distribute this information to other VTEPs through the BGP-EVPN control-plane. As a result, most active IP hosts in VXLAN EVPN must be learned by the VTEPs either through local learning or control-plane-based remote learning. So, ARP-suppression reduces the network flooding that is caused by host ARP learning behavior.

You can use the following figure to understand how OS10 learns host's MAC-IP by snooping the ARP or ND exchanges between hosts. Every VTEP learns the MAC-IP bindings of the hosts present in the local access ports and update the ARP-cache locally. It is exchanged to other VTEPs through BGP-EVPN. Only MAC-IP bindings corresponding to the local hosts are learned through packet snooping. MAC-IP bindings corresponding to the remote hosts are learned through BGP-EVPN.



Both the MAC-IP bindings are updated in the same ARP cache. After the VTEPs learn the MAC-IP bindings of both local and remote hosts, the VTEPs can avoid flooding the broadcast ARP-request or multicast NS received on any access port by ARP or NS proxy replying to the originator on behalf of the local or remote hosts. It helps reduce the flooding of ARP-request or NS to all other VTEPs, where the host is not present. It reduces network bandwidth utilization and CPU cycles of actual end host and other hosts, which unnecessarily process and ignore the transient ARP-request or NS.

In the Figure, gratuitous-ARP is taken as an example to show how ARP-snooping module learns the host's MAC-IP bindings. Also, the snooper module learns through other packets (ARP-request or reply and NA).

## Restrictions and limitations

The following restriction and limitation applies when you configure ARP suppression:

### S4200 Qumran (Dune) limitation:

- ARP-suppression feature is not supported on the S4200 platform.

### Impact on software upgrade or downgrade

There is no impact on software upgrades or downgrades because of the ARP-suppression feature.

ARP-suppression feature an optimization feature that helps in reducing the ARP or NS packets flooding in the VXLAN network. So, it can work with other VTEPs that runs old software versions, which do not support ARP-suppression.

ARP Suppression feature is enabled by default when EVPN configuration is present.

You can upgrade the VLT-nodes with ARP-suppression supported software one after the other without any impact. Until the other VLT-node is upgraded, the peer VLT-node snoops the ARP or ND packets and performs proxy-reply for ARP-req or NS packets received on that node. After other the VLT-node is upgraded, VLT-sync synchronizes the snooped MAC-IP binding of local hosts.

## Configuration notes

- The ARP-suppression feature is enabled by default, after BGP-EVPN is enabled in SmartFabric OS10.
- ARP-suppression is supported only on VxLAN bridges (Virtual network interfaces) and is not supported on legacy VLAN bridges.
- ARP-suppression is supported on both Layer 2 and Layer 3 VxLAN bridges.
- ARP-suppression is supported on both asymmetric and symmetric BGP-EVPN modes.
- Disable ARP-suppression globally using the following command:

```
arp-nd-suppression disable
```

Reenable ARP-suppression using the following command:

```
no arp-nd-suppression disable
```

- Reenable ARP-suppression using the following command:

```
no arp-nd-suppression disable
```

## VLT functionality

- VLT-sync for L2-VxLAN bridges is enabled in OS10, so that snooped ARP-entries are synchronized between VLT-nodes.
- Proxy-replies to the ARP-requests sent by the local-host are replied only by the first VLT-node.
- No proxy-replies for the ARP-requests are received on the VLTi link. These proxy-replies are flooded to other virtual network interfaces by following VLT and VXLAN split-horizon rules.
- If there is a VLTi link failure, there is no change in the existing behavior.
- After clearing ARP or IPv6 neighbor entries in VLT peers, learning of ARP or IPv6 neighbors through ARP request or neighbor solicitation frames from host to gateway (virtual IP) happens only on VLT peer which receives the frame. Other VLT peer learns the ARP or IPv6 neighbor once traffic hits the other node or the ARP/IPv6 neighbor resolution packets hashes to other peer.

## BGP EVPN with VLT

OS10 supports BGP EVPN operation between VLT peers that you configure as VTEPs. For more information about configurations and best practices to set up VLT for VXLAN, see [Configure VXLAN — Configure VLT](#). This information also applies to BGP EVPN for VXLAN.

Dell Technologies recommends configuring iBGP peering for the IPv4 address family between the VTEPs in a VLT pair on a dedicated L3 VLAN that is used when connectivity to the underlay L3 network is lost. It is NOT required to enable the EVPN address family on the iBGP peering session between the VTEPs in a VLT pair because EVPN peering to the spine switch is performed on Loopback interfaces.

Both VTEPs in a VLT pair advertise identical EVPN routes, which provides redundancy if one of the VTEP peers fails. To set up redundant EVPN route advertisement, configure the same EVI, RD, and RT values for each VNI on both VTEPs in a VLT pair, including:

- In auto-EVI mode, this identical configuration is automatically ensured if the VNID-to-VNI association is the same on both VTEP peers.
- In manual EVI mode, you must configure the same EVI-to-VNID association on both VTEP peers.
- In manual EVI mode, you must configure the same RD and RT values on both VTEP peers.

In an EVPN configuration, increase the VLT delay-restore timer to allow for BGP EVPN adjacency to establish and for the remote MAC and neighbor entries to download by EVPN and install in the dataplane. The VLT delay-restore determines the amount of time the VLT Port channels are kept operationally down at bootup to allow the dataplane to set up and forward traffic, resulting in minimal traffic loss as the VLT peer node boots up and joins the VLT domain.

 **NOTE:** The network links of the NVE or VTEP VLT pair cannot be VLT-Port channels.

For a sample BGP EVPN VLT configuration, see [Example: VXLAN with BGP EVPN](#).

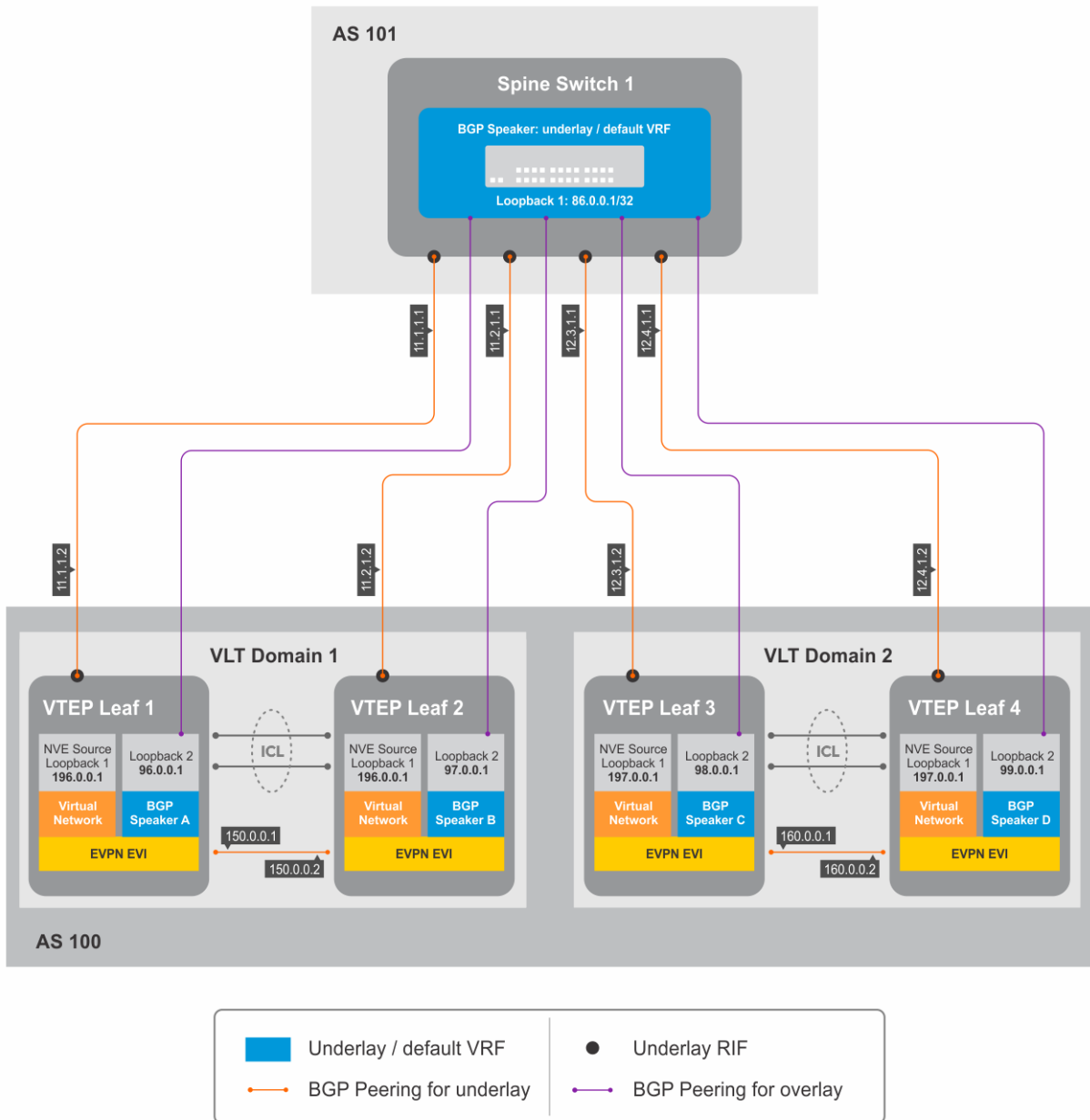


Figure 4. BGP EVPN in VLT domain

## Setting up eBGP peering with VLT

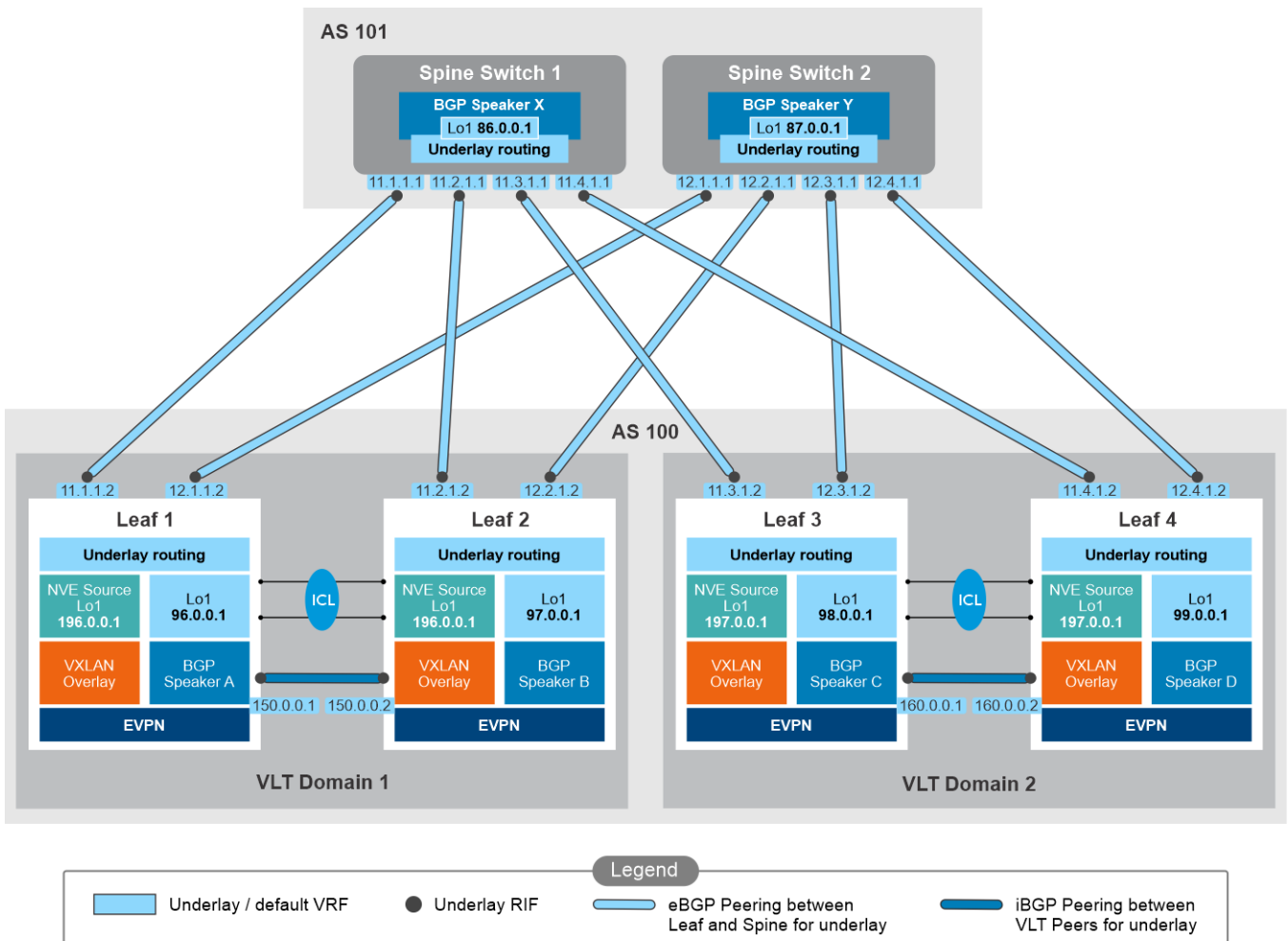
This section describes how eBGP peering is setup for underlay IP route advertisement and for overlay EVPN route advertisement.

### BGP peering for underlay IP routes

eBGP peering session between Leaf and Spine using direct interface IP address.

iBGP session between VLT peers on dedicated VLAN configured over VLTi.

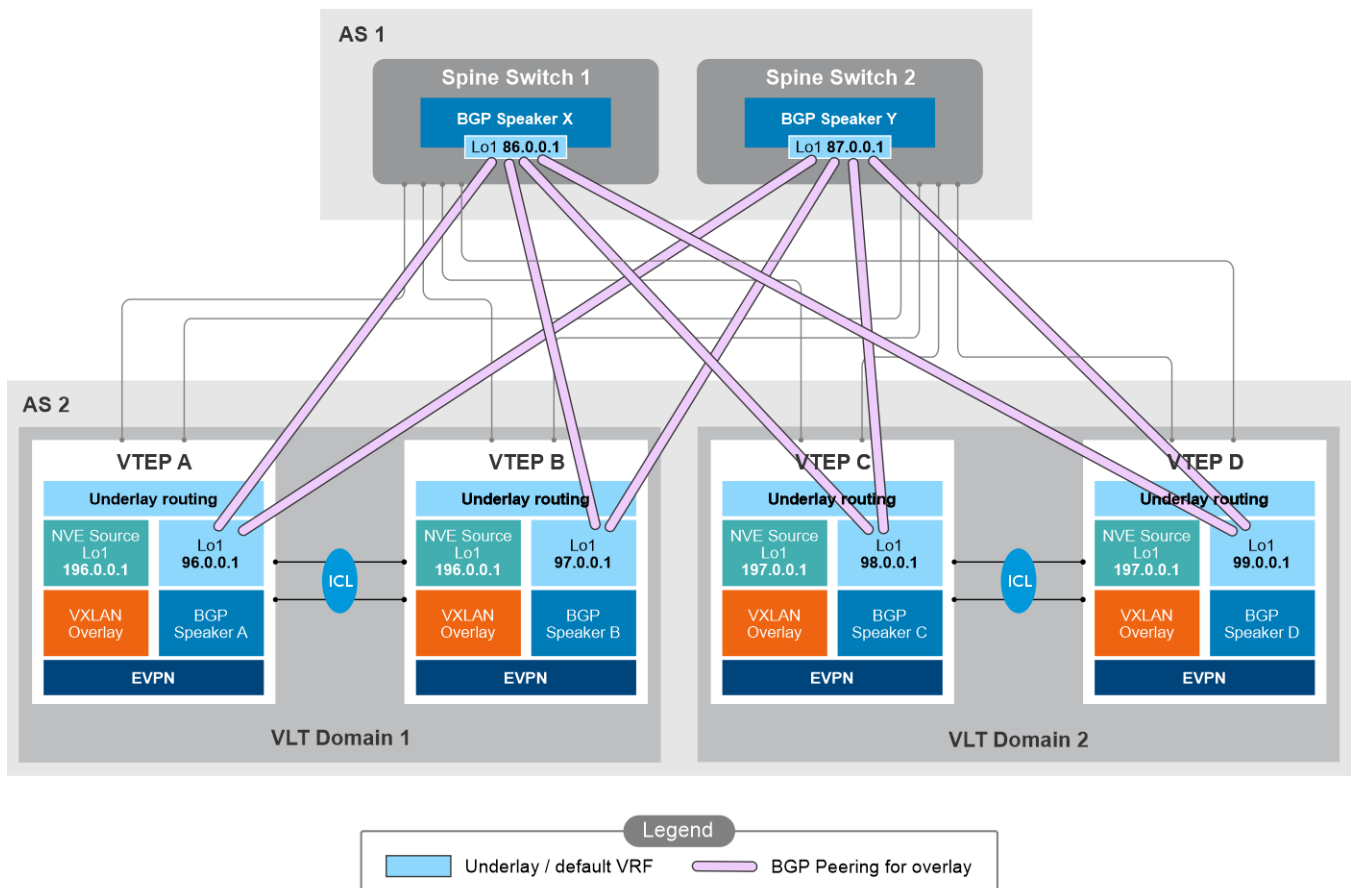
Following illustration shows setting up of eBGP peering with VLT:



## BGP peering for Overlay services

This section describes eBGP Peering between loopback IP addresses between Leaf and Spine.

The following illustration depicts BGP peering for overlay services:



Each BGP router has two separate BGP sessions between each Leaf and Spine - one for underlay routes and one for overlay routes (Virtual Network MACs).

The following sequence describes the topology depicted in the BGP peering for overlay services image:

1. This topology requires two separate loopback interfaces.
2. BGP underlay session - IPv4 address family - used to learn underlay IP route reachability to remote VTEPs.
  - a. Uses the point-to-point link IP address as the BGP neighbor IP address.
  - b. IP reachability to both loopback IP addresses are advertised with the point-to-point IP as Next-hop.
  - c. In case of failure of all network links on one of the VLT nodes, BGP sessions between VLT peers are needed for IPv4 address family to retain underlay routes.
3. BGP Overlay Session - EVPN address family - used to exchange overlay Layer 2 information between the VTEP nodes (Leaf nodes).
  - a. Uses a unique loopback interface IP on each node (Leaf and Spine) as the BGP neighbor IP address.

- b. An explicit override command (`update-source`) is needed to ensure that the local IP address that is used in the TCP session is the loopback IP address instead of the link IP address. This behavior ensures that the session is retained even if the links fail.
  - c. At the VTEPs, irrespective of the `update-source` override, the EVPN routes are always advertised with the NVE source IP address (Lo1) as the Next-Hop.
  - d. eBGP next-hop modification is disabled by default on all EVPN routes. So, the Spine nodes do not use their own Next-hop when advertising the EVPN routes to the remote VTEPs.
  - e. There is no EVPN address family peering between the VLT peer nodes within the same VLT domain.
4. By default, SmartFabric DellEMC OS10 does not override the Next-hop when advertising EVPN address family routes to even eBGP peers. So, EVPN routes received by Leaf nodes contain the originating Leaf node's IP address in the Next-Hop.
  5. If all Leaf nodes are in the same AS, then the Spine nodes need to be explicitly configured to disable sender-side-loop-detection. Without this configuration, the eBGP on Spine does not distribute routes received from VTEP A (AS2) back to VTEP C (also AS2).
    - NOTE:** You must enable loop detection on the Leaf nodes (default). This configuration ensures that the Leaf VTEP nodes do not distribute the routes back to other Spine nodes again.
  6. You can use any IP address that is unique to a node as the BGP Router ID. Typically, a separate loopback interface with static IP address is used for this purpose to avoid allocating this IP address for other purposes.
  7. Even when a VLT node loses all its direct network links to the fabric, it still has reachability to its previous underlay First-hops (Spine or RR) over the VLTi with the VLT peer node as the new underlay First-hop.
    - a. A VLT node's BGP EVPN sessions on loopback interfaces do not terminate even if it loses all direct connectivity to the underlay network.
    - b. This behavior requires the BGP multi-hop parameter to be configured as 3.
- NOTE:** EVPN user configurations (for example, Auto-EVI or manual-EVI, RT and RD) are not synchronized between the VLT peers. Inline with existing VLT support, the expectation is that you configure both VLT nodes identically. Any mismatches are shown in the `show vlt mismatch` command.

The following sequence depicts the setup for common BGP session for underlay and overlay:

1. The same BGP session advertises both IP routes and EVPN address family routes.
2. The Network link IP address on each node is used as the BGP neighbor IP address.
3. A loopback interface IP address is needed on each node to act as NVE source IP address.
4. IP reachability to the loopback IP address is advertised with the link IP address as Next-hop.
5. At the VTEP, EVPN routes are always advertised with the NVE source IP address as Next-hop.
6. The Spine is internally setup to not override the EVPN address family route next-hops with its own IP address.
7. Any IP address that is unique to a node can be used as the BGP Router ID. Typically, a separate loopback interface with static IP address is used for this purpose to avoid allocating this IP address for other purposes.

## EVPN local route advertisement

BGP EVPN running on each VTEP listens to local overlay information, encodes them as BGP EVPN routes, and injects them into BGP to be advertised to remote VTEPs. This section describes EVPN local route advertisement functionality when the VTEPs are set up as VLT pairs.

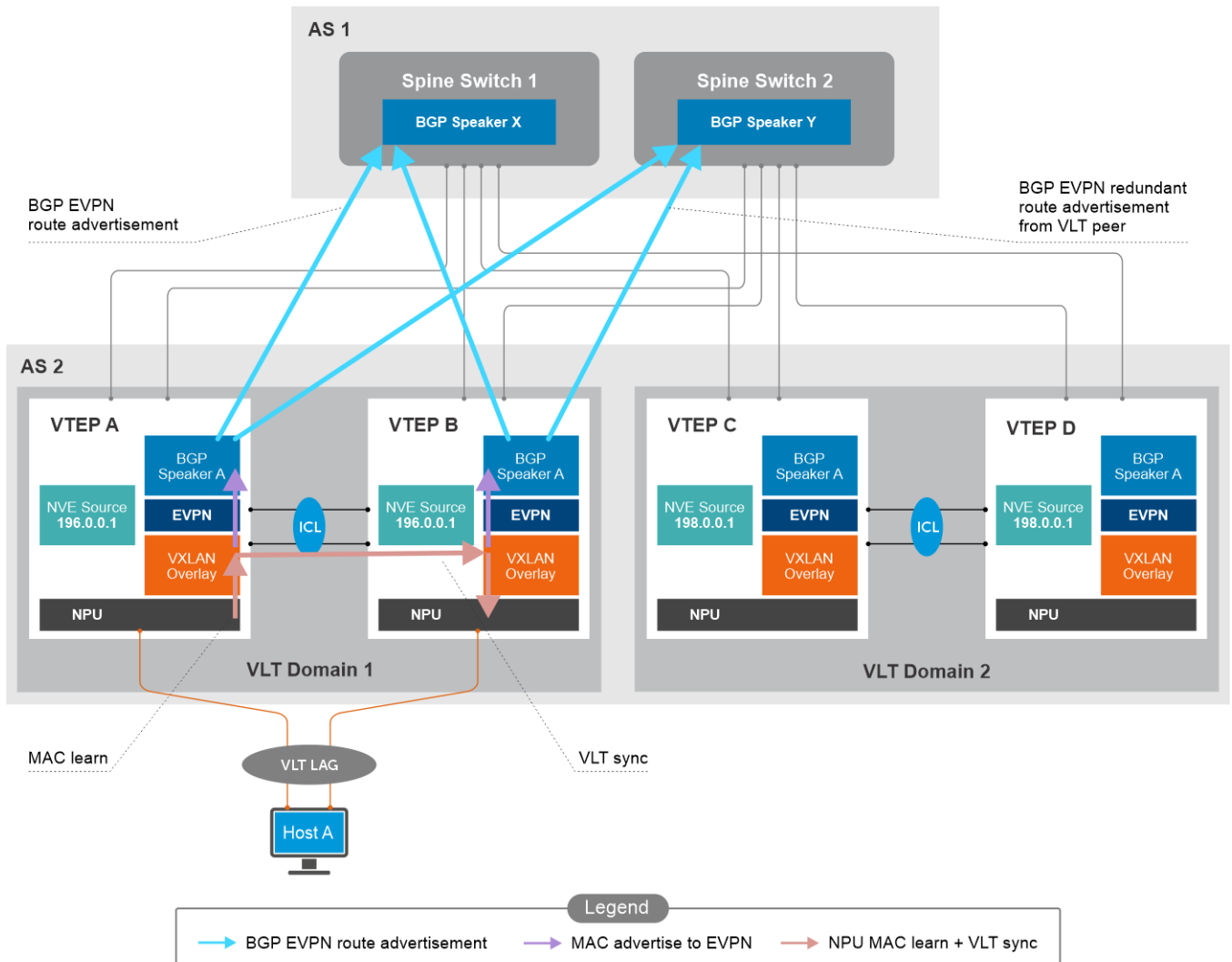
### Overlay access MAC learn or age or flush or clear

For all MAC learn or age or flush or clear operations on local ports in the overlay, there is no change in the regular VLT functionality as defined for static Virtual-networks.

Meaning, when VLT node A learns a MAC address in the virtual-network from the data-plane on its access port, it synchronizes this MAC address with VLT to its VLT peer node B.

Similarly, in case of MAC age or clear or flush operations for MACs on VLT ports, the MAC is removed from both VLT peers only if it has aged on both nodes. Also, for MACs on orphan ports the MAC is immediately removed on both VLT peers.

The following topology diagram depicts the overlay access MAC learn or age or clear process:



## Identical EVPN routes by both VLT peer nodes

Since each VLT node is a separate BGP router, both VLT nodes individually advertise EVPN route for a MAC irrespective of whether it is locally learnt or is synchronized by VLT from the peer. However, since the remote VTEPs see the VLT pair as a single logical VTEP, the remote VTEPs need to see only one copy of this route.

This behavior is achieved by ensuring that the EVPN routes advertised by both VLT peer nodes have the same NLRI and Next-hop. The Next-hop is always same, since it is the common NVE source IP address that is shared by both VLT peers.

The NLRI varies according to Route type.

### Route Type 2 - MAC or IP

- Advertised for each MAC learnt in each Virtual-network (EVI).
- The NLRI consists of RD, MAC, IP, Ethernet tag ID, or MPLS Label1 (VNI). Ethernet tag ID is always 0 in this case (1-1 VNI to EVI).
- Except RD, all the other parameters advertised by both VLT peers are the same.

### Route Type 3 - Inclusive multicast route

- Advertised for each Virtual-network (EVI) that has a VNI associated with it.
- The NLRI consists of RD, Ethernet tag ID, or IP of VTEP. Ethernet tag ID is always 0 in this case (1-1 VNI to EVI).
- Except RD, all the other parameters advertised by both VLT peers are the same.

RD is the Route distinguisher that is used to ensure that BGP maintains identical route information belonging to different nodes that are separate from each other.

For EVPN we use Type 1 RD, which comprises of IP address to identify the VTEP node. EVI to identify each Virtual-network or VNI.

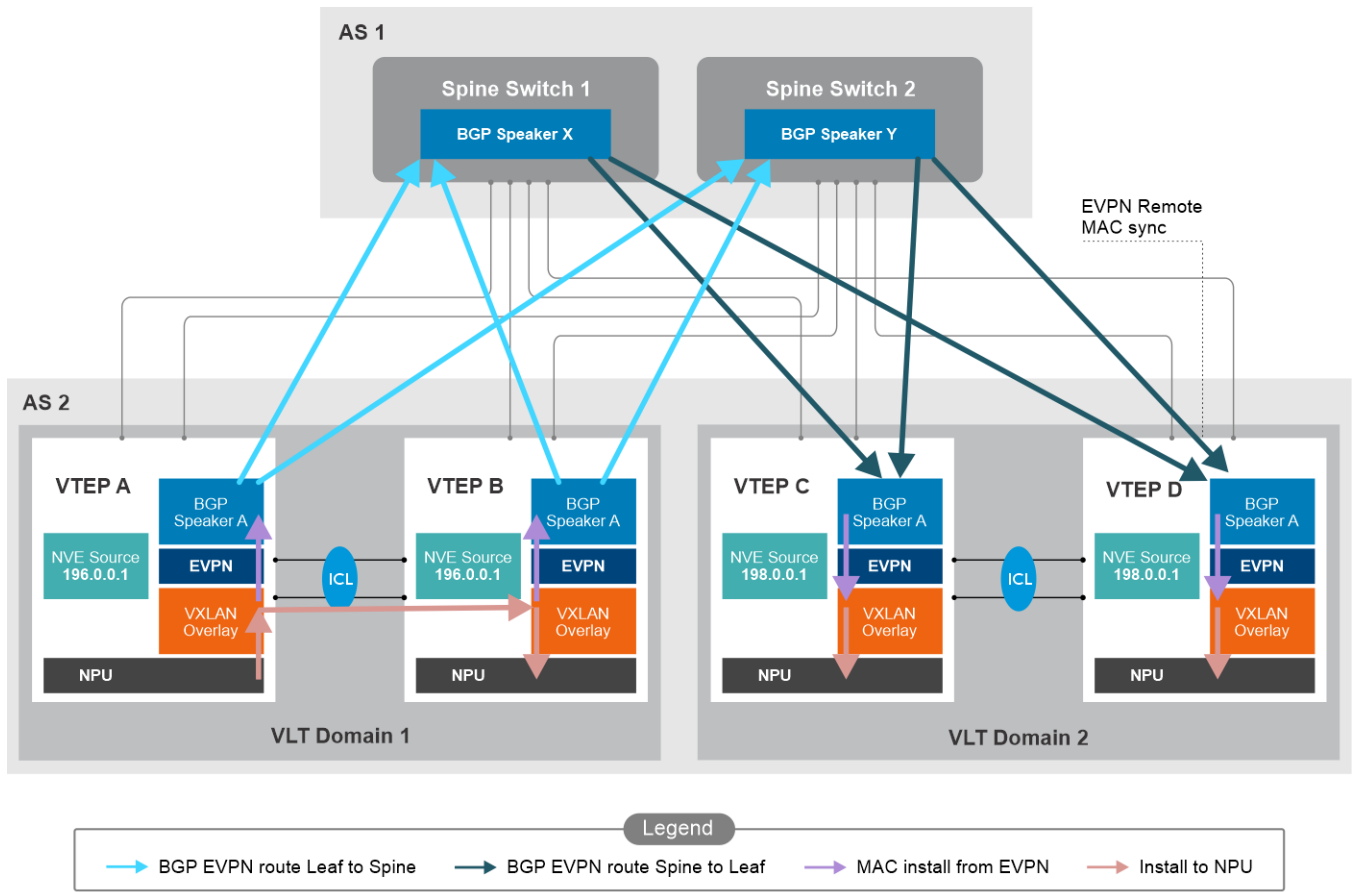
In order to have identical RDs between VLT peers in all route types:

- The common NVE source IP is used as the IP address in RD.
- Auto-EVI mode -
  - The Virtual-network ID is used as the EVI.
  - The Virtual-network ID is explicitly configured by user on both VLT peers and is already mandated to be identical for a given VNI.
- Manual-EVI mode -
  - If the RD is auto, then the EVI to VNI mapping that you configure needs to be identical on both VLT peers.
  - If the RD is manual, then the RD to VNI mapping that you configure needs to be identical on both VLT peers.

## EVPN remote route install

BGP EVPN running on each VTEP receives BGP EVPN routes from BGP, decodes the overlay information received from remote VTEPs in these routes, and configures the data-plane based on the decoded information. This section describes the functionality when the VTEPs are set up as VLT pairs.

The following topology diagram depicts the EVPN remote route install functionality:



## Overlay remote MAC learn or remove

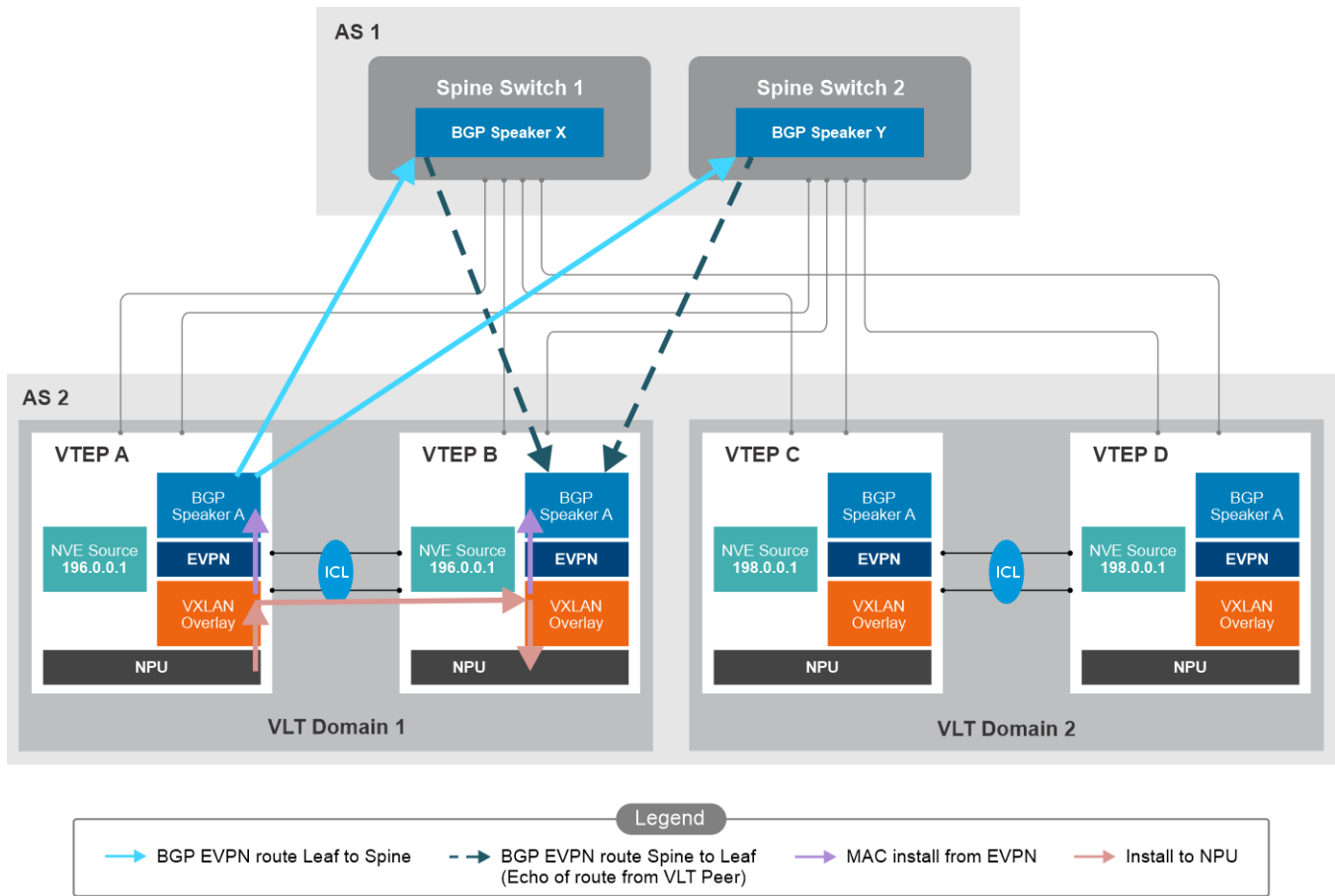
Remote MACs (MACs of hosts sitting behind remote VTEPs) that are installed by EVPN are not VLT synced to the peer since BGP-EVPN running on each VLT node will install these MACs independently.

Similarly withdrawal of Remote MACs is also not synced to the VLT peer.

## MAC clear

The `mac clear` command issued on one of the VLT nodes is synced to the VLT peer node and all MACs including remote MACs are removed from data-plane - this behavior is the regular VLT behavior.

However, remote MACs installed by EVPN are considered sticky and are re-installed back immediately in the data-plane on both VLT peers.



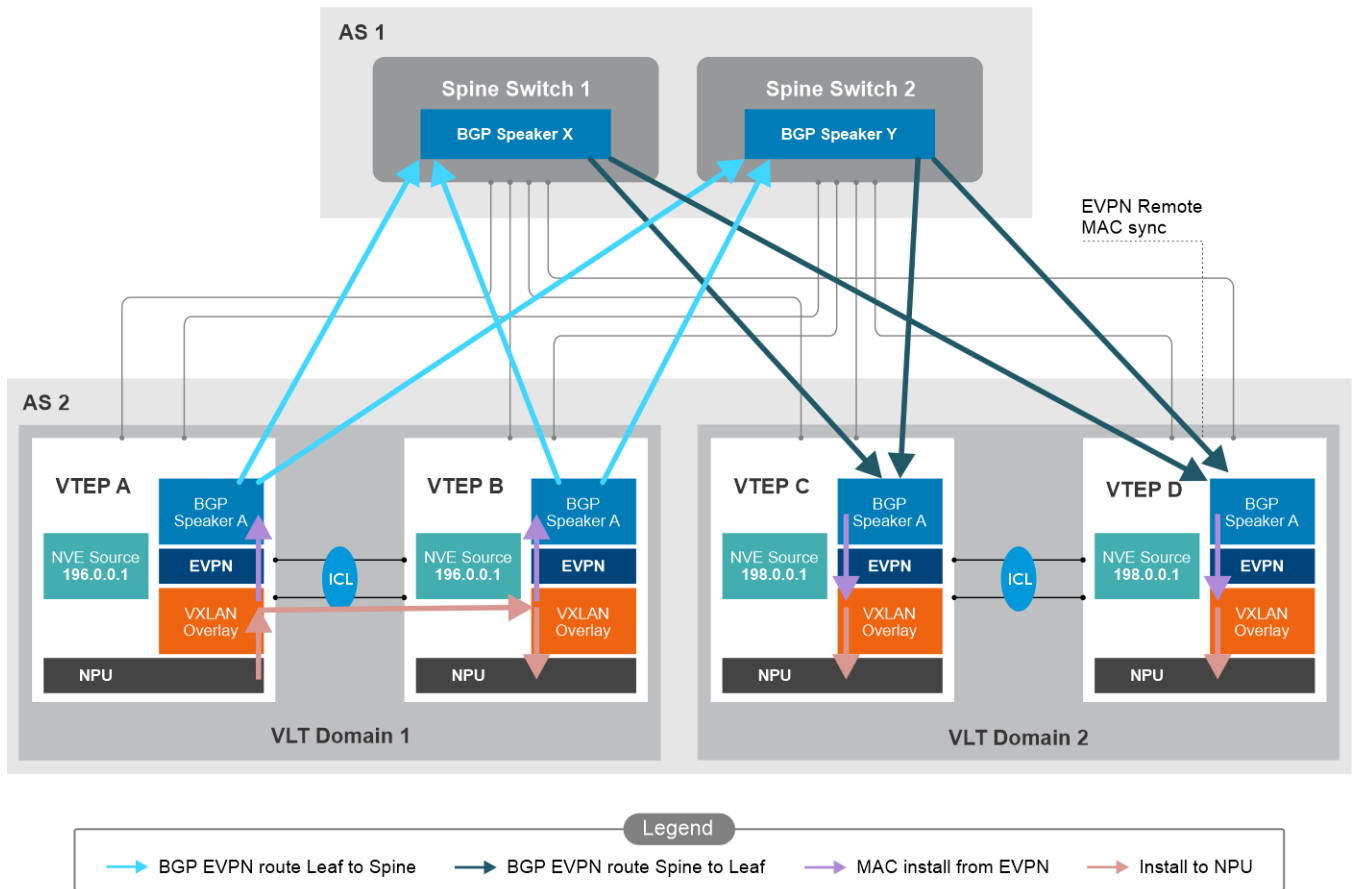
All EVPN routes advertised by a VLT node corresponding to its local EVIs or MAC addresses are also advertised by the BGP on VLT Peer. These BGP EVPN routes are echo-received back on the peer VLT nodes as remote routes. Since the EVPN routes advertised by both VLT nodes are identical for a given EVI or MAC address, BGP always prefers the local route and will not download the duplicate route received from the VLT peer to EVPN.

In case of race conditions, the EVPN echo route from VLT peer can be received before the local route is available. These peer EVPN routes are differentiated using their Next-hop VTEP IP, which is the same as their own local NVE source IP and ignored by the EVPN layer.

## EVPN MAC move

When MAC moves from remote VTEP to local VTEP, the MAC is learnt on one of the Leaf nodes and is synced with VLT to the VLT peer node. This MAC is advertised to the EVPN and BGP modules. The MAC learn gets advertised as identical BGP-EVPN route updates from both VLT peer leaf nodes.

When MAC moves from local VTEP to remote VTEP or from one remote VTEP to another, both VLT leaf pairs independently receive BGP-EVPN route updates and install the MAC with the new destination in the hardware.



## VLT events

This section describes the impact of VLT events on BGP EVPN control plane.

### VLT local port failure

No change in BGP EVPN. The MAC addresses learnt on these ports are re-directed by VLT logic and continue to be advertised in EVPN.

When VLT port fails on both VLT nodes, then the MAC is removed from both nodes and is withdrawn by both BGP speakers. This causes the MAC to be removed from all remote VTEPs.

### VLT single node failure

Since each VLT node is an independent BGP speaker there is no special handling needed in BGP EVPN. The local VLT MACs continue to be advertised by EVPN on the VLT peer.

All orphan port MACs on the failed peer that were installed on the VLTi locally, are removed by VLT logic; as a result, these MAC EVPN routes are withdrawn by the remaining VLT peer nodes as well.

#### **VLT VLTi failure**

When the Heartbeat is up (nodes are still alive), all VLT local ports and network ports are forced down on the VLT secondary node using UFD. This behavior causes all BGP sessions to go down and all remote MACs to be removed from the VLT secondary node. Effectively, the VLT secondary node is isolated from the network.

Orphan local MACs on the VLT secondary node are withdrawn from all the remote VTEPs.

VLT local MACs continue to be advertised by the VLT Primary node; as a result, they are retained in the remote VTEPs.

#### **Network port failure**

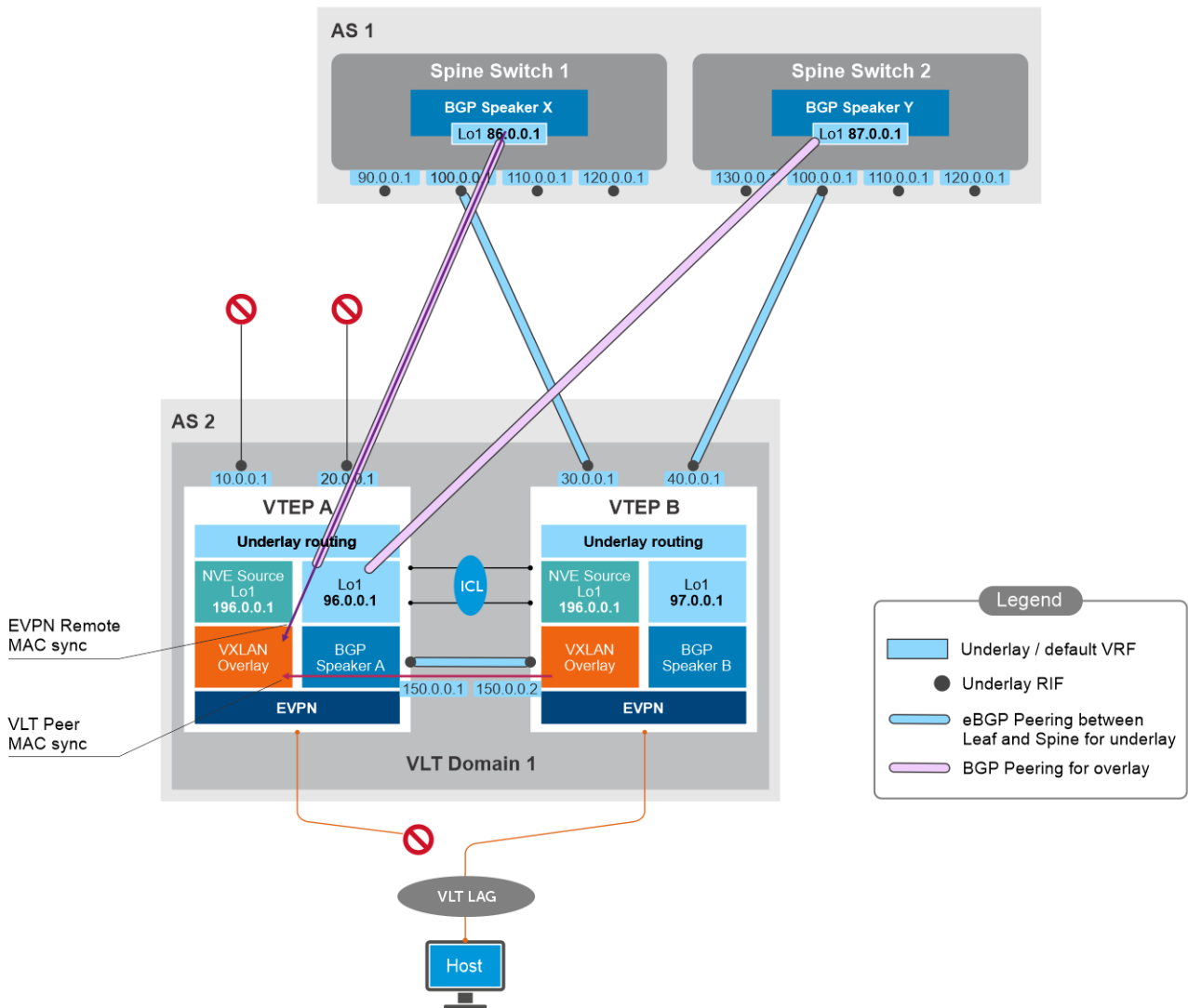
As long as the VTEP has connectivity to at least one Spine node, all the BGP routes are intact and there is no change.

When a VLT node has lost connectivity to all Spine nodes, the behavior differs depending on the deployment scenario:

- Separate overlay BGP session on separate loopback interface:
  - Leaf and Spine loopback IP addresses should still be reachable from each other in the underlay based on underlay routing re-convergence using the VLTi with the VLT peer as the first-hop.
  - In this case, the BGP sessions between Leaf and Spine nodes are still active.
  - Encapsulated data traffic from Leaf to Spine node is also redirected through VLTi once the underlay routing re-converges.
- Combined underlay and overlay BGP sessions directly on the network link:
  - BGP sessions between Leaf and Spine nodes go down. All EVPN remote MACs or ARPs are removed from the data-plane.
  - Overlay Layer 2 switched data traffic from Leaf to Spine node is flooded to all remote VTEPs. The encapsulated traffic is redirected through VLTi after the underlay routing re-converges.
  - Overlay Layer 3 routed data traffic from Leaf to Spine node is dropped.

#### **VLT single node recovery**

The following topology diagram depicts the VLT single node recovery scenario:



When a VLT node boots up, if the VLT role is detected as secondary, all the server facing VLT ports and all the network facing ports are held in down state for a specified interval. This phase is called the delay-restore phase.

Before the ports come up, the following are expected to be completed:

- The overlay BGP sessions are established ; this establishment happens between Leaf and Spine loopback IP addresses over VLTi with the VLT primary acting as transit hop.
- Overlay EVPN routes are received from remote VTEPs and all remote MACs and ARPs are installed in the data-plane.
- Underlay route reachability to remote VTEPs is installed in the data-plane through VLTi with primary acting as transit hop.

The VLT delay restore timer needs to be adjusted to accommodate these changes.

## VLT with IRB

This section describes VLT with IRB.

### Local ARP

Just as in the case of MAC, any ARP learnt from a locally attached host is VLT synced to the peer node. Both VLT nodes advertise identical MAC-IP EVPN routes so that the Spine nodes sees multiple redundant routes and does not withdraw the route even if one of the VLT peers fail.

## Remote ARP

Just as in the case of MAC, any ARP for hosts connected to remote VTEPs is not VLT synced to the peer node.

## ARP clear

Unlike MAC, ARP clear command executed on one of the VLT nodes is not synced to the VLT peer node. As a result, these ARPs are withdrawn from the remote VTEPs unless the ARP is cleared on both VLT Peers.

Immediately after ARP clears, all ARPs for hosts connected to remote VTEPs is installed back into the data-plane.

## VLT events

VLT failure events like link failure node, failure, or VLTi failure have no impact on the ARPs advertised by EVPN routes.

When a VLT node fails and recovers, all remote MAC-IP routes need to be received and processed before the VLT delay-restore time is complete.

## Configuration sequence

### Create Virtual-network

```
OS10(config)# virtual-network 10000
OS10(config-vn)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn)# vxlan-vni 10000
OS10(config-vn)# exit
OS10(config)# virtual-network 20000
OS10(config-vn)# member-interface ethernet 1/1/6 untagged
OS10(config-vn)#vxlan-vni 20000
OS10(config-vn)# exit
```

### Create Virtual-network interface


```
OS10(config)# interface virtual-network10000
OS10(conf-if-vn-10000)# ip vrf forwarding VRF-TENANT-1
OS10(conf-if-vn-10000)# ip address 10.10.10.1/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.10.10.10
OS10(conf-if-vn-10000)# no shutdown
OS10(config)# interface virtual-network20000
OS10(conf-if-vn-20000)# ip vrf forwarding VRF-TENANT-1
OS10(conf-if-vn-20000)# ip address 20.20.20.1/16
OS10(conf-if-vn-10000)# ip virtual-router address 20.20.20.20
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-if-vn-20000)# exit
```

### Create anycast gateway MAC address

```
OS10(config)# ip virtual-router mac-address 00:00:01:00:00:01
```

### Enable EVPN and EVI

```
OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# rd 110.111.170.195:10000
OS10(config-evpn-evi-10000)# route-target auto both
OS10(config-evpn-evi-10000)# vni 10000
OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# rd 110.111.170.195:20000
OS10(config-evpn-evi-20000)# route-target auto both
OS10(config-evpn-evi-20000)# vni 20000
```

 **NOTE:** The **auto-evi** option can be enabled instead of manual EVI configuration.

## Verification

```
OS10# show evpn evi 10000
EVI : 10000, State : up
  Bridge-Domain      : Virtual-Network 10000, VNI 10000
  Route-Distinguisher : 1:110.111.170.195:10000 (auto)
  Route-Targets       : 0:10000:16787216 (auto) both
  Inclusive Multicast : 110.111.170.107
  IRB                 : Enabled (VRF: VRF-TENANT-1)

OS10# show evpn evi 20000
EVI : 20000, State : up
  Bridge-Domain      : Virtual-Network 20000, VNI 20000
  Route-Distinguisher : 1:110.111.170.195:20000 (auto)
  Route-Targets       : 0:20000:16797216 (auto) both
  Inclusive Multicast : 110.111.170.107
  IRB                 : Enabled (VRF: VRF-TENANT-1)

OS10# show evpn vrf VRF-TENANT-1
EVI      VXLAN-VNI      Virtual-Network
10000    10000             10000
20000    20000             20000

OS10# show evpn mac-ip
Type -(lcl): Local (rmt): remote
EVI      Mac Address      Type      Seq No  Host-IP      Interface/Next-Hops
10000    00:00:0b:0b:0b:0a lcl       0       10.10.10.10  ethernet1/1/6
10000    14:18:77:25:4e:82 rmt       0       10.10.10.11  110.111.170.107
```

## ARP suppression

This feature provides support to configure ARP-suppression on the switch.

Network Virtualization Overlay (NVO) is a solution in which an overlay network is used to extend L2 connectivity among VMs belonging to a tenant segment (or virtual network) over an underlay IP network.

This feature encapsulates the payload tenant within an IP packet at the originating end-point (ingress VTEP) and strips the encapsulated packets to access the payload at the destination end-point (egress VTEP). VXLAN is an example of NVO encapsulation.

Ethernet Virtual Private Network (EVPN) is a standards-based technology that is used to exchange control-plane information between the VTEPs. This control-plane information is exchanged using BGP, instead of manual configuration or flooding and learning in hardware. EVPN supports exchange of tenant IP-MAC binding between all VTEPs as part of its Type-2 route.

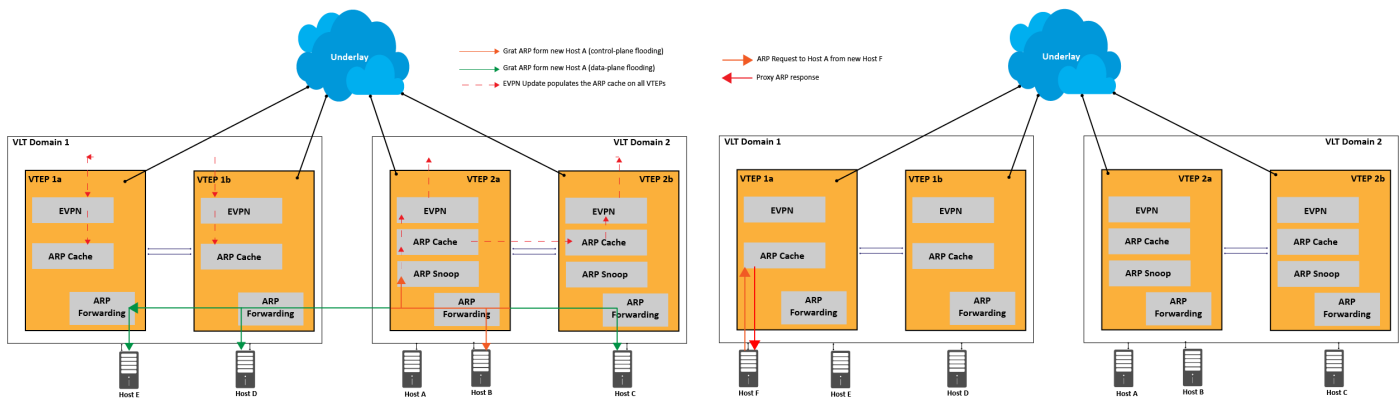
ARP suppression provides an option to minimize the flooding of tenant ARP or NS or NA packets in the underlay IP fabric to all the remote VTEPs saving both underlay bandwidth and CPU cycles on the end hosts. It requires each VTEP to maintain a cache of all the remote tenant IP-MAC bindings, so that when an ARP-request or NS is received for a remote tenant host within the tenant IP subnet (within virtual network), the ingress VTEP can retrieve the remote IP-MAC binding from its cache and responds on behalf of the remote host instead of flooding the ARP-requests or NS. This optimization is called ARP-suppression.

ARP flooding can occur for the initial ARP request to a silent host in the network. The VTEPs in the network do not see any traffic from the silent host until another host sends an ARP request for its IP address and an ARP response is sent back.

After the local VTEP learns about the MAC and IP addresses of the silent host, the information is distributed through BGP-EVPN control-plane to all other VTEPs. Any subsequent ARP requests do not must be flooded. Most end hosts send GARP, or RARP requests to announce themselves to the network immediately after they come online.

The local VTEP immediately has the opportunity to learn their MAC and IP addresses and distribute this information to other VTEPs through the BGP-EVPN control-plane. As a result, most active IP hosts in VXLAN EVPN must be learned by the VTEPs either through local learning or control-plane-based remote learning. So, ARP-suppression reduces the network flooding that is caused by host ARP learning behavior.

You can use the following figure to understand how OS10 learns host's MAC-IP by snooping the ARP or ND exchanges between hosts. Every VTEP learns the MAC-IP bindings of the hosts present in the local access ports and update the ARP-cache locally. It is exchanged to other VTEPs through BGP-EVPN. Only MAC-IP bindings corresponding to the local hosts are learned through packet snooping. MAC-IP bindings corresponding to the remote hosts are learned through BGP-EVPN.



Both the MAC-IP bindings are updated in the same ARP cache. After the VTEPs learn the MAC-IP bindings of both local and remote hosts, the VTEPs can avoid flooding the broadcast ARP-request or multicast NS received on any access port by ARP or NS proxy replying to the originator on behalf of the local or remote hosts. It helps reduce the flooding of ARP-request or NS to all other VTEPs, where the host is not present. It reduces network bandwidth utilization and CPU cycles of actual end host and other hosts, which unnecessarily process and ignore the transient ARP-request or NS.

In the Figure, gratuitous-ARP is taken as an example to show how ARP-snooping module learns the host's MAC-IP bindings. Also, the snooper module learns through other packets (ARP-request or reply and NA).

## Restrictions and limitations

The following restriction and limitation applies when you configure ARP suppression:

### S4200 Qumran (Dune) limitation:

- ARP-suppression feature is not supported on the S4200 platform.

## Impact on software upgrade or downgrade

There is no impact on software upgrades or downgrades because of the ARP-suppression feature.

ARP-suppression feature an optimization feature that helps in reducing the ARP or NS packets flooding in the VXLAN network. So, it can work with other VTEPs that runs old software versions, which do not support ARP-suppression.

ARP Suppression feature is enabled by default when EVPN configuration is present.

You can upgrade the VLT-nodes with ARP-suppression supported software one after the other without any impact. Until the other VLT-node is upgraded, the peer VLT-node snoops the ARP or ND packets and performs proxy-reply for ARP-req or NS packets received on that node. After other the VLT-node is upgraded, VLT-sync synchronizes the snooped MAC-IP binding of local hosts.

## Configuration notes

- The ARP-suppression feature is enabled by default, after BGP-EVPN is enabled in SmartFabric OS10.
- ARP-suppression is supported only on VxLAN bridges (Virtual network interfaces) and is not supported on legacy VLAN bridges.
- ARP-suppression is supported on both Layer 2 and Layer 3 VxLAN bridges.
- ARP-suppression is supported on both asymmetric and symmetric BGP-EVPN modes.
- Disable ARP-suppression globally using the following command:

```
arp-nd-suppression disable
```

Reenable ARP-suppression using the following command:

```
no arp-nd-suppression disable
```

- Reenable ARP-suppression using the following command:

```
no arp-nd-suppression disable
```

## VLT functionality

- VLT-sync for L2-VxLAN bridges is enabled in OS10, so that snooped ARP-entries are synchronized between VLT-nodes.
- Proxy-replies to the ARP-requests sent by the local-host are replied only by the first VLT-node.
- No proxy-replies for the ARP-requests are received on the VLTi link. These proxy-replies are flooded to other virtual network interfaces by following VLT and VXLAN split-horizon rules.
- If there is a VLTi link failure, there is no change in the existing behavior.
- After clearing ARP or IPv6 neighbor entries in VLT peers, learning of ARP or IPv6 neighbors through ARP request or neighbor solicitation frames from host to gateway (virtual IP) happens only on VLT peer which receives the frame. Other VLT peer learns the ARP or IPv6 neighbor once traffic hits the other node or the ARP/IPv6 neighbor resolution packets hashes to other peer.

## EVPN route selection based on AS path length

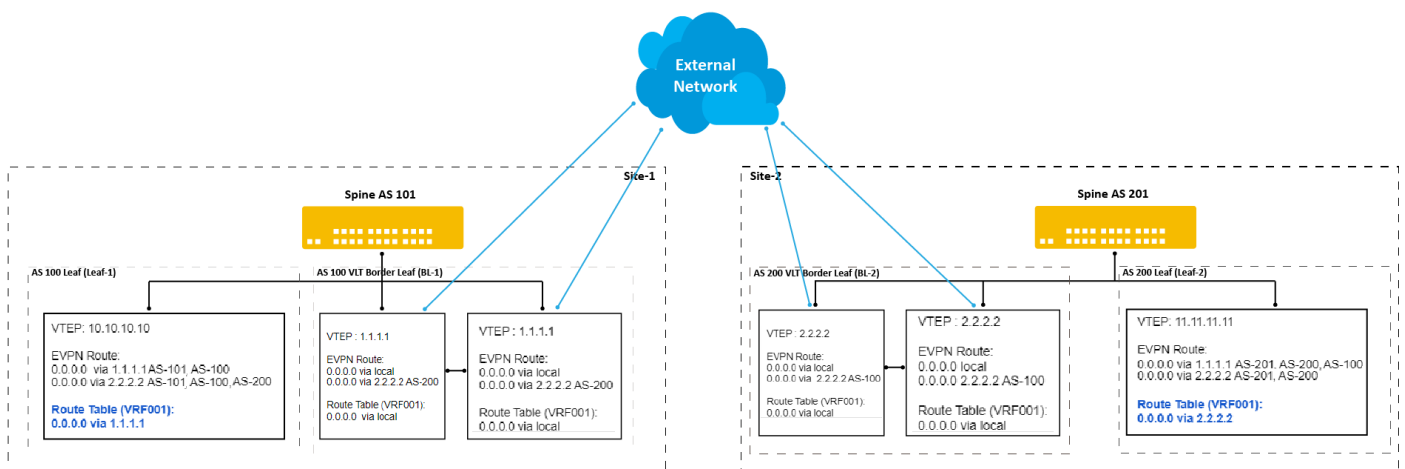
In BGP EVPN VXLAN, consider a scenario where there are two sites, Site1 and Site2, with each site containing its own border leaf switch.

When these sites are interconnected, the routes that are learned or configured on each border leaf switch on a site is advertised through BGP EVPN to other border leaf switches and also to other VTEP leaf switches in both the sites.

If the same prefix is learned or configured on all the border leaf switches, the leaf switches corresponding to both the sites have more than one path for the destination prefix. If all the paths for that route are considered as ECMP paths, the traffic may take a suboptimal path through a remote site.

The AS path length is shorter for the route that is received from the local border leaf switch and longer for routes that are received from the remote border leaf switches. The AS path length is considered for best path selection, so that the path through the local border leaf switch is preferred over the path through the remote border leaf switches.

The following topology depicts this scenario:



For a leaf VTEP to communicate to the external network, a default route is configured on the border leaf VTEP and advertised to all the leaf VTEPs through EVPN. This behavior can also be achieved by advertising the routes that are learned from the external network in the border leaf switch to all other leaf VTEPs.

When the routes that are advertised by the border leaf switch reach the leaf VTEPs, the routes have the AS path updated with the AS through which the route is advertised.

In Leaf-1 switch, the routes originating from BL-1 have the following AS paths: AS-100 and AS-101. The routes originating from BL-2 in Site2 have the following AS paths: AS-200, AS-100, and AS-101.

When traffic hits the default route in leaf-1 and if the default route is installed as ECMP, the traffic is forwarded to either BL-1 in the local site or BL-2 in the remote site based on the traffic hashing.

To send traffic to BL-2 in the other site is suboptimal. For optimal forwarding, the default route must be installed in the local site with BL-1 as the only path.

Achieve this behavior considering the AS path length for EVPN route selection. So, in Leaf-1, if there exists two routes for the prefix from BL-1 and BL-2, the route with the shorter AS path length (AS-100, AS-101) is selected as the best route.

**NOTE:** In an asymmetric scenario, the EVPN L2VPN routes are installed in the routing table. So, in an asymmetric scenario, there is no impact on EVPN L2VPN route selection based on AS path length.

The route selection process is applicable for both default routes and all other type5 route prefixes. For all the EVPN type5 routes in Symmetric IRB, route selection is done by default.

## Supported platforms

All Symmetric IRB supported platforms including: S6010-ON, S4048T-ON, S4100-ON Series, S5200-ON Series, S5448F-ON, Z9432F-ON.

## Restrictions and Limitations

In a scenario where a route has more than one path with different AS path lengths, the `show ip bgp l2vpn evpn` command displays all the routes. The `show ip/ipv6 route vrf vrf-name` command displays only those routes which have the shortest AS path length. However, the `show ip/ipv6 route vrf vrf-name all` command does not display the routes with longer AS path lengths as inactive or active.

## VXLAN BGP EVPN routing

This section describes how EVPN implements overlay routing between L2 segments associated with EVIs belonging to the same tenant on a VTEP. *IETF draft draft-ietf-bess-evpn-inter-subnet-forwarding-05* describes EVPN inter-subnet forwarding, Integrated Routing and Bridging (IRB), and how to use EVPN with IP routing between L2 tenant domains.

You set up overlay routing by assigning a VRF to each tenant, creating a virtual-network interface, and assigning an IP subnet in the VRF to each virtual-network interface. The VTEP acts as the L3 gateway that routes traffic from one tenant subnet to another in the overlay before encapsulating it in the VXLAN header and transporting it over the underlay fabric. On virtual networks that associate with EVIs, EVPN IRB is enabled only after you create a virtual-network interface.

When you enable IRB for a virtual network/EVI, EVPN operation on each VTEP also advertises the local tenant IP-MAC bindings learned on the EVPN-enabled virtual networks to all other VTEPs. The local tenant IP-MAC bindings are learned from ARP or ICMPv6 protocol operation. They advertise as EVPN Type-2 BGP route updates to other VTEPs, each of whom then imports and installs them as ARP/IPv6 neighbor entries in the dataplane.

To enable efficient traffic forwarding on a VTEP, OS10 supports distributed gateway routing. A distributed gateway allows multiple VTEPs to act as the gateway router for a tenant subnet. The VTEP that is located nearest to a host acts as its gateway router.

To enable L3 gateway/IRB functionality for BGP EVPN, configure a VXLAN overlay network and enable routing on a switch:

1. Create a non-default VRF instance for overlay routing. For multi-tenancy, create a VRF instance for each tenant.
2. Configure globally the anycast gateway MAC address used by all VTEPs.
3. Configure a virtual-network interface for each virtual network, (optional) assign it to the tenant VRF, and configure an IP address. Then enable the interface.
4. Configure an anycast gateway IP address for each virtual network. OS10 supports distributed gateway routing.

EVPN supports different types of IRB routing for tenants, VMs, and servers, that connect to each VTEP:

- Centralized routing: For each tenant subnet, one VTEP is designated as the L3 gateway to perform IRB inter-subnet routing. All other VTEPs perform L2 bridging.
- Distributed routing: For each tenant subnet, all VTEPs perform L3 gateway routing for the tenant VMs and servers connected to a VTEP. In a large multi-tenant network, distributed routing allows for more efficient bandwidth use and traffic forwarding. IRB routing is performed either:
  - Only on an ingress VTEP.
  - On both ingress and egress VTEPs.

## Asymmetric IRB routing

In asymmetric IRB routing, IRB routing is performed only on ingress VTEPs. Egress VTEPs perform L2 bridging in the tenant subnet.

An ingress VTEP directly routes packets to a destination host MAC address in the destination virtual-network VNI. An egress VTEP only bridges packets to a host by removing the VXLAN header and forwarding a packet to the local Layer 2 domain using the VNI-to-VLAN mapping.

The ingress VTEP is configured with all destination virtual networks, and has the ARP entries and MAC addresses for all destination hosts in its hardware tables. Each VTEP learns the host MAC and MAC-to-IP bindings using ARP snooping for local addresses and type-2 route advertisements from remote VTEPs.

For VXLAN BGP EVPN examples that use asymmetric IRB, see [Example: VXLAN with BGP EVPN](#) and [Example: VXLAN BGP EVPN — Multiple AS topology](#).

## Symmetric IRB routing

In symmetric IRB routing, both ingress and egress VTEPs perform IRB routing and bridging for a tenant subnet. The ingress VTEP routes packets to an egress VTEP MAC address in an intermediate virtual-network VNI. The egress VTEP then routes the packet again to the destination host in the destination virtual-network VNI.

Using the L3 VNI associated with each tenant VRF, an ingress VTEP routes all traffic for the prefix to an egress VTEP on the L3 VNI. The egress VTEP routes from the L3 VNI to the destination virtual network or bridge domain. The L3 VNI does not have to be associated with an IP address; routing is set up in the data plane using the egress VTEP's MAC address. This behavior is known as IP-VRF to IP-VRF interface-less routing.

The ingress VTEP does not have to be configured with every destination virtual network; it must have the ARP and MAC addresses only to the egress VTEP, not to each host connected to the VTEP. For this reason, symmetric IRB routing allows the overlay network to scale larger than asymmetric routing. Assign the same router MAC address to each VLT peer in a VTEP VLT domain.

Each VTEP learns host MAC and MAC-to-IP bindings using ARP snooping for local addresses, and type-2 and type-5 route advertisements from remote VTEPs. In addition to L3 VNI-connected networks, type-5 route advertisements communicate external routes from a border leaf VTEP to all other VTEPs.

For a VXLAN BGP EVPN example that uses symmetric IRB and Type-5 route, see [Example: VXLAN BGP EVPN — Symmetric IRB](#).

## Configure Symmetric IRB for VXLAN BGP EVPN

### Before you start

1. Follow the procedure in [Configure VXLAN](#) to:
  - Configure the VXLAN overlay network.
  - Enable routing for VXLAN virtual networks. Integrated Routing and Bridging (IRB) is automatically enabled.
  - Enable an overlay routing profile with the number of reserved ARP table entries for VXLAN overlay routing.
2. Follow the procedure in [Configure BGP EVPN for VXLAN](#) to:
  - Configure BGP to advertise EVPN routes.
  - Configure EVPN for VXLAN virtual networks.

For a sample configuration, see [Example: VXLAN with BGP EVPN](#).

### Configure symmetric IRB

1. (Optional) If the switch is a VTEP VLT peer, configure a local router MAC that is used by remote VTEPs as the destination address in VXLAN encapsulated packets sent to the switch in EVPN mode.

If you assign a unique VLT MAC address on each pair of VLT peers, use the same MAC address as the local router MAC. By default, the router MAC is derived as an offset from the local system MAC address.

In a VLT VTEP pair, the router MAC configured in both the VLT peers must be the same. Router MAC configuration is mandatory for VTEP VLT peers.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac nn:nn:nn:nn:nn:nn
```

2. Configure a non-default VRF with a dedicated VXLAN VNI for each tenant VRF in EVPN mode. The tenant VRF is created using the `ip vrf` command when you enable overlay routing with IRB; see [Enable overlay routing between virtual networks](#). The VXLAN VNI associated with the tenant VRF for EVPN symmetric IRB must be unique on the switch.

By default, the route distinguisher value is auto-generated. To reconfigure it, use the `rd A.B.C.D:[1-65535]` command. The route target value is a mandatory entry.

```
OS10(config-evpn)# vrf tenant-vrf-name
OS10(config-evpn-vrf-vrf-tenant)# vni vxlan-vni
OS10(config-evpn-vrf-vrf-tenant)# rd {A.B.C.D:[1-65535]}
OS10(config-evpn-vrf-vrf-tenant)# route-target {auto | value {import | export | both}
[asn4]}
OS10(config-evpn-vrf-vrf-tenant)# exit
```

- (Optional) Advertise the IP prefixes learned from external networks and directly connected networks into EVPN type-5 route advertisements in EVPN-VRF mode; for example:

```
OS10(config)# evpn
OS10(config-evpn)# vrf vrf-tenant1
OS10(config-evpn-vrf-vrf-tenant1)# advertise {ipv4 | ipv6} {connected | static| ospf
| bgp} [route-map map-name]
```

- (Optional) To redistribute EVPN routes to a BGP or OSPF neighbor, configure the redistribution of L2VPN EVPN routes into BGP or OSPF IPv4/IPv6 routes on a border leaf VTEP in ROUTER-BGP or ROUTER-OSPF mode; for example:

```
OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# vrf blue
OS10(conf-router-bgp-101-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute l2vpn evpn [route-map map-name]
```

- Verify the VXLAN BGP EVPN with symmetric IRB configuration.

#### Display the EVPN instance configuration

```
OS10# show evpn evi 10000

EVI : 10000, State : up
  Bridge-Domain : Virtual-Network 10000, VNI 10000
  Route-Distinguisher : 1:110.111.170.195:10000(auto)
  Route-Targets : 0:10000:16787216(auto) both
  Inclusive Multicast : 110.111.170.107
  IRB : Enabled(VRF-TENANT-1)

OS10# show evpn evi 20000
EVI : 20000, State : up
  Bridge-Domain : Virtual-Network 20000, VNI 20000
  Route-Distinguisher : 1:110.111.170.195:20000(auto)
  Route-Targets : 0:20000:16797216(auto) both
  Inclusive Multicast :
  IRB : Enabled(VRF-TENANT-1)
```

#### Display the EVPN Type 2 routes for host MAC/IP addresses

```
show evpn mac-ip
Type -(lcl): Local (rmt): remote
EVI  Mac Address      Type Seq No Host-IP      Interface/Next-Hops
10000 00:00:0b:0b:0b:0a lcl  0      10.10.10.10 ethernet1/1/6
10000 14:18:77:25:4e:82 rmt  0      10.10.10.11 110.111.170.107
```

#### Display the VRF instances used to forward EVPN routes in VXLAN overlay networks

```
OS10# show evpn vrf
VXLAN-VNI  EVI  Virtual-Network-Instance  VRF-Name
30          30   30                          vrf_30
40          40   40                          vrf_40

OS10# show evpn vrf l3-vni
VRF : vrf_30, State : up
  L3-VNI : 3030
  Route-Distinguisher : 1:80.80.1.1:3030(auto)
  Route-Targets : 0:200:268438486(auto) both
  Remote VTEP : 4.4.4.4

VRF : vrf_40, State : up
```

```
L3-VNI : 4040
Route-Distinguisher : 1:80.80.1.1:4040(auto)
Route-Targets : 0:200:268439496(auto) both
Remote VTEP : 4.4.4.4
```

```
VRF : vrf_50, State : up
L3-VNI : 5050
Route-Distinguisher : 1:80.80.1.1:5050(auto)
Route-Targets : 0:200:268430506(auto) both
Remote VTEP : 4.4.4.4
```

### Display the router MAC address used in overlay network for symmetric IRB

```
show evpn router-mac
Local Router MAC : 14:18:77:25:4e:4d
```

```
Remote-VTEP Router's-MAC
4.4.4.4      14:18:77:25:6f:4d
5.5.5.5      00:00:01:00:a3:b4
```

### Display the learned EVPN Type 5 routes

```
OS10# show ip bgp l2vpn evpn
BGP local RIB : Routes to be Added , Replaced , Withdrawn
BGP local router ID is 95.0.0.4
Status codes: s suppressed, S stale, d dampened, h history, * valid, > best
Path source: I - internal, a - aggregate, c - confed-external,
r - redistributed/network, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*>r Route distinguisher: 4.4.4.4:65001 VNI:65001 [5]:[0]:[24]:[11.11.11.0]:[0.0.0.0]/224	4.4.4.4	0	100	32768	?
*>r Route distinguisher: 3.3.3.3:65002 VNI:65002 [5]:[0]:[24]:[12.12.12.0]:[0.0.0.0]/224	3.3.3.3	0	100	0 100 101	?
*>r Route distinguisher: 4.4.4.4:101 VNI:101 [2]:[0]:[48]:[14:18:77:25:6f:4d]:[32]:[11.11.11.2]/224	4.4.4.4	0	100	32768	?
*>r Route distinguisher: 3.3.3.3:102 VNI:102 [2]:[0]:[48]:[14:18:77:25:8f:6d]:[32]:[12.12.12.1]/224	3.3.3.3	0	100	0 100 101	?
*> Route distinguisher: 3.3.3.3:101 [3]:[0]:[32]:[3.3.3.3]/152	3.3.3.3	0	100	0 100 101	?
*>r Route distinguisher: 4.4.4.4:101 [3]:[0]:[32]:[4.4.4.4]/152	4.4.4.4	0	100	32768	?
*>r Route distinguisher: 4.4.4.4:102 [3]:[0]:[32]:[4.4.4.4]/152	4.4.4.4	0	100	32768	?

```
OS10# show ip route vrf blue
Codes: C - connected
S - static
B - BGP, IN - internal BGP, EX - external BGP, EV - EVPN BGP
O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
E2 - OSPF external type 2, * - candidate default,
+ - summary route, > - non-active route
Gateway of last resort is not set
```

Destination	Gateway	Dist/Metric	Last Change
C 11.11.11.0/24	via 11.11.11.1	0/0	1 day 02:54:39
	virtual-network101		
B EV 15.15.15.2/32	via 4.4.4.4	200/0	1 day 02:09:19
B EV 15.15.15.0/24	via 4.4.4.4	200/0	1 day 02:09:19

```
B EV 11.11.11.2/32 via 4.4.4.4 100/0 1 day 05:10:11
B EV 12.12.12.0/24 via 3.3.3.3 100/0 1 day 00:10:01
```

## Example: VXLAN with BGP EVPN with asymmetric IRB

The following VXLAN with BGP EVPN example uses a Clos leaf-spine topology with VXLAN tunnel endpoints (VTEPs). The individual switch configuration shows how to set up an end-to-end VXLAN. eBGP is used to exchange IP routes in the IP underlay network, and EVPN routes in the VXLAN overlay network. All spine nodes are in one autonomous system—AS 101. All leaf nodes are in another autonomous system—AS 100.

- On VTEPs 1 and 2: Access ports are assigned to the virtual network using a switch-scoped VLAN. EVPN is configured using auto-EVI mode.
- On VTEPs 3 and 4: Access ports are assigned to the virtual network using a port-scoped VLAN. The EVPN instance is configured using manual configuration mode. The RD and RT are configured using auto mode.

All VTEPs perform asymmetric IRB routing, in which:

- IRB routing is performed only on ingress VTEPs.
- Egress VTEPs perform IRB bridging.

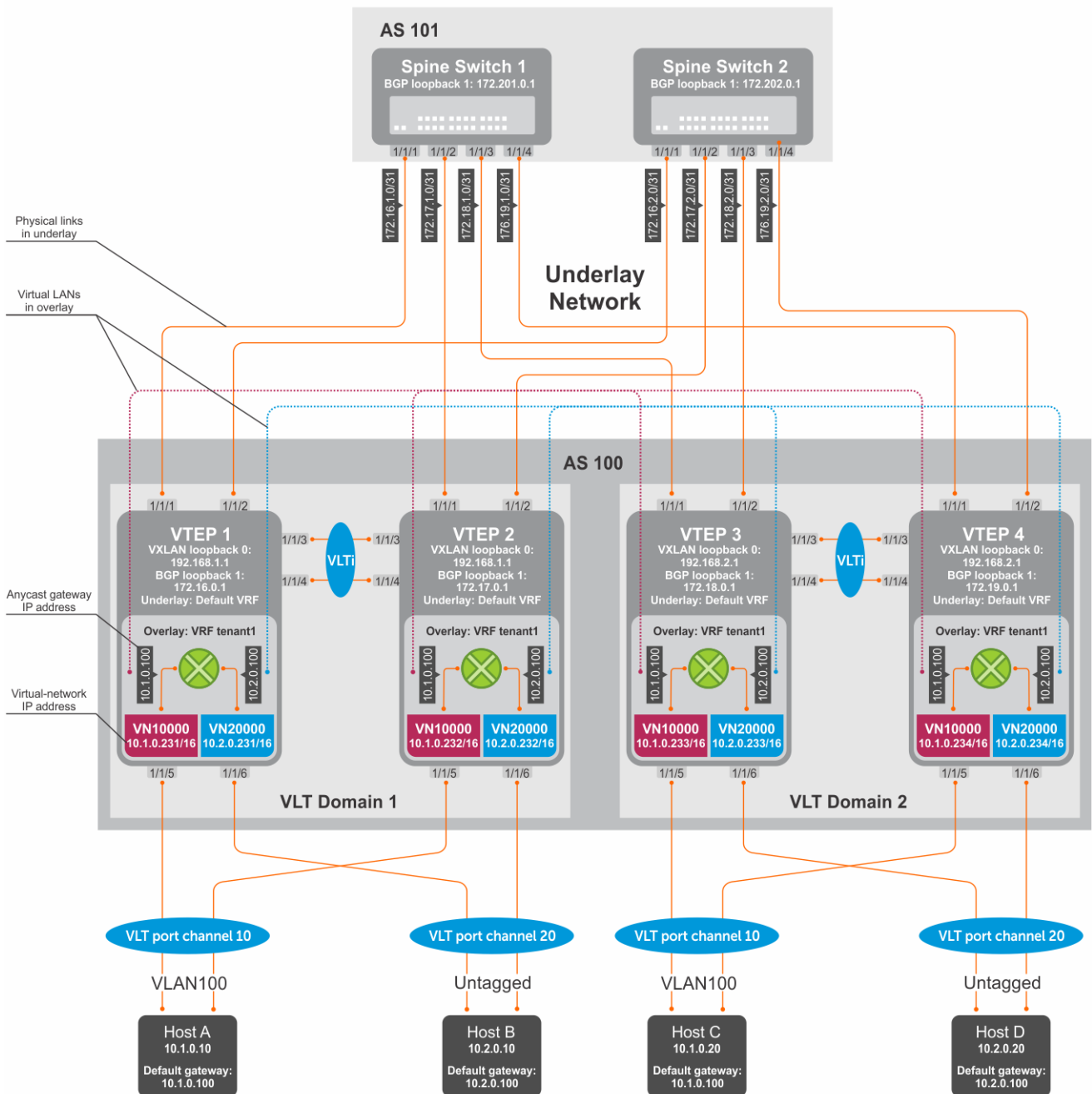


Figure 5. VXLAN BGP EVPN use case

## VTEP 1 Leaf Switch

1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# exit
```

2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Assign VLAN member interfaces to the virtual networks.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
OS10(config)# interface vlan200
OS10(config-if-vl-200)# virtual-network 20000
OS10(config-if-vl-200)# no shutdown
OS10(config-if-vl-200)# exit
```

### 5. Configure access ports as VLAN members for a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# switchport trunk allowed vlan 100
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# switchport access vlan 200
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit
```

### 6. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.16.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.16.2.0/31
OS10(config-if-eth1/1/2)# exit
```

### 7. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.16.0.1
```

```
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(config-router-bgp-af)# redistribute connected
OS10(config-router-bgp-af)# exit
```

## 8. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.16.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.16.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 9. Configure a Loopback interface for BGP EVPN peering different from the VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.16.0.1/32
OS10(conf-if-lo-1)# exit
```

## 10. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 11. Configure EVPN.

Configure the EVPN instance, RD, and RT using auto-EVI mode:

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# exit
```

## 12. Configure VLT.

**Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.**

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.0/31
OS10(config-if-vl-4000)# exit
```

**Configure the VLT port channel.**

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit
```

**Configure the VLTi member links.**

```
OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

**Configure the VLT domain.**

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.1
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(conf-vlt-1)# exit
```

**Configure UFD with uplink VLT ports and downlink network ports.**

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

**Configure iBGP IPv4 peering between VLT peers.**

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.1
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 13. Configure IP switching in the overlay network.

**Create a tenant VRF.**

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

**Configure an anycast gateway MAC address.**

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.231/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.231/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 2 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using the same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.1.1/32
OS10(config-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Assign VLAN member interfaces to the virtual networks.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
OS10(config)# interface vlan200
OS10(config-if-vl-200)# virtual-network 20000
OS10(config-if-vl-200)# no shutdown
OS10(config-if-vl-200)# exit
```

### 5. Configure access ports as VLAN members for a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# switchport trunk allowed vlan 100
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
```

```

OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# switchport access vlan 200
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit

```

## 6. Configure upstream network-facing ports.

```

OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.17.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.17.2.0/31
OS10(config-if-eth1/1/2)# exit

```

## 7. Configure eBGP.

```

OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.17.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(config-router-bgp-af)# redistribute connected
OS10(config-router-bgp-af)# exit

```

## 8. Configure eBGP for the IPv4 point-to-point peering.

```

OS10(config-router-bgp-100)# neighbor 172.17.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.17.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit

```

## 9. Configure a Loopback interface for BGP EVPN peering different from VLT peer IP address.

```

OS10(config)# interface loopback1
OS10(config-if-lo-1)# no shutdown
OS10(config-if-lo-1)# ip address 172.17.0.1/32
OS10(config-if-lo-1)# exit

```

## 10. Configure BGP EVPN peering.

```

OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended

```

```

OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-bgp-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit

```

## 11. Configure EVPN.

Configure the EVPN instance, RD, and RT using auto-EVI mode:

```

OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# exit

```

## 12. Configure VLT.

**Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.**

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.1/31
OS10(config-if-vl-4000)# exit

```

**Configure the VLT port channel.**

```

OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit

```

**Configure VLTi member links.**

```

OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit

```

**Configure the VLT domain.**

```

OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.2
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4

```

```
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

### Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 13. Configure IP switching in overlay network.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.232/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.232/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 3 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.2.1/32
OS10(config-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
```

```
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

#### 4. Configure unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

#### 5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
OS10(conf-if-po-10)# no switchport access vlan
OS10(conf-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# no switchport
OS10(conf-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# no shutdown
OS10(conf-if-po-20)# switchport mode trunk
OS10(conf-if-po-20)# no switchport access vlan
OS10(conf-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(conf-if-eth1/1/6)# no shutdown
OS10(conf-if-eth1/1/6)# channel-group 20 mode active
OS10(conf-if-eth1/1/6)# no switchport
OS10(conf-if-eth1/1/6)# exit
```

#### 6. Add the access ports to virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn-20000)# exit
```

#### 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.18.1.0/31
OS10(conf-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/2)# ip address 172.18.2.0/31
OS10(conf-if-eth1/1/2)# exit
```

#### 8. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.18.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
```

```
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

### 9. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.18.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.18.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 10. Configure a Loopback interface for BGP EVPN peering different from VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.18.0.1/32
OS10(conf-if-lo-1)# exit
```

### 11. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 12. Configure EVPN.

Configure the EVPN instance in manual configuration mode, and RD and RT configuration in auto mode:

```
OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# vni 10000
OS10(config-evpn-evi-10000)# rd auto
```

```

OS10(config-evpn-evi-10000)# route-target auto
OS10(config-evpn-evi-10000)# exit

OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd auto
OS10(config-evpn-evi-20000)# route-target auto
OS10(config-evpn-evi-20000)# exit
OS10(config-evpn)# exit

```

### 13. Configure VLT.

#### Configure a VLTi VLAN for the virtual network.

```

OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vlti-vlan 100
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit

```

#### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.10/31
OS10(config-if-vl-4000)# exit

```

#### Configure the VLT port channels.

```

OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit

```

#### Configure VLTi member links.

```

OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit

```

#### Configure the VLT domain.

```

OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.3
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(conf-vlt-1)# exit

```

#### Configure UFD with uplink VLT ports and downlink network ports.

```

OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit

```

### Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.11
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 14. Configure IP routing in the overlay network.

#### Create the tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

#### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

#### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network 10000
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.1.0.233/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(conf-if-vn-10000)# no shutdown
OS10(conf-if-vn-10000)# exit

OS10(config)# interface virtual-network 20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.233/16
OS10(conf-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-if-vn-20000)# exit
```

## VTEP 4 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.2.1/32
OS10(conf-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Configure the unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

## 5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit
```

## 6. Add the access ports to the virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn)# exit
```

## 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.19.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.19.2.0/31
OS10(config-if-eth1/1/2)# exit
```

## 8. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.19.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

## 9. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.19.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.19.2.1
```

```

OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit

```

#### 10. Configure a Loopback interface for BGP EVPN peering different from the VLT peer IP address.

```

OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.19.0.1/32
OS10(conf-if-lo-1)# exit

```

#### 11. Configure BGP EVPN peering.

```

OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit

```

#### 12. Configure EVPN.

Configure the EVPN instance manual configuration mode, and RD, and RT configuration in auto mode:

```

OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# vni 10000
OS10(config-evpn-evi-10000)# rd auto
OS10(config-evpn-evi-10000)# route-target auto
OS10(config-evpn-evi-10000)# exit

OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd auto
OS10(config-evpn-evi-20000)# route-target auto
OS10(config-evpn-evi-20000)# exit
OS10(config-evpn)# exit

```

#### 13. Configure VLT.

### Configure a VLTi VLAN for the virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vlti-vlan 100
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit
```

### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.11/31
OS10(config-if-vl-4000)# exit
```

### Configure VLT port channels.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt-port-channel 10
OS10(config-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt-port-channel 20
OS10(config-if-po-20)# exit
```

### Configure VLTi member links.

```
OOS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

### Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.4
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

### Configure iBGP IPv4 peering between the VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.10
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 14. Configure IP routing in the overlay network.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.234/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.234/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## Spine Switch 1

### 1. Configure downstream ports on underlay links to the leaf switches.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# ip address 172.16.1.1/31
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# ip address 172.17.1.1/31
OS10(config-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# ip address 172.18.1.1/31
OS10(config-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# ip address 172.19.1.1/31
OS10(config-if-eth1/1/4)# exit
```

### 2. Configure eBGP.

```
OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.201.0.1
OS10(config-router-bgp-101)# address-family ipv4 unicast
OS10(config-router-bgp-101-af)# redistribute connected
OS10(config-router-bgp-101-af)# exit
```

### 3. Configure eBGP IPv4 peer sessions on the P2P links.

```
OS10(config-router-bgp-101)# neighbor 172.16.1.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-neighbor-af)# no sender-side-loop-detection
OS10(config-router-neighbor-af)# exit
OS10(config-router-neighbor)# exit
```

```

OS10(conf-router-bgp-101)# neighbor 172.17.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit
OS10(conf-router-bgp-101)# exit

```

#### 4. Configure a Loopback interface for BGP EVPN peering.

```

OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.201.0.1/32
OS10(conf-if-lo-1)# exit

```

#### 5. Configure BGP EVPN peer sessions.

```

OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# neighbor 172.16.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit

```

```

OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

```

## Spine Switch 2

### 1. Configure downstream ports on the underlay links to the leaf switches.

```

OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# ip address 172.16.2.1/31
OS10(conf-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/2)# ip address 172.17.2.1/31
OS10(conf-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# ip address 172.18.2.1/31
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# ip address 172.19.2.1/31
OS10(conf-if-eth1/1/4)# exit

```

### 2. Configure eBGP.

```

OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.202.0.1
OS10(config-router-bgp-101)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit

```

### 3. Configure eBGP IPv4 peer sessions on the P2P links.

```

OS10(conf-router-bgp-101)# neighbor 172.16.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

```

```

OS10(conf-router-bgp-101)# neighbor 172.18.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit
OS10(conf-router-bgp-101)# exit

```

#### 4. Configure a Loopback interface for BGP EVPN peering.

```

OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.202.0.1/32
OS10(conf-if-lo-1)# exit

```

#### 5. Configure BGP EVPN peer sessions.

```

OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# neighbor 172.16.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4

```

```

OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

```

## Verify VXLAN with BGP EVPN configuration.

### 1. Verify virtual network configurations.

```

LEAF1# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
Virtual Network: 10000
  Members:
    VLAN 100: port-channel10, port-channel1000
  VxLAN Virtual Network Identifier: 10000
  Source Interface: loopback0(192.168.1.1)
  Remote-VTEPs (flood-list): 192.168.2.1(CP)

Virtual Network: 20000
  Members:
    Untagged: port-channel20
  VLAN 200: port-channel1000
  VxLAN Virtual Network Identifier: 20000
  Source Interface: loopback0(192.168.1.1)
  Remote-VTEPs (flood-list): 192.168.2.1(CP)
LEAF1#

```

### 2. Verify EVPN configurations and EVPN parameters.

```

LEAF1# show evpn evi

EVI : 10000, State : up
  Bridge-Domain      : Virtual-Network 10000, VNI 10000
  Route-Distinguisher : 1:192.168.1.1:10000(auto)
  Route-Targets      : 0:100:268445456(auto) both
  Inclusive Multicast : 192.168.2.1
  IRB                : Enabled(tenant1)

EVI : 20000, State : up
  Bridge-Domain      : Virtual-Network 20000, VNI 20000
  Route-Distinguisher : 1:192.168.1.1:20000(auto)
  Route-Targets      : 0:100:268455456(auto) both
  Inclusive Multicast : 192.168.2.1
  IRB                : Enabled(tenant1)
LEAF1#

```

### 3. Verify BGP EVPN neighborship between leaf and spine nodes.

```

LEAF1# show ip bgp l2vpn evpn summary
BGP router identifier 172.16.0.1 local AS number 100
Neighbor    AS    MsgRcvd  MsgSent  Up/Down  State/Pfx
172.201.0.1 101  1132    1116    13:29:00 27
172.202.0.1 101  1131    1118    13:29:02 28
LEAF1#

```

### 4. Check connectivity between host A and host B.

```

root@HOST-A:~# ping 10.2.0.10 -c 5
PING 10.2.0.10 (10.2.0.10) 56(84) bytes of data.
64 bytes from 10.2.0.10: icmp_seq=1 ttl=63 time=0.824 ms
64 bytes from 10.2.0.10: icmp_seq=2 ttl=63 time=0.847 ms
64 bytes from 10.2.0.10: icmp_seq=3 ttl=63 time=0.835 ms

```

```

64 bytes from 10.2.0.10: icmp_seq=4 ttl=63 time=0.944 ms
64 bytes from 10.2.0.10: icmp_seq=5 ttl=63 time=0.806 ms

--- 10.2.0.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4078ms
rtt min/avg/max/mdev = 0.806/0.851/0.944/0.051 ms
root@HOST-A:~#

```

### 5. Check connectivity between host A and host C.

```

root@HOST-A:~# ping 10.1.0.20 -c 5
PING 10.1.0.20 (10.1.0.20) 56(84) bytes of data.
64 bytes from 10.1.0.20: icmp_seq=1 ttl=64 time=0.741 ms
64 bytes from 10.1.0.20: icmp_seq=2 ttl=64 time=0.737 ms
64 bytes from 10.1.0.20: icmp_seq=3 ttl=64 time=0.772 ms
64 bytes from 10.1.0.20: icmp_seq=4 ttl=64 time=0.799 ms
64 bytes from 10.1.0.20: icmp_seq=5 ttl=64 time=0.866 ms

--- 10.1.0.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4061ms
rtt min/avg/max/mdev = 0.737/0.783/0.866/0.047 ms
root@HOST-A:~#

```

### 6. Check connectivity between host A and host D.

```

root@HOST-A:~# ping 10.2.0.20 -c 5
PING 10.2.0.20 (10.2.0.20) 56(84) bytes of data.
64 bytes from 10.2.0.20: icmp_seq=1 ttl=63 time=0.707 ms
64 bytes from 10.2.0.20: icmp_seq=2 ttl=63 time=0.671 ms
64 bytes from 10.2.0.20: icmp_seq=3 ttl=63 time=0.687 ms
64 bytes from 10.2.0.20: icmp_seq=4 ttl=63 time=0.640 ms
64 bytes from 10.2.0.20: icmp_seq=5 ttl=63 time=0.644 ms

--- 10.2.0.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4089ms
rtt min/avg/max/mdev = 0.640/0.669/0.707/0.041 ms
root@HOST-A:~#

```

**NOTE:** Follow Steps 1 to 6 to check ping connectivity between combinations of other hosts, and between hosts through different virtual-network IP addresses.

## Example: VXLAN BGP EVPN — Multiple AS topology with asymmetric IRB

The following VXLAN with BGP EVPN example uses a Clos leaf-spine example. The individual switch configuration shows how to set up an end-to-end VXLAN. eBGP is used to exchange IP routes in the IP underlay network, and EVPN routes in the VXLAN overlay network. All VTEPs perform asymmetric IRB routing, in which:

- IRB routing is performed only on ingress VTEPs.
- Egress VTEPs perform IRB bridging.

In this example, each node in the spine network and each VTEP in the leaf network belongs to a different autonomous system. Spine switch 1 is in AS 101. Spine switch 2 is in AS 102. For leaf nodes, VLT domain 1 is in AS 99; VLT domain 2 is in AS 100.

- On VTEPs 1 and 2: Access ports are assigned to the virtual network using a switch-scoped VLAN. EVPN instance along with RD and RT values are configured in manual mode.
- On VTEPs 3 and 4: Access ports are assigned to the virtual network using a port-scoped VLAN. EVPN instance along with RD and RT values are configured in manual mode.

**NOTE:** In multiple AS topology, you can configure route targets in an easier way using the `disable-rt-asn` command with `route-target auto` or `auto evi` commands.

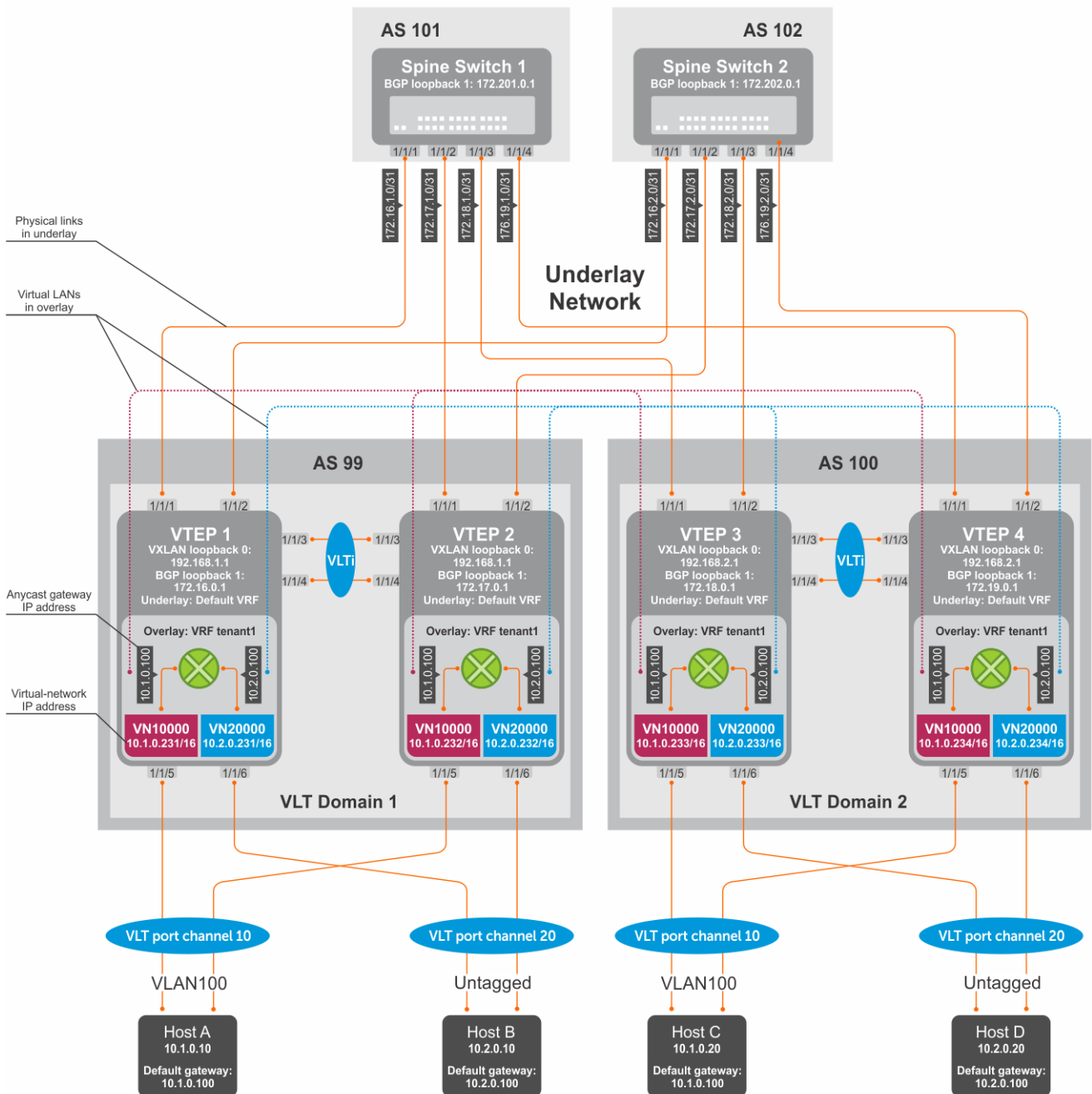


Figure 6. VXLAN BGP EVPN with multiple AS

## VTEP 1 Leaf Switch

1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# exit
```

2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Assign VLAN member interfaces to the virtual networks.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
OS10(config)# interface vlan200
OS10(config-if-vl-200)# virtual-network 20000
OS10(config-if-vl-200)# no shutdown
OS10(config-if-vl-200)# exit
```

### 5. Configure access ports as VLAN members for a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# switchport trunk allowed vlan 100
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# switchport access vlan 200
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit
```

### 6. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.16.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.16.2.0/31
OS10(config-if-eth1/1/2)# exit
```

### 7. Configure eBGP.

```
OS10(config)# router bgp 99
OS10(config-router-bgp-99)# router-id 172.16.0.1
```

```
OS10(config-router-bgp-99)# address-family ipv4 unicast
OS10(config-router-bgp-af)# redistribute connected
OS10(config-router-bgp-af)# exit
```

## 8. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-99)# neighbor 172.16.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-99)# neighbor 172.16.2.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-99)# exit
```

## 9. Configure a Loopback interface for BGP EVPN peering different from the VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.16.0.1/32
OS10(conf-if-lo-1)# exit
```

## 10. Configure BGP EVPN peering.

```
OS10(config)# router bgp 99
OS10(config-router-bgp-99)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-99)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 11. Configure EVPN.

Configure the EVPN instance with RD and RT values in manual mode:

```
OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# vni 10000
OS10(config-evpn-evi-10000)# rd 192.168.1.1:10000
OS10(config-evpn-evi-10000)# route-target 99:10000 both
OS10(config-evpn-evi-10000)# route-target 100:10000 import
OS10(config-evpn-evi-10000)#exit

OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd 192.168.1.1:20000
OS10(config-evpn-evi-20000)# route-target 99:20000 both
```

```
OS10(config-evpn-evi-20000)# route-target 100:20000 import
OS10(config-evpn-evi-20000)#exit
OS10(config-evpn)#
```

## 12. Configure VLT.

**Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.**

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.0/31
OS10(config-if-vl-4000)# exit
```

**Configure the VLT port channel.**

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit
```

**Configure the VLTi member links.**

```
OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

**Configure the VLT domain.**

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.1
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(conf-vlt-1)# exit
```

**Configure UFD with uplink VLT ports and downlink network ports.**

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

**Configure iBGP IPv4 peering between VLT peers.**

```
OS10(config)# router bgp 99
OS10(config-router-bgp-99)# neighbor 172.16.250.1
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-99)# exit
```

## 13. Configure IP switching in the overlay network.

**Create a tenant VRF**

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.231/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.231/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 2 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using the same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.1.1/32
OS10(config-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn)# exit
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Assign VLAN member interfaces to the virtual networks.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
OS10(config)# interface vlan200
OS10(config-if-vl-200)# virtual-network 20000
OS10(config-if-vl-200)# no shutdown
OS10(config-if-vl-200)# exit
```

### 5. Configure access ports as VLAN members for a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# switchport trunk allowed vlan 100
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit
```

```

OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# no switchport
OS10(conf-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# no shutdown
OS10(conf-if-po-20)# switchport mode trunk
OS10(conf-if-po-20)# switchport access vlan 200
OS10(conf-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(conf-if-eth1/1/6)# no shutdown
OS10(conf-if-eth1/1/6)# channel-group 20 mode active
OS10(conf-if-eth1/1/6)# no switchport
OS10(conf-if-eth1/1/6)# exit

```

## 6. Configure upstream network-facing ports.

```

OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.17.1.0/31
OS10(conf-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/2)# mtu 1650
OS10(conf-if-eth1/1/2)# ip address 172.17.2.0/31
OS10(conf-if-eth1/1/2)# exit

```

## 7. Configure eBGP.

```

OS10(config)# router bgp 99
OS10(config-router-bgp-99)# router-id 172.17.0.1
OS10(config-router-bgp-99)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit

```

## 8. Configure eBGP for the IPv4 point-to-point peering.

```

OS10(config-router-bgp-99)# neighbor 172.17.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-99)# neighbor 172.17.2.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-99)# exit

```

## 9. Configure a Loopback interface for BGP EVPN peering different from VLT peer IP address.

```

OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.17.0.1/32
OS10(conf-if-lo-1)# exit

```

## 10. Configure BGP EVPN peering.

```

OS10(config)# router bgp 99
OS10(config-router-bgp-99)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown

```

```

OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-99)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-bgp-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-99)# exit

```

## 11. Configure EVPN.

Configure the EVPN instance with RD and RT in manual configuration mode:

```

OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# vni 10000
OS10(config-evpn-evi-10000)# rd 192.168.1.1:10000
OS10(config-evpn-evi-10000)# route-target 99:10000 both
OS10(config-evpn-evi-10000)# route-target 100:10000 import
OS10(config-evpn-evi-10000)#exit

OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd 192.168.1.1:20000
OS10(config-evpn-evi-20000)# route-target 99:20000 both
OS10(config-evpn-evi-20000)# route-target 100:20000 import
OS10(config-evpn-evi-20000)#exit
OS10(config-evpn)#

```

## 12. Configure VLT.

**Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.**

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.1/31
OS10(config-if-vl-4000)# exit

```

**Configure the VLT port channel.**

```

OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit

```

**Configure VLTi member links.**

```

OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown

```

```
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

### Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.2
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

### Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 99
OS10(config-router-bgp-99)# neighbor 172.16.250.0
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-99)# exit
```

## 13. Configure IP switching in overlay network.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.232/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.232/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 3 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.2.1/32
OS10(config-if-lo-0)# exit
```

## 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

## 3. Configure VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

## 4. Configure unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

## 5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
OS10(conf-if-po-10)# no switchport access vlan
OS10(conf-if-po-10)# exit
```

```
OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# no switchport
OS10(conf-if-eth1/1/5)# exit
```

```
OS10(config)# interface port-channel20
OS10(conf-if-po-20)# no shutdown
OS10(conf-if-po-20)# switchport mode trunk
OS10(conf-if-po-20)# no switchport access vlan
OS10(conf-if-po-20)# exit
```

```
OS10(config)# interface ethernet1/1/6
OS10(conf-if-eth1/1/6)# no shutdown
OS10(conf-if-eth1/1/6)# channel-group 20 mode active
OS10(conf-if-eth1/1/6)# no switchport
OS10(conf-if-eth1/1/6)# exit
```

## 6. Add the access ports to virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn-10000)# exit
```

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn-20000)# exit
```

## 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.18.1.0/31
OS10(conf-if-eth1/1/1)# exit
```

```
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
```

```
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.18.2.0/31
OS10(config-if-eth1/1/2)# exit
```

## 8. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.18.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

## 9. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.18.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.18.2.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 10. Configure a Loopback interface for BGP EVPN peering different from VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.18.0.1/32
OS10(conf-if-lo-1)# exit
```

## 11. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 12. Configure EVPN.

Configure the EVPN instance, RD, and RT in manual configuration mode:

```
OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# vni 10000
```

```

OS10(config-evpn-evi-10000)# rd 192.168.2.1:10000
OS10(config-evpn-evi-10000)# route-target 99:10000 import
OS10(config-evpn-evi-10000)# route-target 100:10000 both
OS10(config-evpn-evi-10000)#exit

OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd 192.168.2.1:20000
OS10(config-evpn-evi-20000)# route-target 99:20000 import
OS10(config-evpn-evi-20000)# route-target 100:20000 both
OS10(config-evpn-evi-20000)#exit
OS10(config-evpn)#

```

### 13. Configure VLT.

#### Configure a VLTi VLAN for the virtual network.

```

OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vlti-vlan 100
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit

```

#### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.10/31
OS10(config-if-vl-4000)# exit

```

#### Configure the VLT port channels.

```

OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit

```

#### Configure VLTi member links.

```

OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit

```

#### Configure the VLT domain.

```

OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.3
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(conf-vlt-1)# exit

```

#### Configure UFD with uplink VLT ports and downlink network ports.

```

OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit

```

### Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.11
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 14. Configure IP routing in the overlay network.

#### Create the tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

#### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

#### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network10000
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.1.0.233/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(conf-if-vn-10000)# no shutdown
OS10(conf-if-vn-10000)# exit

OS10(config)# interface virtual-network20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.233/16
OS10(conf-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-if-vn-20000)# exit
```

## VTEP 4 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.2.1/32
OS10(conf-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Configure the unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

## 5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit
```

## 6. Add the access ports to the virtual networks.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# member-interface port-channel 10 vlan-tag 100
OS10(config-vn)# exit

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn)# exit
```

## 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.19.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.19.2.0/31
OS10(config-if-eth1/1/2)# exit
```

## 8. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.19.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

## 9. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.19.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.19.2.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# no shutdown
```

```
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 10. Configure a Loopback interface for BGP EVPN peering different from the VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(config-if-lo-1)# no shutdown
OS10(config-if-lo-1)# ip address 172.19.0.1/32
OS10(config-if-lo-1)# exit
```

## 11. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 102
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 12. Configure EVPN.

Configure the EVPN instance, RD, RT in manual configuration mode:

```
OS10(config)# evpn
OS10(config-evpn)# evi 10000
OS10(config-evpn-evi-10000)# vni 10000
OS10(config-evpn-evi-10000)# rd 192.168.2.1:10000
OS10(config-evpn-evi-10000)# route-target 99:10000 import
OS10(config-evpn-evi-10000)# route-target 100:10000 both
OS10(config-evpn-evi-10000)#exit

OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd 192.168.2.1:20000
OS10(config-evpn-evi-20000)# route-target 99:20000 import
OS10(config-evpn-evi-20000)# route-target 100:20000 both
OS10(config-evpn-evi-20000)#exit
OS10(config-evpn)#
```

## 13. Configure VLT.

Configure a VLTi VLAN for the virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vlti-vlan 100
OS10(config-vn-10000)# exit

OS10(config)# virtual-network 20000
```

```
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit
```

### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.11/31
OS10(config-if-vl-4000)# exit
```

### Configure VLT port channels.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt-port-channel 10
OS10(config-if-po-10)# exit

OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt-port-channel 20
OS10(config-if-po-20)# exit
```

### Configure VLTi member links.

```
OOS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

### Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.4
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

### Configure iBGP IPv4 peering between the VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.10
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 14. Configure IP routing in the overlay network.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

## Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.234/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.234/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## Spine Switch 1

### 1. Configure downstream ports on underlay links to the leaf switches.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# ip address 172.16.1.1/31
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# ip address 172.17.1.1/31
OS10(config-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# ip address 172.18.1.1/31
OS10(config-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# ip address 172.19.1.1/31
OS10(config-if-eth1/1/4)# exit
```

### 2. Configure eBGP.

```
OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.201.0.1
OS10(config-router-bgp-101)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit
```

### 3. Configure eBGP IPv4 peer sessions on the P2P links.

```
OS10(config-router-bgp-101)# neighbor 172.16.1.0
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-101)# neighbor 172.17.1.0
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-101)# neighbor 172.18.1.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-101)# neighbor 172.19.1.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
```

```
OS10(conf-router-neighbor)# exit
OS10(conf-router-bgp-101)# exit
```

#### 4. Configure a Loopback interface for BGP EVPN peering.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.201.0.1/32
OS10(conf-if-lo-1)# exit
```

#### 5. Configure BGP EVPN peer sessions.

```
OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# neighbor 172.16.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 99
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 99
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit
```

## Spine Switch 2

### 1. Configure downstream ports on the underlay links to the leaf switches.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# ip address 172.16.2.1/31
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# ip address 172.17.2.1/31
OS10(config-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# ip address 172.18.2.1/31
OS10(config-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# ip address 172.19.2.1/31
OS10(config-if-eth1/1/4)# exit
```

### 2. Configure eBGP.

```
OS10(config)# router bgp 102
OS10(config-router-bgp-102)# router-id 172.202.0.1
OS10(config-router-bgp-102)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit
```

### 3. Configure eBGP IPv4 peer sessions on the P2P links.

```
OS10(config-router-bgp-102)# neighbor 172.16.2.0
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-102)# neighbor 172.17.2.0
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-102)# neighbor 172.18.2.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-102)# neighbor 172.19.2.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-102)# exit
```

### 4. Configure a Loopback interface for BGP EVPN peering.

```
OS10(config)# interface loopback1
OS10(config-if-lo-1)# no shutdown
OS10(config-if-lo-1)# ip address 172.202.0.1/32
OS10(config-if-lo-1)# exit
```

### 5. Configure BGP EVPN peer sessions.

```
OS10(config)# router bgp 102
OS10(config-router-bgp-102)# neighbor 172.16.0.1
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# remote-as 99
OS10(config-router-neighbor)# send-community extended
```

```

OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-102)# neighbor 172.17.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 99
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-102)# neighbor 172.18.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-102)# neighbor 172.19.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

```

## Verify VXLAN with BGP EVPN — Multiple AS topology.

### 1. Verify virtual network configurations.

```

LEAF1# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
Virtual Network: 10000
Members:
  VLAN 100: port-channel10, port-channel1000
VxLAN Virtual Network Identifier: 10000
Source Interface: loopback0(192.168.1.1)
Remote-VTEPs (flood-list): 192.168.2.1(CP)

Virtual Network: 20000
Members:
  Untagged: port-channel20
VLAN 200: port-channel1000
VxLAN Virtual Network Identifier: 20000
Source Interface: loopback0(192.168.1.1)
Remote-VTEPs (flood-list): 192.168.2.1(CP)
LEAF1#

```

## 2. Verify EVPN configurations and EVPN parameters.

```
LEAF1# show evpn evi

EVI : 10000, State : up
  Bridge-Domain      : Virtual-Network 10000, VNI 10000
  Route-Distinguisher : 1:192.168.1.1:10000
  Route-Targets       : 0:99:10000 both, 0:100:10000 import
  Inclusive Multicast : 192.168.2.1
  IRB                  : Enabled(tenant1)

EVI : 20000, State : up
  Bridge-Domain      : Virtual-Network 20000, VNI 20000
  Route-Distinguisher : 1:192.168.1.1:20000
  Route-Targets       : 0:99:10000 both, 0:100:10000 import
  Inclusive Multicast : 192.168.2.1
  IRB                  : Enabled(tenant1)
LEAF1#
```

## 3. Verify BGP EVPN neighborship between leaf and spine nodes.

```
LEAF1# show ip bgp l2vpn evpn summary
BGP router identifier 172.16.0.1 local AS number 99
Neighbor      AS      MsgRcvd  MsgSent  Up/Down  State/Pfx
172.201.0.1   101  1132    1116    13:29:00 27
172.202.0.1   102  1131    1118    13:29:02 28
LEAF1#
```

## 4. Check connectivity between host A and host B.

```
root@HOST-A:~# ping 10.2.0.10 -c 5
PING 10.2.0.10 (10.2.0.10) 56(84) bytes of data.
64 bytes from 10.2.0.10: icmp_seq=1 ttl=63 time=0.824 ms
64 bytes from 10.2.0.10: icmp_seq=2 ttl=63 time=0.847 ms
64 bytes from 10.2.0.10: icmp_seq=3 ttl=63 time=0.835 ms
64 bytes from 10.2.0.10: icmp_seq=4 ttl=63 time=0.944 ms
64 bytes from 10.2.0.10: icmp_seq=5 ttl=63 time=0.806 ms

--- 10.2.0.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4078ms
rtt min/avg/max/mdev = 0.806/0.851/0.944/0.051 ms
root@HOST-A:~#
```

## 5. Check connectivity between host A and host C.

```
root@HOST-A:~# ping 10.1.0.20 -c 5
PING 10.1.0.20 (10.1.0.20) 56(84) bytes of data.
64 bytes from 10.1.0.20: icmp_seq=1 ttl=64 time=0.741 ms
64 bytes from 10.1.0.20: icmp_seq=2 ttl=64 time=0.737 ms
64 bytes from 10.1.0.20: icmp_seq=3 ttl=64 time=0.772 ms
64 bytes from 10.1.0.20: icmp_seq=4 ttl=64 time=0.799 ms
64 bytes from 10.1.0.20: icmp_seq=5 ttl=64 time=0.866 ms

--- 10.1.0.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4061ms
rtt min/avg/max/mdev = 0.737/0.783/0.866/0.047 ms
root@HOST-A:~#
```

## 6. Check connectivity between host A and host D.

```
root@HOST-A:~# ping 10.2.0.20 -c 5
PING 10.2.0.20 (10.2.0.20) 56(84) bytes of data.
64 bytes from 10.2.0.20: icmp_seq=1 ttl=63 time=0.707 ms
64 bytes from 10.2.0.20: icmp_seq=2 ttl=63 time=0.671 ms
64 bytes from 10.2.0.20: icmp_seq=3 ttl=63 time=0.687 ms
64 bytes from 10.2.0.20: icmp_seq=4 ttl=63 time=0.640 ms
64 bytes from 10.2.0.20: icmp_seq=5 ttl=63 time=0.644 ms

--- 10.2.0.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4089ms
```

```
rtt min/avg/max/mdev = 0.640/0.669/0.707/0.041 ms
root@HOST-A:~#
```

**NOTE:** Follow Steps 1 to 6 to check ping connectivity between combinations of other hosts, and between hosts through different virtual-network IP addresses.

## Example: VXLAN BGP EVPN — Centralized L3 gateway with asymmetric IRB

The following VXLAN with BGP EVPN example uses a centralized Layer 3 gateway to perform virtual-network routing. It is based on the sample configuration in [Example: VXLAN BGP EVPN — Multiple AS topology](#).

In the VXLAN BGP EVPN multiple AS topology, all VTEPs are configured to perform distributed L3 gateway routing, in which each VTEP routes VXLAN traffic. Routing decisions are made by ingress VTEPs.

However, in a multi-tenant network, some VTEPs may operate only in Layer 2 VXLAN mode and perform only Layer 2 functions. In this case, configure routing for Layer 2 VTEPs on one Layer 3 VTEP that supports Layer 3 VXLAN functionality. The Layer 2 VXLAN-capable VTEPs are connected with the centralized Layer 3 gateway either directly or through an IP underlay fabric. Any ingress routing traffic on a Layer 2 VTEP is switched to the Layer 3 centralized gateway. All routing decisions are made by the centralized gateway to forward VXLAN traffic to the destination Layer 2 VTEP.

The following centralized L3 gateway example for VXLAN BGP EVPN uses a Clos leaf-spine topology. In this example:

- VTEP 1 and VTEP 2 in VLT 1 operate as a L2 gateway.
- VTEP 3 and VTEP 4 in VLT 2 operate as a centralized L3 gateway.
- Host A and Host B are connected to the L2 gateway. The L2 gateway is connected to a centralized L3 gateway through an IP underlay fabric.
- You must configure the IP address and anycast IP address of the virtual networks in the centralized L3 gateway VTEP. It is not necessary to configure these addresses in the L2 gateway VTEPs.

Routing for tenant L3 traffic is not performed on the L2 VTEPs. The L2 VTEPs forward tenant traffic to the centralized L3 gateway in VLT 2. The L3 gateway routes traffic between L2 tenant segments.

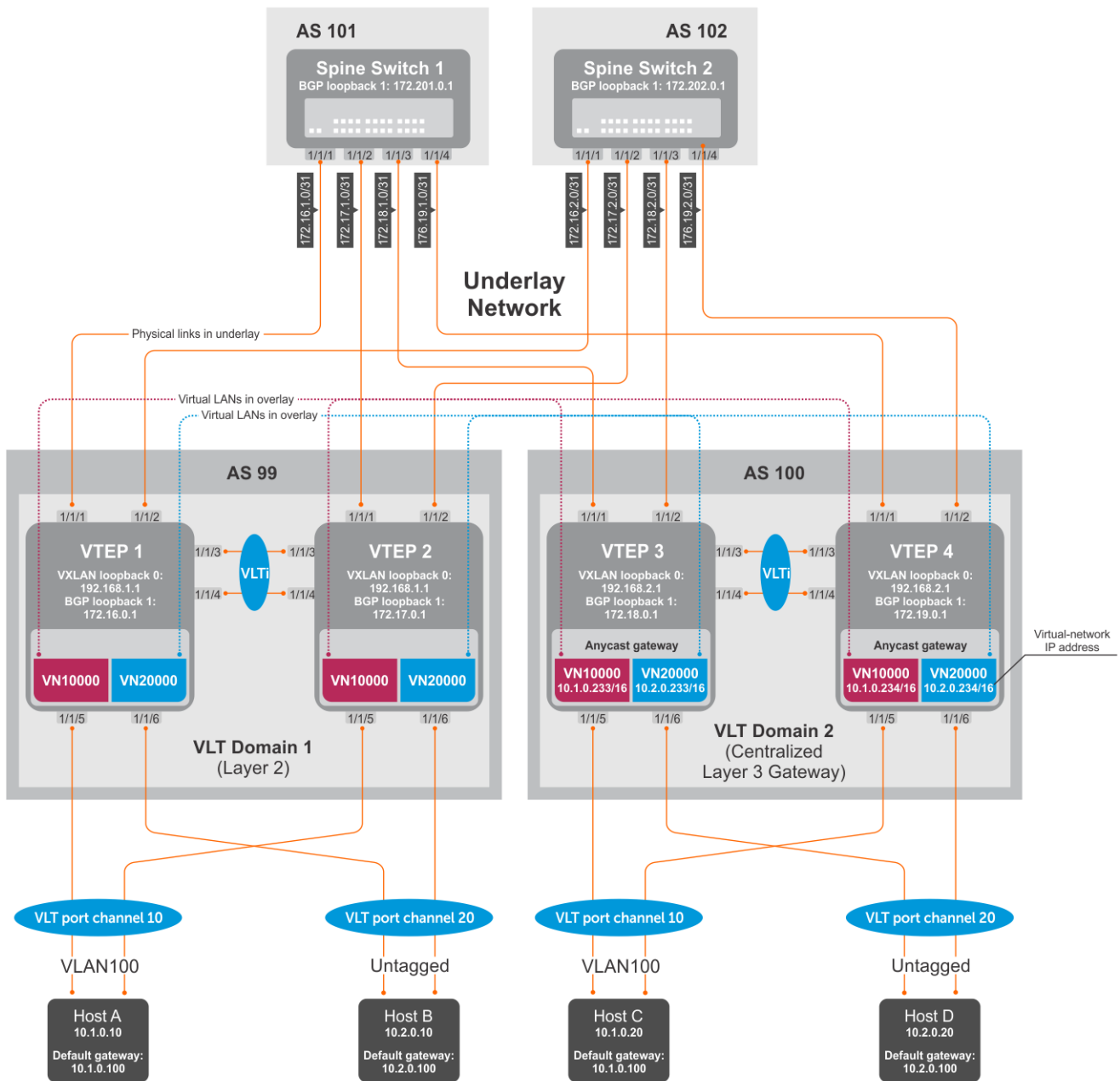


Figure 7. VXLAN BGP EVPN with centralized L3 gateway

- NOTE:** This centralized L3 gateway example for VXLAN BGP EVPN uses the same configuration steps as in [Example: VXLAN BGP EVPN — Multiple AS topology](#). Configure each spine and leaf switch as in the Multiple AS topology example, except:
- Because VTEPs 1 and 2 operate only in Layer 2 VXLAN mode, do not configure **IP switching in the overlay network**. This step consists of configuring virtual network interfaces with IP addresses, anycast IP addresses, and anycast gateway MAC addresses.
  - Configure **IP switching in the overlay network** only on VTEPs 3 and 4.

## VTEP 3 Leaf Switch

### 1. Configure IP switching in the overlay network.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.233/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.233/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## VTEP 4 Leaf Switch

### 1. Configure IP switching in overlay network.

#### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

#### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

#### Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.234/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit

OS10(config)# interface virtual-network20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.234/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

## Example: VXLAN BGP EVPN — Border leaf gateway with asymmetric IRB

This VXLAN BGP EVPN example shows how to transmit VXLAN traffic to an external network. Traffic from a tenant host that is destined to the Internet is transmitted to a border leaf gateway over L3 VTEPs and an IP underlay fabric.

**NOTE:** After VXLAN decapsulation, routing between virtual networks and tenant VLANs is supported only on the S4200-ON series and S5200-ON series due to NPU capability. On other Dell switches that support VXLAN routing, such as

S4048T-ON, S6010-ON, and the S4100-ON series, routing after decapsulation is performed only between virtual networks. You can connect an egress virtual network to a VLAN in an external router, which connects to the external network.

In the following example, VLT domain 1 is a VLT VTEP. VLT domain 2 is the border leaf VLT VTEP pair. All virtual networks in the data center network are configured in all VTEPs with virtual-network IP and anycast IP gateway addresses.

Configure a dedicated virtual network for sending VXLAN traffic to an external network on all VTEPs. Configure the anycast L3 gateway for the dedicated virtual network only on the border leaf VTEP pair in VLT domain 2. For asymmetric IRB, configure a static default route on all VTEPs, except the border leaf VTEPs. This allows traffic destined to an external network to be transmitted to the anycast L3 address of the dedicated virtual network on the border leaf VTEP. A different static route is configured on the border leaf VTEP. Using this second static route, traffic to an external network is transmitted on an egress VLAN to a WAN router or an Internet address.

When VLT domain 1 receives traffic destined to an external network, the traffic is routed to the dedicated virtual network in the ingress VTEP and sent to the border leaf VTEP. On the border leaf VTEP, the traffic is routed to the VLAN to which an external WAN router is connected or directly connected to the Internet. Similarly, any traffic destined to a VXLAN virtual network that is received on the border leaf VTEP is routed to the destination virtual network.

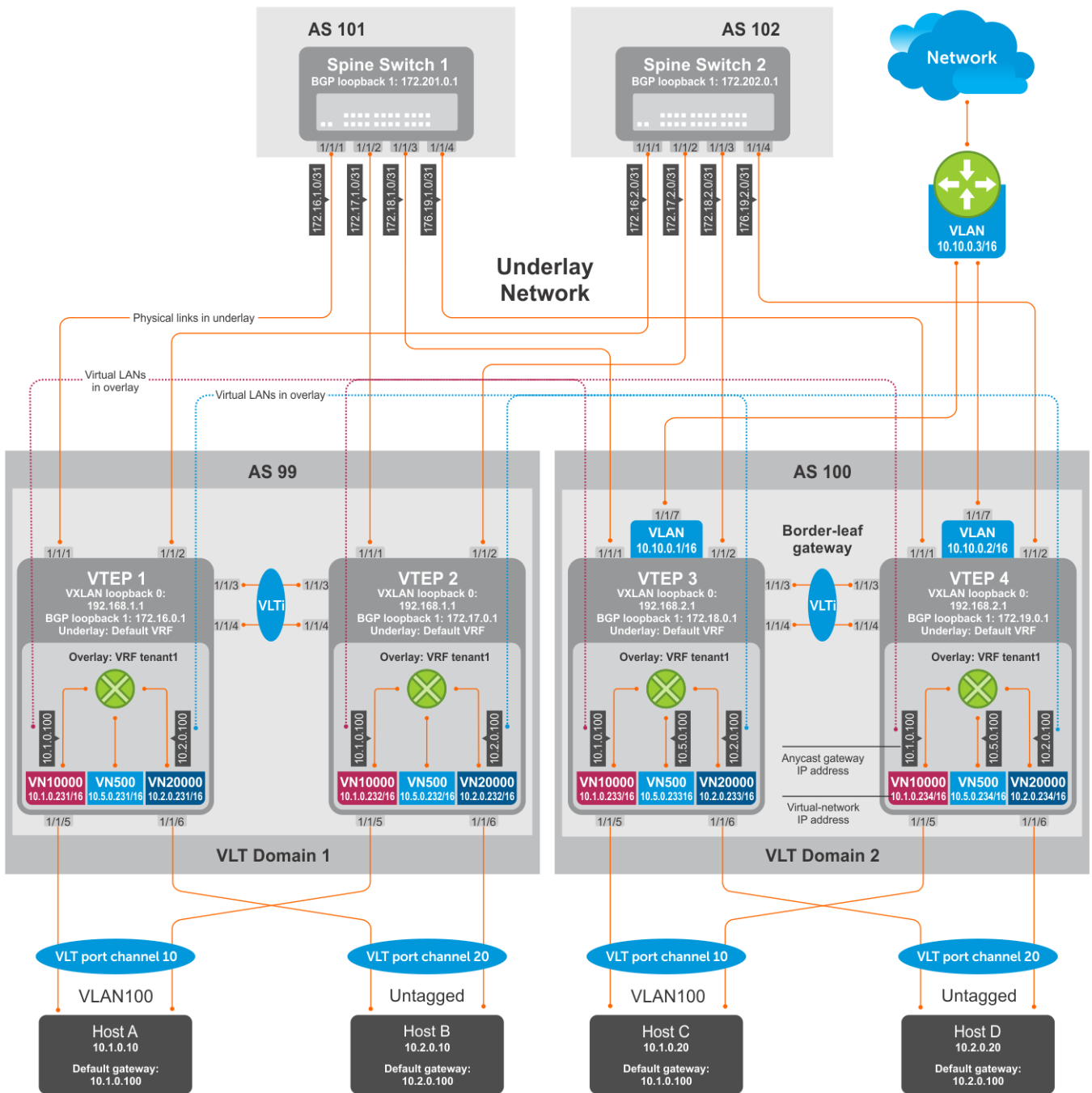


Figure 8. VXLAN BGP EVPN with border leaf gateway

**NOTE:** This border leaf gateway example for VXLAN BGP EVPN uses the same configuration steps as in [Example: VXLAN BGP EVPN — Multiple AS topology](#). Configure each spine and leaf switch as in the Multiple AS topology example and add the following additional configuration steps on each VTEP.

## VTEP 1 Leaf Switch

### 1. Configure a dedicated VXLAN virtual network.

```
OS10(config)# virtual-network 500
OS10(config-vn-500)# vxlan-vni 500
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

## 2. Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 500
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.5.0.231/16
```

## 3. Configure a static route for outbound traffic sent to the anycast MAC address of the dedicated virtual network.

```
OS10(config)#ip route 0.0.0.0/0 10.5.0.100
```

## VTEP 2 Leaf Switch

### 1. Configure a dedicated VXLAN virtual network.

```
OS10(config)# virtual-network 500
OS10(config-vn-500)# vxlan-vni 500
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

### 2. Configure routing on the virtual networks.

```
OS10(config)# interface virtual-network 500
OS10(conf-if-vn-10000)# ip vrf forwarding tenant2
OS10(conf-if-vn-10000)# ip address 10.5.0.232/16
```

### 3. Configure a static route for outbound traffic sent to the anycast MAC address of the dedicated virtual network.

```
OS10(config)#ip route 0.0.0.0/0 10.5.0.100
```

## VTEP 3 Leaf Switch

### 1. Configure a dedicated VXLAN virtual network.

```
OS10(config)# virtual-network 500
OS10(config-vn-500)# vxlan-vni 500
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

### 2. Configure an anycast gateway MAC address on the border leaf VTEP. This MAC address must be different from the anycast gateway MAC address configured on non-border-leaf VTEPs.

```
OS10(config)# ip virtual-router mac-address 00:02:02:02:02:02
```

### 3. Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 500
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.5.0.233/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.5.0.100
OS10(conf-if-vn-10000)# no shutdown
OS10(conf-if-vn-10000)# exit
```

### 4. Configure externally connected VLAN.

```
OS10(conf)#interface vlan 200
OS10(conf-if-vlan)#ip address 10.10.0.1/16
OS10(conf-if-vlan)#no shutdown
OS10(conf-if-vlan)#exit
```

```
OS10(conf)#interface ethernet 1/1/7
switchport mode trunk
switchport trunk allowed vlan 200
```

## 5. Configure a static route for outbound traffic sent to VLAN 200.

```
OS10(config)#ip route 0.0.0.0/0 10.10.0.3
```

## VTEP 4 Leaf Switch

### 1. Configure a dedicated VXLAN virtual network.

```
OS10(config)# virtual-network 500
OS10(config-vn-500)# vxlan-vni 500
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

### 2. Configure an anycast gateway MAC address on the border leaf VTEP. This MAC address must be different from the anycast gateway MAC address configured on non-border-leaf VTEPs.

```
OS10(config)# ip virtual-router mac-address 00:02:02:02:02:02
```

### 3. Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 500
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.5.0.234/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.5.0.100
OS10(conf-if-vn-10000)# no shutdown
OS10(conf-if-vn-10000)# exit
```

### 4. Configure an externally connected VLAN.

```
OS10(conf)#interface vlan 200
OS10(conf-if-vlan)#ip address 10.10.0.2/16
OS10(conf-if-vlan)#no shutdown
OS10(conf-if-vlan)#exit
```

```
OS10(conf)#interface ethernet 1/1/7
switchport mode trunk
switchport trunk allowed vlan 200
```

### 5. Configure a static route for outbound traffic sent to VLAN 200.

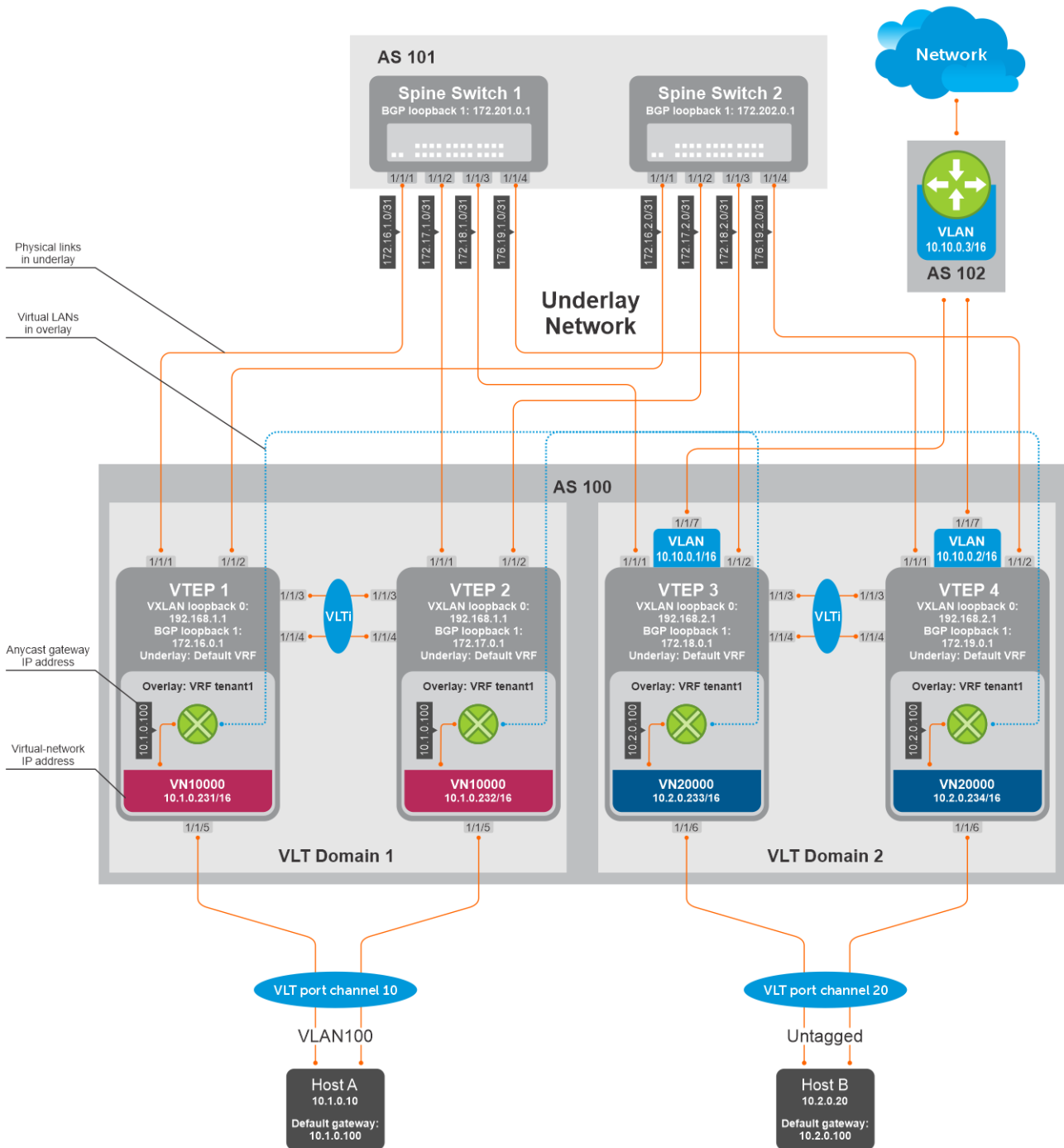
```
OS10(config)#ip route 0.0.0.0/0 10.10.0.3
```

## Example: VXLAN BGP EVPN—Symmetric IRB

The following VXLAN with BGP EVPN example uses a Clos leaf-spine topology to show how to set up an end-to-end VXLAN with symmetric IRB. eBGP is used to exchange IP routes in the IP underlay network, and EVPN routes in the VXLAN overlay network. All spine nodes are in one autonomous system—AS 101. All leaf nodes are in another autonomous system—AS 100.

- On VTEPs 1 and 2, access ports are assigned to the virtual network using a switch-scoped VLAN. EVPN for the overlay VXLAN is configured using auto-EVI mode.
- On VTEPs 3 and 4, access ports are assigned to the virtual network using a port-scoped VLAN. The EVPN instance for the overlay VXLAN is configured using manual configuration mode. The RD and RT are configured using auto mode.
- On all VTEPs, symmetric IRB is configured in EVPN mode using a unique, dedicated VXLAN VNI and EVPN RD and RT values for each tenant VRF.
- The VLAN to an external network is configured only on VTEPs 3 and 4 in the VLT domain that serves as the border leaf gateway.

**NOTE:** In asymmetric IRB, you must configure all destination virtual-network subnets on each VTEP. Symmetric IRB simplifies the VXLAN intersubnet configuration by reducing the number of required VNI configurations. In this example, VLT domain 1 requires only VNI subnet 10.1.0.0/16; VLT domain 2 requires only VNI subnet 10.2.0.0/16. Symmetric IRB facilitates the scaling of VXLAN virtual networks.



## VTEP 1 Leaf Switch

1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# exit
```

2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

### 4. Assign VLAN member interfaces to the virtual network.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
```

### 5. Configure access ports as VLAN members for a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# switchport trunk allowed vlan 100
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit

OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# no switchport
OS10(config-if-eth1/1/5)# exit
```

### 6. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.16.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.16.2.0/31
OS10(config-if-eth1/1/2)# exit
```

### 7. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.16.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(config-router-bgp-af)# redistribute connected
OS10(config-router-bgp-af)# exit
```

### 8. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.16.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.16.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
```

```
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 9. Configure a Loopback interface for BGP EVPN peering different from the VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(config-if-lo-1)# no shutdown
OS10(config-if-lo-1)# ip address 172.16.0.1/32
OS10(config-if-lo-1)# exit
```

### 10. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
```

```
OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 11. Configure EVPN for the VXLAN virtual network.

Configure the EVPN instance, RD, and RT using auto-EVI mode.

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# exit
```

### 12. Configure VLT.

#### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.0/31
OS10(config-if-vl-4000)# exit
```

#### Configure the VLT port channel.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# vlt-port-channel 10
OS10(config-if-po-10)# exit
```

#### Configure the VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
```

```
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

### Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.1
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(config-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel10
OS10(config-uplink-state-group-1)# exit
```

### Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.1
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 13. Configure IP routing in the overlay network.

### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(config-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.231/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit
```

## 14. Configure symmetric IRB.

In EVPN mode, configure the router MAC used by remote VTEPs as the destination address in VXLAN encapsulated packets sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:05
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# vni 3000
OS10(config-evpn-vrf-tenant1)# route-target 65535:30000 both
OS10(config-evpn-vrf-tenant1)# exit
OS10(config-evpn)# exit
OS10(config)#
```

## 15. Configure advertisement of connected networks through EVPN type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 connected
OS10(config-evpn-vrf-tenant1)# exit
```

## VTEP 2 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using the same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn)# exit
```

### 4. Assign VLAN member interfaces to the virtual network.

Use a switch-scoped VLAN-to-VNI mapping:

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# no shutdown
OS10(config-if-vl-100)# exit
```

### 5. Configure access ports as VLAN members for a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
OS10(conf-if-po-10)# switchport trunk allowed vlan 100
OS10(conf-if-po-10)# no switchport access vlan
OS10(conf-if-po-10)# exit
```

```
OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# no switchport
OS10(conf-if-eth1/1/5)# exit
```

### 6. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.17.1.0/31
OS10(conf-if-eth1/1/1)# exit
```

```
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
```

```
OS10(config-if-eth1/1/2)# ip address 172.17.2.0/31
OS10(config-if-eth1/1/2)# exit
```

## 7. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.17.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

## 8. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.17.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.17.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 9. Configure a Loopback interface for BGP EVPN peering different from VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.17.0.1/32
OS10(conf-if-lo-1)# exit
```

## 10. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-bgp-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 11. Configure EVPN for the VXLAN virtual network.

Configure the EVPN instance, RD, and RT using auto-EVI mode.

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# exit
```

## 12. Configure VLT.

Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.1/31
OS10(config-if-vl-4000)# exit
```

Configure the VLT port channel.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit
```

Configure VLTi member links.

```
OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.2
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(conf-vlt-1)# exit
```

Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# exit
```

Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.0
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 13. Configure IP routing in overlay network.

Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(config-if-vn-10000)# ip vrf forwarding tenant1
OS10(config-if-vn-10000)# ip address 10.1.0.232/16
OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(config-if-vn-10000)# no shutdown
OS10(config-if-vn-10000)# exit
```

### 14. Configure symmetric IRB.

In EVPN mode, configure the router MAC used by remote VTEPs as the destination address in VXLAN encapsulated packets sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:05
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# vni 3000
OS10(config-evpn-vrf-tenant1)# route-target 65535:30000 both
OS10(config-evpn-vrf-tenant1)# exit
OS10(config-evpn)# exit
OS10(config)#
```

### 15. Configure advertisement of connected networks through EVPN type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 connected
OS10(config-evpn-vrf-tenant1)# exit
```

## VTEP 3 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.2.1/32
OS10(config-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Configure unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

### 5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit
```

```
OS10(config)# interface ethernet1/1/6
OS10(conf-if-eth1/1/6)# no shutdown
OS10(conf-if-eth1/1/6)# channel-group 20 mode active
OS10(conf-if-eth1/1/6)# no switchport
OS10(conf-if-eth1/1/6)# exit
```

## 6. Add the access ports to the virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn-20000)# exit
```

## 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ip address 172.18.1.0/31
OS10(conf-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/2)# ip address 172.18.2.0/31
OS10(conf-if-eth1/1/2)# exit
```

## 8. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.18.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

## 9. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.18.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.18.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 10. Configure a Loopback interface for BGP EVPN peering different from VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.18.0.1/32
OS10(conf-if-lo-1)# exit
```

## 11. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
```

```

OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit

```

## 12. Configure EVPN for the VXLAN virtual network.

Configure the EVPN instance in manual configuration mode, and RD and RT configuration in auto mode.

```

OS10(config)# evpn
OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd auto
OS10(config-evpn-evi-20000)# route-target auto
OS10(config-evpn-evi-20000)# exit
OS10(config-evpn)# exit

```

## 13. Configure VLT.

### Configure a VLTi VLAN for the virtual network.

```

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit

```

### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```

OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.10/31
OS10(config-if-vl-4000)# exit

```

### Configure the VLT port channel.

```

OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit

```

### Configure VLTi member links.

```

OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit

OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit

```

### Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.3
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(conf-vlt-1)# exit
```

### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

### Configure iBGP IPv4 peering between VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.11
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

## 14. Configure IP routing in the overlay network.

### Create the tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.233/16
OS10(conf-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-if-vn-20000)# exit
```

## 15. Configure symmetric IRB.

In EVPN mode, configure the router MAC used by remote VTEPs as the destination address in VXLAN encapsulated packets sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:06
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# vni 3000
OS10(config-evpn-vrf-tenant1)# route-target 65535:30000 both
OS10(config-evpn-vrf-tenant1)# exit
OS10(config-evpn)# exit
OS10(config)#
```

## 16. Configure an externally connected VLAN.

```
OS10(conf)# interface vlan 200
OS10(conf-if-vlan)# ip vrf forwarding tenant1
OS10(conf-if-vlan)# ip address 10.10.0.1/16
OS10(conf-if-vlan)# no shutdown
OS10(conf-if-vlan)# exit

OS10(conf)# interface ethernet 1/1/7
```

```
OS10(config-if-eth1/1/7)# switchport mode trunk
OS10(config-if-eth1/1/7)# switchport trunk allowed vlan 200
```

### 17. Configure advertisement of the connected networks via EVPN Type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 connected
OS10(config-evpn-vrf-tenant1)# exit
```

### 18. Configure BGP session with external router on the border-leaf VTEPs.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# neighbor 10.10.0.3
OS10(config-router-vrf-neighbor)# remote-as 102
OS10(config-router-vrf-neighbor)# no shutdown
OS10(config-router-vrf-neighbor)# end
```

### 19. Import external routes in to EVPN on the border-leaf switches.

External routes for WAN connectivity and other appliances can be imported in to a VXLAN pod using the following configuration on the border-leaf router.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 bgp
OS10(config-evpn-vrf-tenant1)# end
```

### 20. Export BGP EVPN routes out of border-leaf switch to external devices.

For interpod connectivity, use the following configuration to export the BGP EVPN routes of a VXLAN pod from the border-leaf router.

With connected routes of virtual networks present in an individual VTEP advertised as type-5 routes, the border-leaf router has information about all the virtual networks present in the pod.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-vrf-af)# redistribute l2vpn evpn
OS10(configure-router-bgpv4-vrf-af)# end
```

The `redistribute l2vpn evpn` command redistributes both type-2 mac-ip (/32 routes) and type-5 routes (subnet routes). Use the `route-map` command to filter type-2 mac-ip (/32 routes) and redistribute only the type-5 routes.

```
OS10(config)# ip prefix-list deny_v4_host_routes seq 10 deny 0.0.0.0/0 ge 32 le 32
OS10(config)# ip prefix-list deny_v4_host_routes seq 20 permit 0.0.0.0/0 le 31
OS10(config)# route-map deny_v4_host_routes permit 10
OS10(config-route-map)# match ip address prefix-list deny_v4_host_routes
OS10(config-route-map)# exit

OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-vrf-af)# redistribute l2vpn evpn route-map
deny_v4_host_routes
OS10(configure-router-bgpv4-vrf-af)# end
```

Use the following configuration to advertise the local connected routes on the border-leaf switches to external device:

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-vrf-af)# redistribute connected
OS10(configure-router-bgpv4-vrf-af)# end
```

## VTEP 4 Leaf Switch

### 1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.2.1/32
OS10(config-if-lo-0)# exit
```

### 2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

### 3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

### 4. Configure the unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

### 5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit

OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# no switchport
OS10(config-if-eth1/1/6)# exit
```

### 6. Add the access ports to the virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn)# exit
```

### 7. Configure upstream network-facing ports.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ip address 172.19.1.0/31
OS10(config-if-eth1/1/1)# exit

OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ip address 172.19.2.0/31
OS10(config-if-eth1/1/2)# exit
```

### 8. Configure eBGP.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# router-id 172.19.0.1
OS10(config-router-bgp-100)# address-family ipv4 unicast
```

```
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

### 9. Configure eBGP for the IPv4 point-to-point peering.

```
OS10(config-router-bgp-100)# neighbor 172.19.1.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.19.2.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 10. Configure a Loopback interface for BGP EVPN peering different from the VLT peer IP address.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.19.0.1/32
OS10(conf-if-lo-1)# exit
```

### 11. Configure BGP EVPN peering.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.201.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit

OS10(config-router-bgp-100)# neighbor 172.202.0.1
OS10(config-router-neighbor)# remote-as 101
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# send-community extended
OS10(config-router-neighbor)# update-source loopback1
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# no activate
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# activate
OS10(config-router-bgp-neighbor-af)# allowas-in 1
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 12. Configure EVPN for the VXLAN virtual network.

Configure the EVPN instance manual configuration mode, and RD, and RT configuration in auto mode.

```
OS10(config)# evpn
OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd auto
```

```
OS10(config-evpn-evi-20000)# route-target auto
OS10(config-evpn-evi-20000)# exit
OS10(config-evpn)# exit
```

### 13. Configure VLT.

#### Configure a VLTi VLAN for the virtual network.

```
OS10(config)# virtual-network 20000
OS10(conf-vn-20000)# vlti-vlan 200
OS10(conf-vn-20000)# exit
```

#### Configure a dedicated L3 underlay path to reach the VLT Peer in case of a network failure.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ip address 172.16.250.11/31
OS10(config-if-vl-4000)# exit
```

#### Configure the VLT port channel.

```
OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit
```

#### Configure VLTi member links.

```
OOS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit
```

```
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

#### Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.4
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(conf-vlt-1)# exit
```

#### Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

#### Configure iBGP IPv4 peering between the VLT peers.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 172.16.250.10
OS10(config-router-neighbor)# remote-as 100
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

### 14. Configure IP routing in the overlay network.

#### Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

### Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

### Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 20000
OS10(config-if-vn-20000)# ip vrf forwarding tenant1
OS10(config-if-vn-20000)# ip address 10.2.0.234/16
OS10(config-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(config-if-vn-20000)# no shutdown
OS10(config-if-vn-20000)# exit
```

### 15. Configure symmetric IRB.

In EVPN mode, configure the router MAC used by remote VTEPs as the destination address in VXLAN encapsulated packets sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:06
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# vni 3000
OS10(config-evpn-vrf-tenant1)# route-target 65535:30000 both
OS10(config-evpn-vrf-tenant1)# exit
OS10(config-evpn)# exit
OS10(config)#
```

### 16. Configure an externally connected VLAN.

```
OS10(config)# interface vlan 200
OS10(config-if-vlan)# ip vrf forwarding tenant1
OS10(config-if-vlan)# ip address 10.10.0.2/16
OS10(config-if-vlan)# no shutdown
OS10(config-if-vlan)# exit

OS10(config)# interface ethernet 1/1/7
OS10(config-if-eth1/1/7)# switchport mode trunk
OS10(config-if-eth1/1/7)# switchport trunk allowed vlan 200
```

### 17. Configure advertisement of the connected networks via EVPN Type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 connected
OS10(config-evpn-vrf-tenant1)# exit
```

### 18. Configure BGP session with external router on the border-leaf VTEPs.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# neighbor 10.10.0.3
OS10(config-router-vrf-neighbor)# remote-as 102
OS10(config-router-vrf-neighbor)# no shutdown
OS10(config-router-vrf-neighbor)# end
```

### 19. Import external routes in to EVPN on the border-leaf switches.

External routes for WAN connectivity and other appliances can be imported in to a VXLAN pod using the following configuration on the border-leaf router.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 bgp
OS10(config-evpn-vrf-tenant1)# end
```

### 20. Export BGP EVPN routes out of border-leaf switch to external devices.

For interpod connectivity, use the following configuration to export the BGP EVPN routes of a VXLAN pod from the border-leaf router.

With connected routes of virtual networks present in an individual VTEP advertised as type-5 routes, the border-leaf router has information about all the virtual networks present in the pod.

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-vrf-af)# redistribute l2vpn evpn
OS10(configure-router-bgpv4-vrf-af)# end
```

The redistribute l2vpn evpn command redistributes both type-2 mac-ip (/32 routes) and type-5 routes (subnet routes). Use the route-map command to filter type-2 mac-ip (/32 routes) and redistribute only the type-5 routes.

```
OS10(config)# ip prefix-list deny_v4_host_routes seq 10 deny 0.0.0.0/0 ge 32 le 32
OS10(config)# ip prefix-list deny_v4_host_routes seq 20 permit 0.0.0.0/0 le 31
OS10(config)# route-map deny_v4_host_routes permit 10
OS10(config-route-map)# match ip address prefix-list deny_v4_host_routes
OS10(config-route-map)# exit

OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-vrf-af)# redistribute l2vpn evpn route-map
deny_v4_host_routes
OS10(configure-router-bgpv4-vrf-af)# end
```

Use the following configuration to advertise the local connected routes on the border-leaf switches to external device:

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# vrf tenant1
OS10(config-router-bgp-100-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-vrf-af)# redistribute connected
OS10(configure-router-bgpv4-vrf-af)# end
```

## Spine Switch 1

### 1. Configure downstream ports on underlay links to the leaf switches.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# ip address 172.16.1.1/31
OS10(conf-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/2)# ip address 172.17.1.1/31
OS10(conf-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# ip address 172.18.1.1/31
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# ip address 172.19.1.1/31
OS10(conf-if-eth1/1/4)# exit
```

### 2. Configure eBGP.

```
OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.201.0.1
OS10(config-router-bgp-101)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit
```

### 3. Configure eBGP IPv4 peer sessions on the P2P links.

```
OS10(conf-router-bgp-101)# neighbor 172.16.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.1.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit
OS10(conf-router-bgp-101)# exit
```

### 4. Configure a Loopback interface for BGP EVPN peering.

```
OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.201.0.1/32
OS10(conf-if-lo-1)# exit
```

### 5. Configure BGP EVPN peer sessions.

```
OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# neighbor 172.16.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit
```

```

OS10(conf-router-bgp-101)# neighbor 172.18.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

```

## Spine Switch 2

### 1. Configure downstream ports on the underlay links to the leaf switches.

```

OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# ip address 172.16.2.1/31
OS10(conf-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/2)# ip address 172.17.2.1/31
OS10(conf-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# ip address 172.18.2.1/31
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# ip address 172.19.2.1/31
OS10(conf-if-eth1/1/4)# exit

```

### 2. Configure eBGP.

```

OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.202.0.1
OS10(config-router-bgp-101)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit

```

### 3. Configure eBGP IPv4 peer sessions on the P2P links.

```

OS10(conf-router-bgp-101)# neighbor 172.16.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection

```

```

OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.2.0
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# exit
OS10(conf-router-bgp-101)# exit

```

#### 4. Configure a Loopback interface for BGP EVPN peering.

```

OS10(config)# interface loopback1
OS10(conf-if-lo-1)# no shutdown
OS10(conf-if-lo-1)# ip address 172.202.0.1/32
OS10(conf-if-lo-1)# exit

```

#### 5. Configure BGP EVPN peer sessions.

```

OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# neighbor 172.16.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.17.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.18.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown

```

```

OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

OS10(conf-router-bgp-101)# neighbor 172.19.0.1
OS10(conf-router-neighbor)# ebgp-multihop 4
OS10(conf-router-neighbor)# remote-as 100
OS10(conf-router-neighbor)# send-community extended
OS10(conf-router-neighbor)# update-source loopback1
OS10(conf-router-neighbor)# no shutdown
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-neighbor-af)# no activate
OS10(conf-router-neighbor-af)# exit
OS10(conf-router-neighbor)# address-family l2vpn evpn
OS10(conf-router-neighbor-af)# no sender-side-loop-detection
OS10(conf-router-neighbor-af)# activate
OS10(conf-router-neighbor-af)# exit

```

## Verify VXLAN with BGP EVPN configuration.

### 1. Verify virtual network configurations.

```

LEAF1# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD - Unknown-Unicast-Drop
Virtual Network: 10000
Members:
  VLAN 100: port-channel10, port-channel1000
VxLAN Virtual Network Identifier: 10000
Source Interface: loopback0(192.168.1.1)
Remote-VTEPs (flood-list):
LEAF1#

```

### 2. Verify EVPN configurations and EVPN parameters.

```

LEAF1# show evpn evi

EVI : 10000, State : up
  Bridge-Domain      : Virtual-Network 10000, VNI 10000
  Route-Distinguisher : 1:192.168.1.1:10000(auto)
  Route-Targets      : 0:100:268445456(auto) both
  Inclusive Multicast :
  IRB                : Enabled(tenant1)

LEAF1#

```

```

LEAF1# show evpn vrf l3-vni

VRF : tenant1, State : up
  L3-VNI      : 3000
  Route-Distinguisher : 1:192.168.1.1:3000(auto)
  Route-Targets : 0:65535:30000 both
  Remote VTEP  : 192.168.2.1

LEAF1#

```

### 3. Verify BGP EVPN neighborhood between leaf and spine nodes.

```

LEAF1# show ip bgp l2vpn evpn summary
BGP router identifier 172.16.0.1 local AS number 100
Neighbor      AS      MsgRcvd  MsgSent  Up/Down  State/Pfx
172.201.0.1   101  1132    1116    13:29:00 27
172.202.0.1   101  1131    1118    13:29:02 28
LEAF1#

```

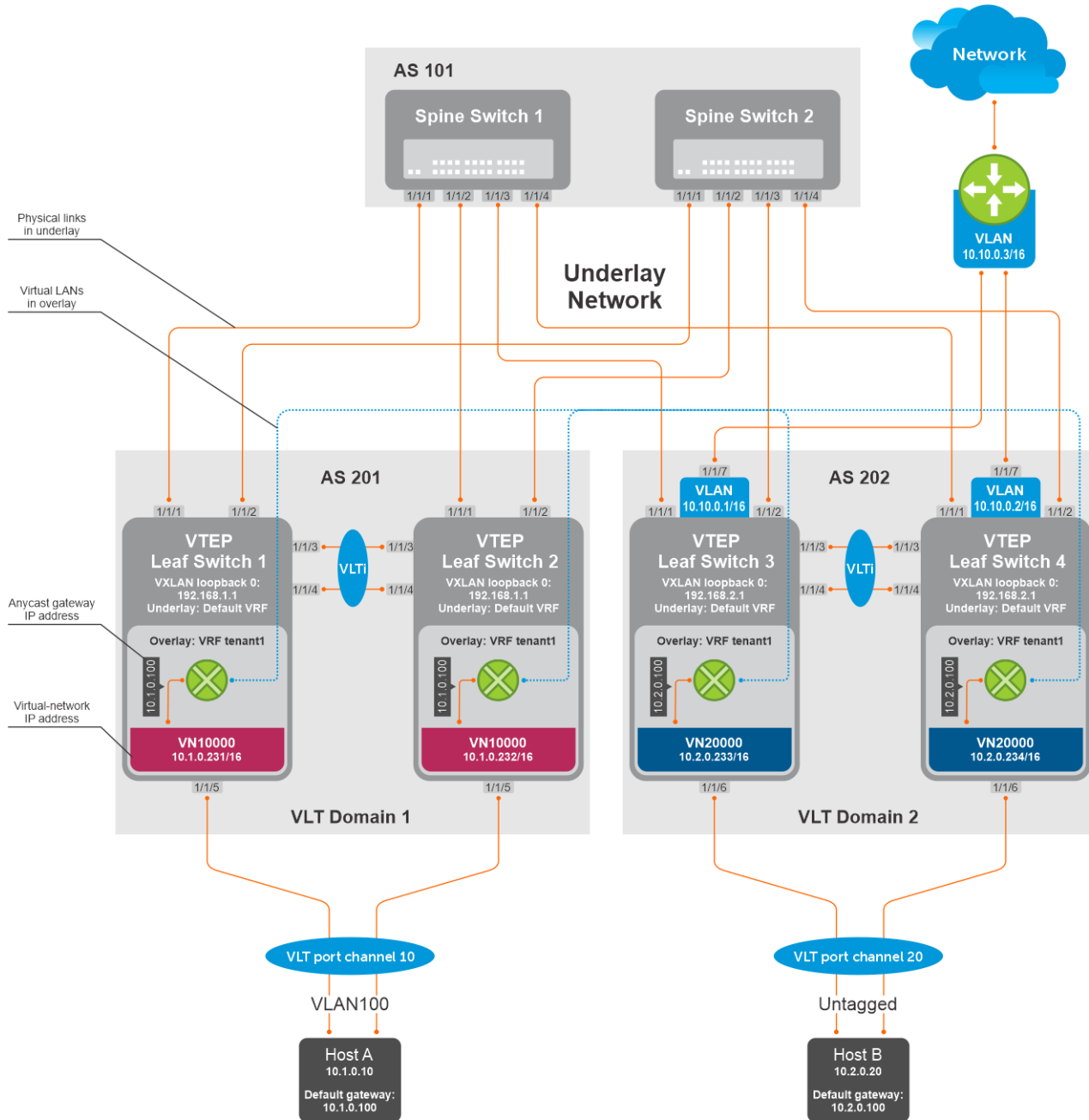
#### 4. Check connectivity between host A and host B.

```
root@HOST-A:~# ping 10.2.0.20 -c 5
PING 10.2.0.10 (10.2.0.10) 56(84) bytes of data.
64 bytes from 10.2.0.10: icmp_seq=1 ttl=63 time=0.824 ms
64 bytes from 10.2.0.10: icmp_seq=2 ttl=63 time=0.847 ms
64 bytes from 10.2.0.10: icmp_seq=3 ttl=63 time=0.835 ms
64 bytes from 10.2.0.10: icmp_seq=4 ttl=63 time=0.944 ms
64 bytes from 10.2.0.10: icmp_seq=5 ttl=63 time=0.806 ms

--- 10.2.0.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4078ms
rtt min/avg/max/mdev = 0.806/0.851/0.944/0.051 ms
root@HOST-A:~#
```

# Example - VXLAN BGP EVPN symmetric IRB with unnumbered BGP peering

The following BGP EVPN example uses a Clos leaf-spine topology with BGP over unnumbered interfaces.



The following explains how the network is configured:

- External BGP (eBGP) over unnumbered interfaces is used to exchange both IPv4 routes and EVPN routes.
- You need not configure IP addresses on links that connect Spine and Leaf switches. BGP Unnumbered peering works without an IP address configuration on Spine-Leaf links.
- The remote AS is autodiscovered from BGP Open messages.
- All VTEPs perform Symmetric IRB routing. All spine nodes are in one autonomous system and each VTEP in the leaf network belongs to different autonomous systems. Both Spine Switch 1 and Spine Switch 2 are in AS 101. For leaf nodes, VLT domain 1 is in AS 201; VLT domain 2 is in AS 202.

- On leaf switches 1 and 2, access ports are assigned to a virtual network using a switch-scoped VLAN. EVPN for the overlay VXLAN is configured using auto-EVI mode.
- On leaf switches 3 and 4, access ports are assigned to a virtual network using a port-scoped VLAN. EVPN for the overlay VXLAN is configured using manual EVI mode with RT and RD values configured in auto mode.
- On all VTEPs, symmetric IRB is configured in EVPN mode using a unique, dedicated VXLAN VNI, and Auto RD and Auto RT values for each tenant VRF.
- On all VTEPs, the `disable-rt-asn` command is used to autoderive the RT that does not include the ASN in the RT value. This allows auto RT to be used even if there are different ASNs for each leaf node.
- The VLAN to an external network is configured only on VTEPs 3 and 4 in the VLT domain that serves as the border leaf gateway.

### Spine Switch 1 configuration

1. Configure downstream ports as unnumbered interfaces. Configure the `ipv6 nd send-ra` command and lower RA intervals. These interfaces are used for BGP unnumbered peering.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/1)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/1)# ipv6 nd send-ra
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/2)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/2)# ipv6 nd send-ra
OS10(config-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# mtu 1650
OS10(config-if-eth1/1/3)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/3)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/3)# ipv6 nd send-ra
OS10(config-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# mtu 1650
OS10(config-if-eth1/1/4)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/4)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/4)# ipv6 nd send-ra
OS10(config-if-eth1/1/4)# exit
```

2. Configure BGP instance with router id.

```
OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.201.0.1
```

3. Configure the BGP unnumbered neighbor on Leaf-facing ports. Use a template to simplify the configuration on multiple interfaces. These neighbors are configured to carry IPv4 address family (default) and L2VPN EVPN address family.

```
OS10(config-router-bgp-101)# template ebgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface ethernet1/1/1
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-101)# neighbor interface ethernet1/1/2
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-101)# neighbor interface ethernet1/1/3
```

```

OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-101)# neighbor interface ethernet1/1/4
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

```

## Spine Switch 2 configuration

1. Configure downstream ports as unnumbered interfaces. Configure the `ipv6 nd send-ra` command and lower RA intervals. These interfaces are used for BGP unnumbered peering.

```

OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ipv6 nd max-ra-interval 4
OS10(conf-if-eth1/1/1)# ipv6 nd min-ra-interval 3
OS10(conf-if-eth1/1/1)# ipv6 nd send-ra
OS10(conf-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/2)# ipv6 nd max-ra-interval 4
OS10(conf-if-eth1/1/2)# ipv6 nd min-ra-interval 3
OS10(conf-if-eth1/1/2)# ipv6 nd send-ra
OS10(conf-if-eth1/1/2)# exit
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# mtu 1650
OS10(conf-if-eth1/1/3)# ipv6 nd max-ra-interval 4
OS10(conf-if-eth1/1/3)# ipv6 nd min-ra-interval 3
OS10(conf-if-eth1/1/3)# ipv6 nd send-ra
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# mtu 1650
OS10(conf-if-eth1/1/4)# ipv6 nd max-ra-interval 4
OS10(conf-if-eth1/1/4)# ipv6 nd min-ra-interval 3
OS10(conf-if-eth1/1/4)# ipv6 nd send-ra
OS10(conf-if-eth1/1/4)# exit

```

2. Configure BGP instance with router id.

```

OS10(config)# router bgp 101
OS10(config-router-bgp-101)# router-id 172.202.0.1

```

3. Configure the BGP unnumbered neighbor on Leaf-facing ports. Use a template to simplify the configuration on multiple interfaces. These neighbors are configured to carry IPv4 address family (default) and L2VPN EVPN address family.

```

OS10(config-router-bgp-101)# template ebgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface ethernet1/1/1
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-101)# neighbor interface ethernet1/1/2
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-101)# neighbor interface ethernet1/1/3
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-101)# neighbor interface ethernet1/1/4

```

```
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

### VTEP Leaf Switch 1 configuration

1. Configure a loopback interface for the VXLAN underlay using the same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(config-if-lo-0)# no shutdown
OS10(config-if-lo-0)# ip address 192.168.1.1/32
OS10(config-if-lo-0)# exit
```

2. Configure the loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-10000)# exit
```

4. Assign VLAN to the virtual network. Use a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# exit
```

5. Configure access ports as VLAN members.

```
OS10(config)# interface port-channel10
OS10(config-if-po-10)# no shutdown
OS10(config-if-po-10)# switchport mode trunk
OS10(config-if-po-10)# switchport trunk allowed vlan 100
OS10(config-if-po-10)# no switchport access vlan
OS10(config-if-po-10)# exit
OS10(config)# interface ethernet1/1/5
OS10(config-if-eth1/1/5)# no shutdown
OS10(config-if-eth1/1/5)# channel-group 10 mode active
OS10(config-if-eth1/1/5)# exit
```

6. Configure upstream network-facing ports as unnumbered interfaces. Configure the `ipv6 nd send-ra` command and lower RA intervals. These interfaces are used for BGP unnumbered peering.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/1)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/1)# ipv6 nd send-ra
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/2)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/2)# ipv6 nd send-ra
OS10(config-if-eth1/1/2)# exit
```

7. Configure BGP instance with router id.

```
OS10(config)# router bgp 201
OS10(config-router-bgp-201)# router-id 172.16.0.1
OS10(config-router-bgp-201)# address-family ipv4 unicast
```

```
OS10(config-router-bgp-af)# redistribute connected
OS10(config-router-bgp-af)# exit
```

8. Configure a BGP unnumbered neighbor over network facing ports. Use a template to simplify the configuration on multiple interfaces. These neighbors are configured to carry IPv4 address family (default) and L2VPN EVPN address family.

```
OS10(config-router-bgp-201)# template ebgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface ethernet1/1/1
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-201)# neighbor interface ethernet1/1/2
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

9. Configure EVPN for the VXLAN virtual network. Configure EVPN instances using auto-EVI mode and disable ASN in the generated RT.

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# exit
```

**i NOTE:** Use the `disable-rt-asn` command to autoderive RT that does not include the ASN in the RT value. This allows auto RT to be used even if the Clos leaf-spine design has separate ASN for each leaf node. Configure this command only when all the VTEPs are OS10 switches.

10. Configure VLT.

- Configure a dedicated Layer 3 forwarding path through the other VLT peer for connectivity even if all spine links go down. This VLAN interface is an unnumbered interface and used for iBGP peering with the other VLT peer.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ipv6 nd max-ra-interval 4
OS10(config-if-vl-4000)# ipv6 nd min-ra-interval 3
OS10(config-if-vl-4000)# ipv6 nd send-ra
OS10(config-if-vl-4000)# exit
```

- Configure the VLT port channel.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit
```

- Configure the VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

- Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.1
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(conf-vlt-1)# exit
```

- Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# exit
```

- Configure iBGP unnumbered peering between VLT peers with both IPv4 and L2VPN EVPN address families.

```
OS10(config)# router bgp 201
OS10(config-router-bgp-201)# template ibgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface vlan4000
OS10(config-router-neighbor)# inherit template ibgp_unified inherit-type ibgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

#### 11. Configure IP routing in the overlay network.

- Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

- Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

- Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.1.0.231/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(conf-if-vn-10000)# no shutdown
OS10(conf-if-vn-10000)# exit
```

#### 12. Configure symmetric IRB.

- In EVPN mode, configure the router MAC address that is used by remote VTEPs as the destination address in VXLAN encapsulated packets that are sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:05
OS10(config-evpn)# vrf tenant1
OS10((config-evpn-vrf-tenant1))# vni 3000
OS10((config-evpn-vrf-tenant1))# route-target auto
OS10((config-evpn-vrf-tenant1))# exit
OS10(config-evpn)# exit
```

#### 13. Configure advertisement of the connected networks through EVPN Type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10((config-evpn-vrf-tenant1))# advertise ipv4 connected
OS10((config-evpn-vrf-tenant1))# exit
```

### VTEP Leaf Switch 2 configuration

1. Configure a loopback interface for the VXLAN underlay using the same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.1.1/32
OS10(conf-if-lo-0)# exit
```

2. Configure the loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 10000
OS10(config-vn-10000)# vxlan-vni 10000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn)# exit
```

4. Assign VLAN member interfaces to the virtual network. Use a switch-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface vlan100
OS10(config-if-vl-100)# virtual-network 10000
OS10(config-if-vl-100)# exit
```

5. Configure access ports as VLAN members.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# no shutdown
OS10(conf-if-po-10)# switchport mode trunk
OS10(conf-if-po-10)# switchport trunk allowed vlan 100
OS10(conf-if-po-10)# no switchport access vlan
OS10(conf-if-po-10)# exit
OS10(config)# interface ethernet1/1/5
OS10(conf-if-eth1/1/5)# no shutdown
OS10(conf-if-eth1/1/5)# channel-group 10 mode active
OS10(conf-if-eth1/1/5)# exit
```

6. Configure upstream network-facing ports as unnumbered interfaces. Configure the `ipv6 nd send-ra` command and lower RA intervals. These interfaces are used for BGP unnumbered peering.

```
OS10(config)# interface ethernet1/1/1
OS10(conf-if-eth1/1/1)# no shutdown
OS10(conf-if-eth1/1/1)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/1)# ipv6 nd max-ra-interval 4
OS10(conf-if-eth1/1/1)# ipv6 nd min-ra-interval 3
OS10(conf-if-eth1/1/1)# ipv6 nd send-ra
OS10(conf-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(conf-if-eth1/1/2)# no shutdown
OS10(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/2)# mtu 1650
OS10(conf-if-eth1/1/2)# ipv6 nd max-ra-interval 4
OS10(conf-if-eth1/1/2)# ipv6 nd min-ra-interval 3
OS10(conf-if-eth1/1/2)# ipv6 nd send-ra
OS10(conf-if-eth1/1/2)# exit
```

7. Configure BGP instance with router id.

```
OS10(config)# router bgp 201
OS10(config-router-bgp-201)# router-id 172.17.0.1
OS10(config-router-bgp-201)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

8. Configure a BGP unnumbered neighbor on network facing ports. Use a template to simplify the configuration on multiple interfaces. These neighbors are configured to carry IPv4 address family (default) and L2VPN EVPN address family.

```
OS10(config-router-bgp-201)# template ebgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface ethernet1/1/1
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

```
OS10(config-router-bgp-201)# neighbor interface ethernet1/1/2
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

9. Configure EVPN for the VXLAN virtual network. Configure the EVPN instances using Auto EVI mode and Disable ASN in the generated RT.

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# exit
```

**i NOTE:** Use the `disable-rt-asn` command to autoderive RT that does not include the ASN in the RT value. This allows auto RT to be used even if the Clos leaf-spine design has separate ASN for each leaf node. Configure this command only when all the VTEPs are OS10 switches.

10. Configure VLT.

- Configure a dedicated Layer 3 forwarding path through the other VLT peer for connectivity even if all spine links go down. This VLAN interface would be unnumbered interface and used for iBGP peering with the other VLT peer.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ipv6 nd max-ra-interval 4
OS10(config-if-vl-4000)# ipv6 nd min-ra-interval 3
OS10(config-if-vl-4000)# ipv6 nd send-ra
OS10(config-if-vl-4000)# exit
```

- Configure the VLT port channel.

```
OS10(config)# interface port-channel10
OS10(conf-if-po-10)# vlt-port-channel 10
OS10(conf-if-po-10)# exit
```

- Configure VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

- Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.2
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ee:ff
OS10(conf-vlt-1)# exit
```

- Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel10
OS10(conf-uplink-state-group-1)# exit
```

- Configure iBGP unnumbered peering between VLT peers with both IPv4 and L2VPN EVPN address families.

```
OS10(config)# router bgp 201
OS10(config-router-bgp-201)# template ibgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface vlan4000
OS10(config-router-neighbor)# inherit template ibgp_unified inherit-type ibgp
```

```
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

11. Configure IP routing in overlay network.

- Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

- Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

- Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 10000
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.1.0.232/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.1.0.100
OS10(conf-if-vn-10000)# no shutdown
OS10(conf-if-vn-10000)# exit
```

12. Configure symmetric IRB. In EVPN mode, configure the router MAC address that is used by remote VTEPs as the destination address in VXLAN encapsulated packets that are sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:05
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-tenant1)# vni 3000
OS10(config-evpn-tenant1)# route-target auto
OS10(config-evpn-tenant1)# exit
OS10(config-evpn)# exit
```

13. Configure advertisement of the connected networks through EVPN Type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-tenant1)# advertise ipv4 connected
OS10(config-evpn-tenant1)# exit
```

### VTEP Leaf Switch 3 configuration

1. Configure a Loopback interface for the VXLAN underlay using same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.2.1/32
OS10(conf-if-lo-0)# exit
```

2. Configure the loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

4. Configure an unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel20
OS10(conf-if-po-20)# no shutdown
```

```

OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit
OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# exit

```

6. Add the access ports to the virtual network.

```

OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn-20000)# exit

```

7. Configure upstream network-facing ports as unnumbered interfaces. Configure the `ipv6 nd send-ra` command and lower RA intervals. These interfaces would be used for BGP unnumbered peering.

```

OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/1)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/1)# ipv6 nd send-ra
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/2)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/2)# ipv6 nd send-ra
OS10(config-if-eth1/1/2)# exit

```

8. Configure BGP instance with router id.

```

OS10(config)# router bgp 202
OS10(config-router-bgp-202)# router-id 172.18.0.1
OS10(config-router-bgp-202)# address-family ipv4 unicast
OS10(config-router-bgp-af)# redistribute connected
OS10(config-router-bgp-af)# exit

```

9. Configure BGP unnumbered neighbor over network facing ports. You can use a template to simplify the configuration on multiple interfaces. These neighbors are configured to carry IPv4 address family (default) and L2VPN EVPN address family.

```

OS10(config-router-bgp-202)# template ebgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface ethernet1/1/1
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor interface ethernet1/1/2
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

```

10. Configure EVPN for the VXLAN virtual network. Configure the EVPN instance in manual configuration mode, and RD and RT configuration in auto mode.

```

OS10(config)# evpn
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd auto
OS10(config-evpn-evi-20000)# route-target auto
OS10(config-evpn-evi-20000)# exit
OS10(config-evpn)# exit

```

**NOTE:** Use the `disable-rt-asn` command to autoderive RT that does not include the ASN in the RT value. This allows auto RT to be used even if the Clos leaf-spine design has separate ASN for each leaf node. Configure this command only when all the VTEPs are OS10 switches.

## 11. Configure VLT.

- Configure a VLTi VLAN for the virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vlti-vlan 200
OS10(config-vn-20000)# exit
```

- Configure a dedicated Layer 3 forwarding path through the other VLT peer for connectivity even if all spine links go down. This VLAN interface is an unnumbered interface and used for iBGP peering with the other VLT peer.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ipv6 nd max-ra-interval 4
OS10(config-if-vl-4000)# ipv6 nd min-ra-interval 3
OS10(config-if-vl-4000)# ipv6 nd send-ra
OS10(config-if-vl-4000)# exit
```

- Configure the VLT port channel.

```
OS10(config)# interface port-channel20
OS10(config-if-po-20)# vlt-port-channel 20
OS10(config-if-po-20)# exit
```

- Configure VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(config-if-eth1/1/3)# no shutdown
OS10(config-if-eth1/1/3)# no switchport
OS10(config-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(config-if-eth1/1/4)# no shutdown
OS10(config-if-eth1/1/4)# no switchport
OS10(config-if-eth1/1/4)# exit
```

- Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(config-vlt-1)# backup destination 10.16.150.3
OS10(config-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(config-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(config-vlt-1)# exit
```

- Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(config-uplink-state-group-1)# enable
OS10(config-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(config-uplink-state-group-1)# upstream port-channel20
OS10(config-uplink-state-group-1)# exit
```

- Configure iBGP unnumbered peering between VLT peers with both IPv4 and L2VPN EVPN address families.

```
OS10(config)# router bgp 202
OS10(config-router-bgp-202)# template ibgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface vlan4000
OS10(config-router-neighbor)# inherit template ibgp_unified inherit-type ibgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

## 12. Configure IP routing in the overlay network.

- Create the tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

- Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

- Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.233/16
OS10(conf-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-if-vn-20000)# exit
```

13. Configure symmetric IRB. In EVPN mode, configure the router MAC address that is used by remote VTEPs as the destination address in VXLAN encapsulated packets that are sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:06
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# vni 3000
OS10(config-evpn-vrf-tenant1)# route-target auto
OS10(config-evpn-vrf-tenant1)# exit
OS10(config-evpn)# exit
```

14. Configure an externally connected VLAN.

```
OS10(conf)# interface vlan 200
OS10(conf-if-vlan)# ip vrf forwarding tenant1
OS10(conf-if-vlan)# ip address 10.10.0.1/16
OS10(conf-if-vlan)# no shutdown
OS10(conf-if-vlan)# exit
OS10(conf)# interface ethernet 1/1/7
OS10(conf-if-eth1/1/7)# switchport mode trunk
OS10(conf-if-eth1/1/7)# switchport trunk allowed vlan 200
```

15. Configure advertisement of the connected networks through EVPN Type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 connected
OS10(config-evpn-vrf-tenant1)# exit
```

### VTEP Leaf Switch 4 configuration

1. Configure a loopback interface for the VXLAN underlay using the same IP address as the VLT peer.

```
OS10(config)# interface loopback0
OS10(conf-if-lo-0)# no shutdown
OS10(conf-if-lo-0)# ip address 192.168.2.1/32
OS10(conf-if-lo-0)# exit
```

2. Configure the Loopback interface as the VXLAN source tunnel interface.

```
OS10(config)# nve
OS10(config-nve)# source-interface loopback0
OS10(config-nve)# exit
```

3. Configure the VXLAN virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# vxlan-vni 20000
OS10(config-vn-vxlan-vni)# exit
OS10(config-vn-20000)# exit
```

4. Configure an unused VLAN ID for untagged membership.

```
OS10(config)# virtual-network untagged-vlan 1000
```

5. Configure access ports as VLAN members for a port-scoped VLAN-to-VNI mapping.

```
OS10(config)# interface port-channel20
OS10(config-if-po-20)# no shutdown
OS10(config-if-po-20)# switchport mode trunk
OS10(config-if-po-20)# no switchport access vlan
OS10(config-if-po-20)# exit
OS10(config)# interface ethernet1/1/6
OS10(config-if-eth1/1/6)# no shutdown
OS10(config-if-eth1/1/6)# channel-group 20 mode active
OS10(config-if-eth1/1/6)# exit
```

6. Add the access ports to the virtual network.

```
OS10(config)# virtual-network 20000
OS10(config-vn-20000)# member-interface port-channel 20 untagged
OS10(config-vn)# exit
```

7. Configure upstream network-facing ports as unnumbered interfaces. Configure the `ipv6 nd send-ra` command and lower RA intervals. These interfaces would be used for BGP unnumbered peering.

```
OS10(config)# interface ethernet1/1/1
OS10(config-if-eth1/1/1)# no shutdown
OS10(config-if-eth1/1/1)# no switchport
OS10(config-if-eth1/1/1)# mtu 1650
OS10(config-if-eth1/1/1)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/1)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/1)# ipv6 nd send-ra
OS10(config-if-eth1/1/1)# exit
OS10(config)# interface ethernet1/1/2
OS10(config-if-eth1/1/2)# no shutdown
OS10(config-if-eth1/1/2)# no switchport
OS10(config-if-eth1/1/2)# mtu 1650
OS10(config-if-eth1/1/2)# ipv6 nd max-ra-interval 4
OS10(config-if-eth1/1/2)# ipv6 nd min-ra-interval 3
OS10(config-if-eth1/1/2)# ipv6 nd send-ra
OS10(config-if-eth1/1/2)# exit
```

8. Configure BGP instance with router id.

```
OS10(config)# router bgp 202
OS10(config-router-bgp-202)# router-id 172.19.0.1
OS10(config-router-bgp-202)# address-family ipv4 unicast
OS10(configure-router-bgp-af)# redistribute connected
OS10(configure-router-bgp-af)# exit
```

9. Configure a BGP unnumbered neighbor over network facing ports. Use a template to simplify the configuration on multiple interfaces. These neighbors are configured to carry IPv4 address family (default) and L2VPN EVPN address family.

```
OS10(config-router-bgp-202)# template ebgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface ethernet1/1/1
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor interface ethernet1/1/2
OS10(config-router-neighbor)# inherit template ebgp_unified inherit-type ebgp
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

10. Configure EVPN for the VXLAN virtual network. Configure the EVPN instance manual configuration mode, and RD, and RT configuration in auto mode.

```
OS10(config)# evpn
OS10(config-evpn)# disable-rt-asn
```

```
OS10(config-evpn)# evi 20000
OS10(config-evpn-evi-20000)# vni 20000
OS10(config-evpn-evi-20000)# rd auto
OS10(config-evpn-evi-20000)# route-target auto
OS10(config-evpn-evi-20000)# exit
OS10(config-evpn)# exit
```

**i NOTE:** Use the `disable-rt-asn` command to autoderive RT that does not include the ASN in the RT value. This allows auto RT to be used even if the Clos leaf-spine design has separate ASN for each leaf node. Configure this command only when all the VTEPs are OS10 switches.

## 11. Configure VLT.

- Configure a VLTi VLAN for the virtual network.

```
OS10(config)# virtual-network 20000
OS10(conf-vn-20000)# vlti-vlan 200
OS10(conf-vn-20000)# exit
```

- Configure a dedicated Layer 3 forwarding path through the other VLT peer if all spine links go down. This VLAN interface is unnumbered interface and is used for iBGP peering with the other VLT peer.

```
OS10(config)# interface vlan4000
OS10(config-if-vl-4000)# no shutdown
OS10(config-if-vl-4000)# ipv6 nd max-ra-interval 4
OS10(config-if-vl-4000)# ipv6 nd min-ra-interval 3
OS10(config-if-vl-4000)# ipv6 nd send-ra
OS10(config-if-vl-4000)# exit
```

- Configure the VLT port channel.

```
OS10(config)# interface port-channel20
OS10(conf-if-po-20)# vlt-port-channel 20
OS10(conf-if-po-20)# exit
```

- Configure VLTi member links.

```
OS10(config)# interface ethernet1/1/3
OS10(conf-if-eth1/1/3)# no shutdown
OS10(conf-if-eth1/1/3)# no switchport
OS10(conf-if-eth1/1/3)# exit
OS10(config)# interface ethernet1/1/4
OS10(conf-if-eth1/1/4)# no shutdown
OS10(conf-if-eth1/1/4)# no switchport
OS10(conf-if-eth1/1/4)# exit
```

- Configure the VLT domain.

```
OS10(config)# vlt-domain 1
OS10(conf-vlt-1)# backup destination 10.16.150.4
OS10(conf-vlt-1)# discovery-interface ethernet1/1/3,1/1/4
OS10(conf-vlt-1)# vlt-mac aa:bb:cc:dd:ff:ee
OS10(conf-vlt-1)# exit
```

- Configure UFD with uplink VLT ports and downlink network ports.

```
OS10(config)# uplink-state-group 1
OS10(conf-uplink-state-group-1)# enable
OS10(conf-uplink-state-group-1)# downstream ethernet1/1/1-1/1/2
OS10(conf-uplink-state-group-1)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

- Configure iBGP unnumbered peering between VLT peers with both IPv4 and L2VPN EVPN address families.

```
OS10(config)# router bgp 202
OS10(config-router-bgp-202)# template ibgp_unified
OS10(config-router-template)# send-community extended
OS10(config-router-template)# address-family l2vpn evpn
OS10(config-router-bgp-template-af)# activate
OS10(config-router-bgp-template-af)# exit
OS10(config-router-template)# neighbor interface vlan4000
OS10(config-router-neighbor)# inherit template ibgp_unified inherit-type ibgp
```

```
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit
```

12. Configure IP routing in the overlay network.

- Create a tenant VRF.

```
OS10(config)# ip vrf tenant1
OS10(conf-vrf)# exit
```

- Configure an anycast gateway MAC address.

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

- Configure routing on the virtual network.

```
OS10(config)# interface virtual-network 20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.234/16
OS10(conf-if-vn-20000)# ip virtual-router address 10.2.0.100
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-if-vn-20000)# exit
```

13. Configure symmetric IRB. In EVPN mode, configure the router MAC address that is used by remote VTEPs as the destination address in VXLAN encapsulated packets that are sent to the switch. Configure a dedicated VXLAN VNI for symmetric IRB for each tenant VRF.

```
OS10(config)# evpn
OS10(config-evpn)# router-mac 00:01:02:03:04:06
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# vni 3000
OS10(config-evpn-vrf-tenant1)# route-target auto
OS10(config-evpn-vrf-tenant1)# exit
OS10(config-evpn)# exit
```

14. Configure an externally connected VLAN.

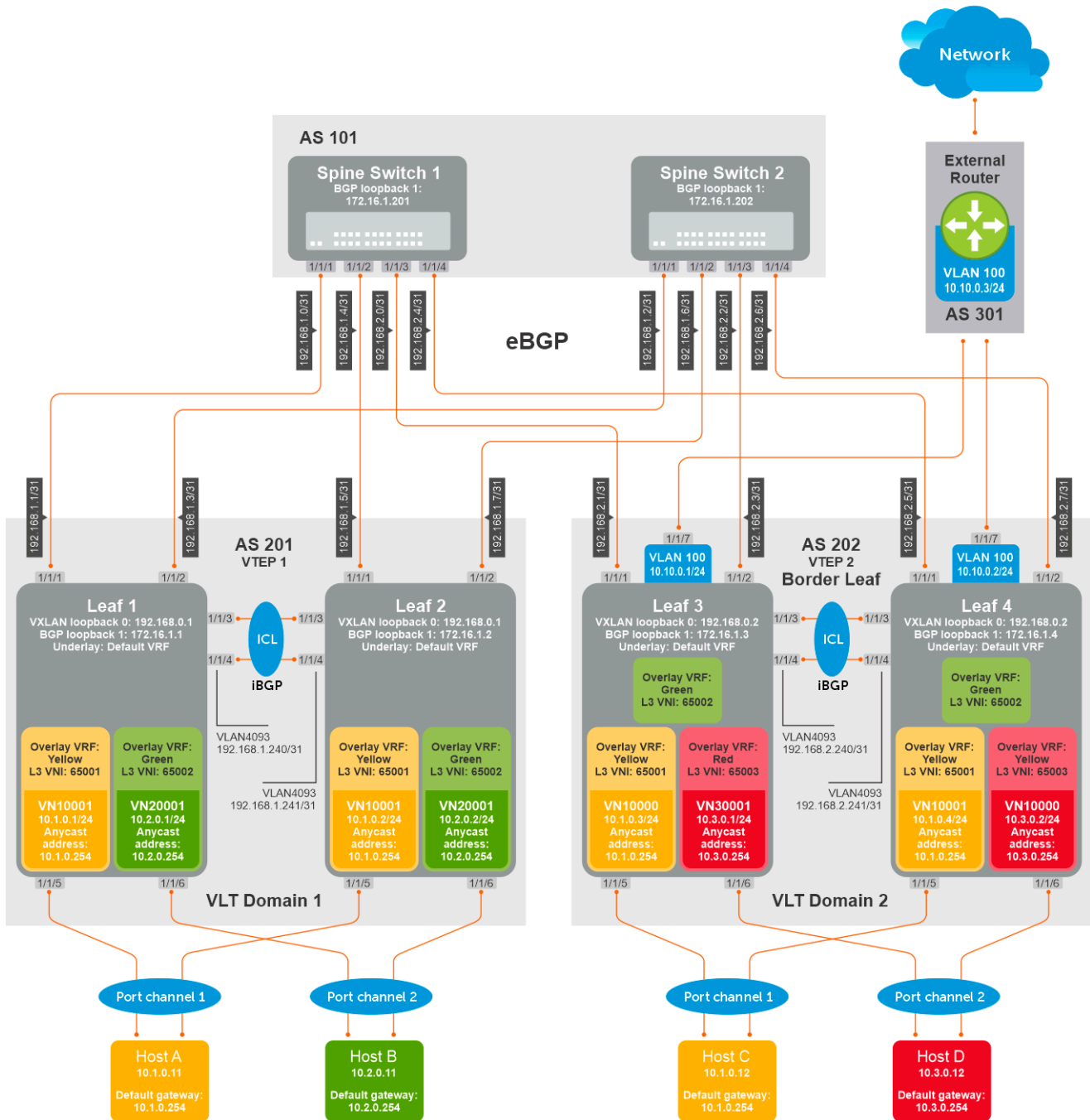
```
OS10(conf)# interface vlan 200
OS10(conf-if-vlan)# ip vrf forwarding tenant1
OS10(conf-if-vlan)# ip address 10.10.0.2/16
OS10(conf-if-vlan)# no shutdown
OS10(conf-if-vlan)# exit
OS10(conf)# interface ethernet 1/1/7
OS10(conf-if-eth1/1/7)# switchport mode trunk
OS10(conf-if-eth1/1/7)# switchport trunk allowed vlan 200
```

15. Configure advertisement of the connected networks through EVPN Type-5 routes.

```
OS10(config)# evpn
OS10(config-evpn)# vrf tenant1
OS10(config-evpn-vrf-tenant1)# advertise ipv4 connected
OS10(config-evpn-vrf-tenant1)# exit
```

# Example - Route leaking across VRFs in a VXLAN BGP EVPN symmetric IRB topology

The following VXLAN with BGP EVPN example uses a Clos leaf-spine topology to show how to set up route leaking across VRF in a symmetric IRB topology.



The following explains how the network is configured:

- All VTEPs perform symmetric IRB routing. In this example, all spine nodes are in one autonomous system and each VTEP in the leaf network belongs to a different autonomous system. Spine switch 1 is in AS 101. Spine switch 2 is in AS 101. For leaf nodes, VLT domain 1 is in AS 201; VLT domain 2 is in AS 202. VLT domain 2 is a border leaf VTEP.
- The individual switch configuration shows how to configure VRFs in the VTEPs and configure route leaking between VRFs. For other VXLAN and BGP EVPN configuration, see other examples and the VXLAN section.
- Route leaking is performed on the Border Leaf VTEP.
- There are three nondefault VRFs present in the network – Yellow, Green, and Red.

- Route leaking is done between:
  - VRF-Yellow and VRF-Green.
  - VRF-Yellow and VRF-Red.
  - VRF-Yellow and VRF-default (underlay with external router)
- **NOTE:** Route leaking is not performed between VRF-Green and VRF-Red.
- On VTEPs 1 and 2, two VRFs are present – VRF-Yellow and VRF-Green. VN10001 is part of VRF-Yellow and VN20001 is part of VRF-Green.
- On VTEPs 3 and 4, three VRFs are present – VRF-Yellow, VRF-Green and VRF-Red. VN10001 is part of VRF-Yellow and VN30001 is part of VRF-Red. VRF-Green does not have local VNs.
- On all VTEPs, symmetric IRB is configured in EVPN mode using a unique, dedicated VXLAN VNI, and Auto RD/RT values for each tenant VRF.
- On all VTEPs, the `disable-rt-asn` command is used to autoderive the RT that does not include the ASN in the RT value. This allows auto RT to be used even if there are separate ASNs for each leaf node.
- A VLAN to an external network is configured on VTEPs 3 and 4 in the VLT domain that serves as the border-leaf gateway.

### Leaf 1 configuration

1. Configure VRFs Yellow and Green.

```
OS10(config)# ip vrf Yellow
OS10(config-vrf)# exit
OS10(config)# ip vrf Green
OS10(config-vrf)# exit
```

2. Configure Layer 3 virtual-network interfaces with VRFs and IP addresses.

```
OS10(config)# interface virtual-network 10001
OS10(config-if-vn-10001)# ip vrf forwarding Yellow
OS10(config-if-vn-10001)# ip address 10.1.0.1/24
OS10(config-if-vn-10001)# ip virtual-router address 10.1.0.254
OS10(config-if-vn-10001)#
OS10(config)# interface virtual-network 20001
OS10(config-if-vn-20001)# ip vrf forwarding Green
OS10(config-if-vn-20001)# ip address 10.2.0.1/24
OS10(config-if-vn-20001)# ip virtual-router address 10.2.0.254
```

**NOTE:** For creating the virtual-networks with access ports, check the relevant sections.

3. Configure EVPN with IP-VRFs.

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# router-mac de:11:de:11:00:01
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# vni 65001
OS10(config-evpn-vrf-Yellow)# route-target auto
OS10(config-evpn-vrf-Yellow)# advertise ipv4 connected
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# vni 65002
OS10(config-evpn-vrf-Green)# route-target auto
OS10(config-evpn-vrf-Green)# advertise ipv4 connected
OS10(config-evpn-vrf-Green)# exit
```

### Leaf 2 configuration

1. Configure VRFs Yellow and Green.

```
OS10(config)# ip vrf Yellow
OS10(config-vrf)# exit
OS10(config)# ip vrf Green
OS10(config-vrf)# exit
```

2. Configure Layer 3 virtual-network interfaces with VRFs and IP addresses.

```
OS10(config)# interface virtual-network 10001
OS10(config-if-vn-10001)# ip vrf forwarding Yellow
```

```

OS10(config-if-vn-10001)# ip address 10.1.0.2/24
OS10(config-if-vn-10001)# ip virtual-router address 10.1.0.254
OS10(config-if-vn-10001)#
OS10(config)# interface virtual-network 20001
OS10(config-if-vn-20001)# ip vrf forwarding Green
OS10(config-if-vn-20001)# ip address 10.2.0.2/24
OS10(config-if-vn-20001)# ip virtual-router address 10.2.0.254

```

### 3. Configure EVPN with IP-VRFs.

```

OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# router-mac de:11:de:11:00:02
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# vni 65001
OS10(config-evpn-vrf-Yellow)# route-target auto
OS10(config-evpn-vrf-Yellow)# advertise ipv4 connected
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# vni 65002
OS10(config-evpn-vrf-Green)# route-target auto
OS10(config-evpn-vrf-Green)# advertise ipv4 connected
OS10(config-evpn-vrf-Green)# exit

```

### Leaf3 configuration:

#### 1. Configure VRFs Yellow, Green, and Red.

```

OS10(config)# ip vrf Yellow
OS10(config-vrf)# exit
OS10(config)# ip vrf Green
OS10(config-vrf)# exit
OS10(config)# ip vrf Red
OS10(config-vrf)# exit

```

#### 2. Configure Layer 3 virtual-network interfaces with VRFs and IP addresses.

```

OS10(config)# interface virtual-network 10001
OS10(config-if-vn-10001)# ip vrf forwarding Yellow
OS10(config-if-vn-10001)# ip address 10.1.0.3/24
OS10(config-if-vn-10001)# ip virtual-router address 10.1.0.254
OS10(config-if-vn-10001)#
OS10(config)# interface virtual-network 30001
OS10(config-if-vn-30001)# ip vrf forwarding Red
OS10(config-if-vn-30001)# ip address 10.3.0.1/24
OS10(config-if-vn-30001)# ip virtual-router address 10.3.0.254

```

#### 3. Configure EVPN with IP-VRFs.

```

OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# router-mac de:11:de:11:00:02
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# vni 65001
OS10(config-evpn-vrf-Yellow)# route-target auto
OS10(config-evpn-vrf-Yellow)# advertise ipv4 connected
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# vni 65002
OS10(config-evpn-vrf-Green)# route-target auto
OS10(config-evpn-vrf-Green)# advertise ipv4 connected
OS10(config-evpn-vrf-Green)# exit
OS10(config-evpn)# vrf Red
OS10(config-evpn-vrf-Red)# vni 65003
OS10(config-evpn-vrf-Red)# route-target auto
OS10(config-evpn-vrf-Red)# advertise ipv4 connected
OS10(config-evpn-vrf-Red)# exit

```

4. Configure the border-leaf to advertise the default route into the EVPN in each VRF. From the other VTEPs, any traffic to an external network and also to networks which are not within the local VRF reaches the Border Leaf router using this default route.

a. **If the border-leaf is already getting a default route from an external router for each VRF:** Advertise the BGP route using the `advertise ipv4 bgp` command for each VRF in the EVPN.

```
OS10(config)# evpn
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# advertise ipv4 bgp
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# advertise ipv4 bgp
OS10(config-evpn-vrf-Green)# exit
```

b. **If the border-leaf does not get a default route from an external router:** Configure a static null default route in each VRF and advertise it using `advertise ipv4 static` command for each VRF in the EVPN.

```
OS10(config)# ip route vrf Yellow 0.0.0.0/0 interface null 0
OS10(config)# ip route vrf Green 0.0.0.0/0 interface null 0
OS10(config)# evpn
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# advertise ipv4 static
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# advertise ipv4 static
OS10(config-evpn-vrf-Green)# exit
```

5. (Optional) Configure route-maps with a prefix-list to leak selective routes from each VRF.

```
OS10(config)# ip prefix-list PrefixList_DefaultVrf_Export permit 10.10.0.0/24
OS10(config)# ip prefix-list PrefixList_YellowVrf_Export permit 10.1.0.0/24 le 32
OS10(config)# ip prefix-list PrefixList_GreenVrf_Export permit 10.2.0.0/24
OS10(config)# ip prefix-list PrefixList_RedVrf_Export permit 10.3.0.0/24
OS10(config)# route-map RouteMap_DefaultVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_DefaultVrf_Export
OS10(config-route-map)# exit
OS10(config)# route-map RouteMap_YellowVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_YellowVrf_Export
OS10(config-route-map)# exit
OS10(config)# route-map RouteMap_GreenVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_GreenVrf_Export
OS10(config-route-map)# exit
OS10(config)# route-map RouteMap_RedVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_RedVrf_Export
OS10(config-route-map)# exit
```

**NOTE:** While leaking EVPN routes, only the subnet routes must be leaked. Host routes (/32) need not be leaked and could be blocked using route-maps. But, if you have certain VNs stretched on the border-leaf as well (like in Yellow VRF), you must leak the host routes as well.

6. Configure route leaking between:

- Yellow VRF and default VRF.
- Yellow VRF and Green VRF.
- Yellow VRF and Red VRF.

```
OS10(config)# ip vrf default
OS10(conf-vrf)# ip route-export 0:0 route-map RouteMap_DefaultVrf_Export
OS10(conf-vrf)# ip route-import 1:1
OS10(conf-vrf)# exit
OS10(config)# ip vrf Yellow
OS10(conf-vrf)# ip route-export 1:1 route-map RouteMap_YellowVrf_Export
OS10(conf-vrf)# ip route-import 0:0
OS10(conf-vrf)# ip route-import 2:2
OS10(conf-vrf)# ip route-import 3:3
OS10(conf-vrf)# exit
OS10(config)# ip vrf Green
OS10(conf-vrf)# ip route-export 2:2 route-map RouteMap_GreenVrf_Export
OS10(conf-vrf)# ip route-import 1:1
OS10(conf-vrf)# exit
OS10(config)# ip vrf Red
```

```

OS10(config-vrf)# ip route-export 3:3 route-map RouteMap_RedVrf_Export
OS10(config-vrf)# ip route-import 1:1
OS10(config-vrf)# exit

```

7. (Optional) For advertising leaked routes from Yellow VRF only to an external router on the default VRF and not to an underlay network, use route-maps on spine-facing eBGP neighbors and also on the iBGP neighbor between the VLT peers.

```

OS10(config)# ip prefix-list PrefixList_Deny_YellowVrfRoutes deny 10.1.0.0/24 le
OS10(config)# ip prefix-list PrefixList_Deny_YellowVrfRoutes permit 0.0.0.0/0 le 32
OS10(config)#
OS10(config)# route-map RouteMap_Deny_YellowVrfRoutes
OS10(config-route-map)# match ip address prefix-list PrefixList_Deny_YellowVrfRoutes
OS10(config-route-map)#
OS10(config-route-map)# router bgp 202
OS10(config-router-bgp-202)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute l2vpn evpn
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit
OS10(config-router-bgp-202)# neighbor 192.168.2.0
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# route-map RouteMap_Deny_YellowVrfRoutes out
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor 192.168.2.2
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# route-map RouteMap_Deny_YellowVrfRoutes out
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor 192.168.2.241
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# route-map RouteMap_Deny_YellowVrfRoutes out
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor 10.10.0.3
OS10(config-router-neighbor)# remote-as 301
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

```

#### Leaf 4 configuration

1. Configure VRFs Yellow, Green, and Red.

```

OS10(config)# ip vrf Yellow
OS10(config-vrf)# exit
OS10(config)# ip vrf Green
OS10(config-vrf)# exit
OS10(config)# ip vrf Red
OS10(config-vrf)# exit

```

2. Configure Layer 3 virtual-network interfaces with VRFs and IP addresses.

```

OS10(config)# interface virtual-network 10001
OS10(config-if-vn-10001)# ip vrf forwarding Yellow
OS10(config-if-vn-10001)# ip address 10.1.0.4/24
OS10(config-if-vn-10001)# ip virtual-router address 10.1.0.254
OS10(config-if-vn-10001)#
OS10(config)# interface virtual-network 30001
OS10(config-if-vn-30001)# ip vrf forwarding Red
OS10(config-if-vn-30001)# ip address 10.3.0.2/24
OS10(config-if-vn-30001)# ip virtual-router address 10.3.0.254

```

3. Configure EVPN with IP-VRFs.

```

OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# vni 65001
OS10(config-evpn-vrf-Yellow)# route-target auto
OS10(config-evpn-vrf-Yellow)# advertise ipv4 connected
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green

```

```

OS10(config-evpn-vrf-Green)# vni 65002
OS10(config-evpn-vrf-Green)# route-target auto
OS10(config-evpn-vrf-Green)# advertise ipv4 connected
OS10(config-evpn-vrf-Green)# exit
OS10(config-evpn)# vrf Red
OS10(config-evpn-vrf-Red)# vni 65003
OS10(config-evpn-vrf-Red)# route-target auto
OS10(config-evpn-vrf-Red)# advertise ipv4 connected
OS10(config-evpn-vrf-Red)# exit

```

4. Configure a border-leaf to advertise the default route into the EVPN in each VRF. From the other VTEPs, any traffic to external network and also to networks which are not within the local VRF reaches the Border-Leaf router using this default route.

- a. **If the border-leaf is already getting a default route from an external router for each VRF:** Advertise the BGP route using the `advertise ipv4 bgp` command for each VRF in the EVPN.

```

OS10(config)# evpn
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# advertise ipv4 bgp
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# advertise ipv4 bgp
OS10(config-evpn-vrf-Green)# exit

```

- b. **If the border-leaf does not get a default route from an external router:** Configure a static null default route in each VRF and advertise it using the `advertise ipv4 static` command for each VRF in the EVPN.

```

OS10(config)# ip route vrf Yellow 0.0.0.0/0 interface null 0
OS10(config)# ip route vrf Green 0.0.0.0/0 interface null 0
OS10(config)# evpn
OS10(config-evpn)# vrf Yellow
OS10(config-evpn-vrf-Yellow)# advertise ipv4 static
OS10(config-evpn-vrf-Yellow)# exit
OS10(config-evpn)# vrf Green
OS10(config-evpn-vrf-Green)# advertise ipv4 static
OS10(config-evpn-vrf-Green)# exit

```

5. (Optional) Configure route-maps with a prefix-list to leak selective routes from each VRF.

```

OS10(config)# ip prefix-list PrefixList_DefaultVrf_Export permit 10.10.0.0/24
OS10(config)# ip prefix-list PrefixList_YellowVrf_Export permit 10.1.0.0/24 le 32
OS10(config)# ip prefix-list PrefixList_GreenVrf_Export permit 10.2.0.0/24
OS10(config)# ip prefix-list PrefixList_RedVrf_Export permit 10.3.0.0/24
OS10(config)#
OS10(config)# route-map RouteMap_DefaultVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_DefaultVrf_Export
OS10(config-route-map)# exit
OS10(config)# route-map RouteMap_YellowVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_YellowVrf_Export
OS10(config-route-map)# exit
OS10(config)# route-map RouteMap_GreenVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_GreenVrf_Export
OS10(config-route-map)# exit
OS10(config)# route-map RouteMap_RedVrf_Export
OS10(config-route-map)# match ip address prefix-list PrefixList_RedVrf_Export
OS10(config-route-map)# exit

```

**NOTE:** While leaking EVPN routes, only the subnet routes must be leaked. Host routes (/32) need not be leaked and could be blocked using route-maps. But, if you have certain VNs stretched on border leaf as well (like in Yellow VRF), you must leak the host routes as well.

6. Configure route leaking between:

- Yellow VRF and default VRF.
- Yellow VRF and Green VRF.
- Yellow VRF and Red VRF.

```

OS10(config)# ip vrf default
OS10(conf-vrf)# ip route-export 0:0 route-map RouteMap_DefaultVrf_Export
OS10(conf-vrf)# ip route-import 1:1
OS10(conf-vrf)# exit

```

```

OS10(config)# ip vrf Yellow
OS10(conf-vrf)# ip route-export 1:1 route-map RouteMap_YellowVrf_Export
OS10(conf-vrf)# ip route-import 0:0
OS10(conf-vrf)# ip route-import 2:2
OS10(conf-vrf)# ip route-import 3:3
OS10(conf-vrf)# exit
OS10(config)# ip vrf Green
OS10(conf-vrf)# ip route-export 2:2 route-map RouteMap_GreenVrf_Export
OS10(conf-vrf)# ip route-import 1:1
OS10(conf-vrf)# exit
OS10(config)# ip vrf Red
OS10(conf-vrf)# ip route-export 3:3 route-map RouteMap_RedVrf_Export
OS10(conf-vrf)# ip route-import 1:1
OS10(conf-vrf)# exit

```

7. (Optional) For advertising leaked routes from the Yellow VRF only to an external router in the default VRF and not to an underlay network, use route-maps on spine facing eBGP neighbors and also on the iBGP neighbor between the VLT peers.

```

OS10(config)# ip prefix-list PrefixList_Deny_YellowVrfRoutes deny 10.1.0.0/24 le 32
OS10(config)# ip prefix-list PrefixList_Deny_YellowVrfRoutes permit 0.0.0.0/0 le 32
OS10(config)#
OS10(config)# route-map RouteMap_Deny_YellowVrfRoutes
OS10(config-route-map)# match ip address prefix-list PrefixList_Deny_YellowVrfRoutes
OS10(config-route-map)#
OS10(config-route-map)# router bgp 202
OS10(config-router-bgp-202)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute l2vpn evpn
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-af)# exit
OS10(config-router-bgp-202)# neighbor 192.168.2.4
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# route-map RouteMap_Deny_YellowVrfRoutes out
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor 192.168.2.5
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# route-map RouteMap_Deny_YellowVrfRoutes out
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor 192.168.2.240
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# route-map RouteMap_Deny_YellowVrfRoutes out
OS10(config-router-bgp-neighbor-af)# exit
OS10(config-router-neighbor)# exit
OS10(config-router-bgp-202)# neighbor 10.10.0.3
OS10(config-router-neighbor)# remote-as 301
OS10(config-router-neighbor)# no shutdown
OS10(config-router-neighbor)# exit

```

#### Verify leaked routes using show outputs on the the Border-Leaf switch:

```

OS10# show ip route vrf Yellow
Codes: C - connected
       S - static
       B - BGP, IN - internal BGP, EX - external BGP, EV - EVPN BGP
       O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
       N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
       E2 - OSPF external type 2, * - candidate default,
       + - summary route, > - non-active route
Gateway of last resort is Direct to network 0.0.0.0

```

Metric	Destination	Last Change	Gateway	Dist/
*S	0.0.0.0/0		Direct	null0
0/0		00:38:51		
C	10.1.0.0/24		via 10.1.0.3	virtual-network10001
0/0		00:47:11		
B EV	10.1.0.1/32		via 192.168.0.1	
200/0		00:48:55		
B EV	10.1.0.2/32		via 192.168.0.1	
200/0		00:48:55		

```

B EV 10.2.0.0/24 via 192.168.0.1,Green
200/0 00:35:48
C 10.3.0.0/24 via 10.3.0.1,Red virtual-network30001
0/0 00:35:48
C 10.10.0.0/24 via 10.10.0.1,default vlan100
0/0 00:25:42
OS10# show ip route vrf Green
Codes: C - connected
S - static
B - BGP, IN - internal BGP, EX - external BGP, EV - EVPN BGP
O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
E2 - OSPF external type 2, * - candidate default,
+ - summary route, > - non-active route
Gateway of last resort is Direct to network 0.0.0.0
Destination Gateway Dist/
Metric Last Change
-----
*S 0.0.0.0/0 Direct null0
0/0 00:39:24
C 10.1.0.0/24 via 10.1.0.3,Yellow virtual-network10001
0/0 00:36:22
B EV 10.1.0.1/32 via 192.168.0.1,Yellow
200/0 00:36:22
B EV 10.1.0.2/32 via 192.168.0.1,Yellow
200/0 00:36:22
B EV 10.2.0.0/24 via 192.168.0.1
200/0 00:41:47
B EV 10.2.0.1/32 via 192.168.0.1
200/0 00:41:47
B EV 10.2.0.2/32 via 192.168.0.1
200/0 00:41:47
B EV 10.2.0.254/32 via 192.168.0.1
200/0 00:41:47
OS10# show ip route vrf Red
Codes: C - connected
S - static
B - BGP, IN - internal BGP, EX - external BGP, EV - EVPN BGP
O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
E2 - OSPF external type 2, * - candidate default,
+ - summary route, > - non-active route
Gateway of last resort is not set
Destination Gateway Dist/
Metric Last Change
-----
C 10.1.0.0/24 via 10.1.0.3,Yellow virtual-network10001
0/0 00:36:26
B EV 10.1.0.1/32 via 192.168.0.1,Yellow
200/0 00:36:26
B EV 10.1.0.2/32 via 192.168.0.1,Yellow
200/0 00:36:26
C 10.3.0.0/24 via 10.3.0.1 virtual-network30001
0/0 00:45:44

```

### Verify routes on the external router

```

OS10# show ip route
Codes: C - connected
S - static
B - BGP, IN - internal BGP, EX - external BGP, EV - EVPN BGP
O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA external type 1,
N2 - OSPF NSSA external type 2, E1 - OSPF external type 1,
E2 - OSPF external type 2, * - candidate default,
+ - summary route, > - non-active route
Gateway of last resort is not set
Destination Gateway Dist/
Metric Last Change
-----
B EX 10.1.0.0/24 via 10.10.0.1

```

```

20/0          00:13:49          via 10.10.0.2
  B EX 10.1.0.1/32          via 10.10.0.1
20/0          00:14:22          via 10.10.0.2
  B EX 10.1.0.2/32          via 10.10.0.1
20/0          00:14:24          via 10.10.0.2
  C      10.10.0.0/24          via 10.10.0.3          vlan100
0/0          00:23:16          via 10.10.0.2
  B EX 172.16.1.1/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 172.16.1.2/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 172.16.1.3/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 172.16.1.4/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 172.16.1.201/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 172.16.1.202/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 192.168.0.1/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 192.168.0.2/32          via 10.10.0.1
20/0          00:22:58          via 10.10.0.2
  B EX 192.168.2.0/31          via 10.10.0.1
20/0          00:14:11          via 10.10.0.2
  B EX 192.168.2.2/31          via 10.10.0.1
20/0          00:14:11          via 10.10.0.2
  B EX 192.168.2.4/31          via 10.10.0.1
20/0          00:13:49          via 10.10.0.2
  B EX 192.168.2.6/31          via 10.10.0.1
20/0          00:13:49          via 10.10.0.2
  B EX 192.168.2.240/31          via 10.10.0.1
20/0          00:14:11          via 10.10.0.2

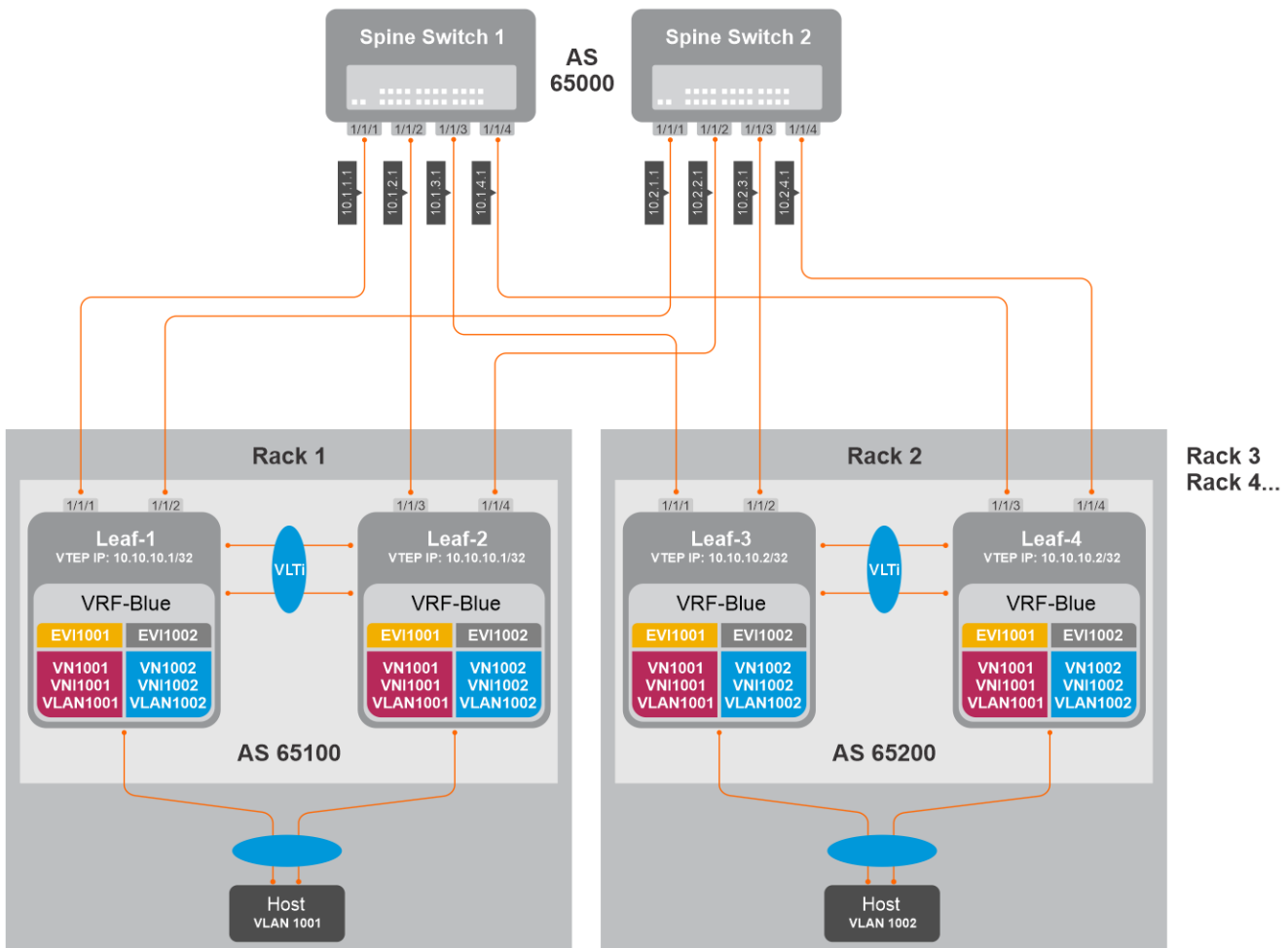
```

## Example: Migrating from Asymmetric IRB to Symmetric IRB

Until Release 10.5.0, OS10 provided support only for the Asymmetric IRB mode. Starting from Release 10.5.1, OS10 supports the Symmetric IRB mode. Symmetric IRB mode efficiently uses next hop tables in the NPU. If there are no local hosts, Symmetric IRB mode does not require creation of destination VNI in the local VTEP. You can migrate your network from Asymmetric IRB mode to Symmetric IRB mode. For a seamless migration with less or no downtime in the VLT environment, perform the following steps:

**NOTE:**

- Before you start this migration, all leaf nodes acting as VTEPs in the Clos network must be upgraded to 10.5.1.x.
- If there are overlay hosts in default VRF, migration to Symmetric IRB mode is not supported because Symmetric IRB mode cannot be used in default VRF.



## Asymmetric to Symmetric IRB migration steps

1. **Make the spines to send overlay traffic only to Leaf-2 by making Leaf-1 advertise VTEP IP with a higher metric in the underlay network.**

### Leaf-1 configuration

- a. Configure route-map with prefix-list to set the metric higher for the VTEP IP.

```
Leaf-1(config)# ip prefix-list vtep_ip seq 10 permit 10.10.10.1/32
Leaf-1(config)# route-map set_higher_metric permit 10
Leaf-1(config-route-map)# match ip address prefix-list vtep_ip
Leaf-1(config-route-map)# continue 20
Leaf-1(config-route-map)# set metric 100
Leaf-1(config-route-map)# exit
Leaf-1(config)# route-map set_higher_metric permit 20
Leaf-1(config-route-map)# exit
```

- b. Configure the route-map to the underlay BGP neighbors towards Spine.

```
Leaf-1(config)# router bgp 65100
Leaf-1(config-router-bgp-65100)# neighbor 10.1.1.1
Leaf-1(config-router-neighbor)# address-family ipv4 unicast
Leaf-1(config-router-bgp-neighbor-af)# route-map set_higher_metric out
Leaf-1(config-router-bgp-neighbor-af)# exit
Leaf-1(config-router-neighbor)# exit
Leaf-1(config-router-bgp-65100)# neighbor 10.2.1.1
Leaf-1(config-router-neighbor)# address-family ipv4 unicast
Leaf-1(config-router-bgp-neighbor-af)# route-map set_higher_metric out
Leaf-1(config-router-bgp-neighbor-af)# end
```

2. Spines would now send the overlay traffic destined to VLT domain 1 (Rack1) only to Leaf-2.
3. Configure Symmetric IRB mode in Leaf-2.

#### Leaf-2 configuration

- a. Configure router-mac.

```
Leaf-2(config)# evpn
Leaf-2(config-evpn)# router-mac 02:10:10:10:10:10
```

- b. Configure IP VRF with L3 VNI.

```
Leaf-2(config-evpn)# vrf BLUE
Leaf-2(config-evpn-vrf-VRF001)# vni 65001
```

- c. Configure RT (auto or manual) and RD (optional, default is auto).

```
Leaf-2(config-evpn-vrf-BLUE)# route-target auto
```

- d. Advertise IPv4 and IPv6 connected routes.

```
Leaf-2(config-evpn-vrf-BLUE)# advertise ipv4 connected
Leaf-2(config-evpn-vrf-BLUE)# advertise ipv6 connected
```

4. Leaf-2 is changed to Symmetric IRB mode. VTEPs in other racks could be using Symmetric IRB or Asymmetric IRB based on its own local configuration. Irrespective of what other remote VTEPs use, Leaf-2 could now handle VXLAN encapsulated traffic from both symmetric and asymmetric modes.
5. Configure Symmetric IRB in Leaf-1.

#### Leaf-1 configuration

- a. Configure router-mac.

```
Leaf-1(config)# evpn
Leaf-1(config-evpn)# router-mac 02:10:10:10:10:10
```

- b. Configure IP VRF with L3 VNI.

```
Leaf-1(config-evpn)# vrf BLUE
Leaf-1(config-evpn-vrf-VRF001)# vni 65001
```

- c. Configure RT (auto or manual) and RD (optional, default is auto).

```
Leaf-1(config-evpn-vrf-BLUE)# route-target auto
```

- d. Advertise IPv4 and IPv6 connected routes.

```
Leaf-1(config-evpn-vrf-BLUE)# advertise ipv4 connected
Leaf-1(config-evpn-vrf-BLUE)# advertise ipv6 connected
```

6. Remove the BGP MED configuration in Leaf-1. Spines start sending traffic to Leaf-1 as well. ECMP path from Spines towards Leaf-1 and Leaf-2 is restored.

#### Leaf-1 configuration

```
Leaf-1(config)# router bgp 65100
Leaf-1(config-router-bgp-65100)# neighbor 10.1.1.1
Leaf-1(config-router-neighbor)# address-family ipv4 unicast
Leaf-1(config-router-bgp-neighbor-af)# no route-map set_higher_metric out
Leaf-1(config-router-bgp-neighbor-af)# exit
Leaf-1(config-router-neighbor)# exit
Leaf-1(config-router-bgp-65100)# neighbor 10.2.1.1
Leaf-1(config-router-neighbor)# address-family ipv4 unicast
Leaf-1(config-router-bgp-neighbor-af)# no route-map set_higher_metric out
Leaf-1(config-router-bgp-neighbor-af)# end
```

7. Rack1 is migrated to use Symmetric IRB.
8. Repeat Steps 1-to-6 on Rack2 and other racks as well.
9. After changing all Racks to Symmetric IRB, you can perform the following optional configuration changes:
  - a. If the L2 VNI (MAC-VRF VNI) does not have local hosts in the VTEPs, you can remove those VNIs on those VTEPs.

- b. Default route configured in VTEPs pointing to border leaf using an intermediate VNI could be removed. Default route or external routes could now be advertised to the VTEPs from border leaf using `advertise` commands under EVPN-IP-VRF mode.

## DHCP relay commands

### ipv6 dhcp-relay source-interface

Configures IPv6 DHCP relay source-interface option.

<b>Syntax</b>	<code>[no] ipv6 dhcp-relay source-interface [ethernet <i>node/slot/port</i>   loopback <i>loopback-id</i>   port-channel <i>id-number</i>   vlan <i>vlan-id</i>   virtual-network <i>vn-id</i>]</code>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ethernet <i>node/slot/port</i></code> - Ethernet interface type.</li> <li>• <code>loopback <i>loopback-id</i></code> - Loopback interface type.</li> <li>• <code>port-channel <i>id-number</i></code> - Port-channel interface type.</li> <li>• <code>vlan <i>vlan-id</i></code> - VLAN interface type.</li> <li>• <code>virtual-network <i>vn-id</i></code> - Virtual network interface type.</li> </ul>
<b>Defaults</b>	ipv6 dhcp-relay source- interface option is disabled by default.
<b>Command Mode</b>	INTERFACE CONFIGURATION
<b>Usage Information</b>	<p>The DHCP IPv6 relay agents forward a DHCP IPv6 client packet with the source IPv6 address set to the IPv6 address corresponding to the outgoing interface of the DHCP IPv6 server. This command provides the capability to configure source IPv6 address for the DHCP request, when DHCP IPv6 relay source interface is configured on a given DHCP client facing interface. DHCP server uses the IPv6 address of that source interface as destination for RELAY-REPLY packets.</p> <p>You can configure source interface on all types of DHCP client interfaces such as, ethernet interface, loopback interfaces, port-channels, VLANs, and virtual-networks.</p>
<b>Security and access</b>	This command is restricted to the <code>netadmin</code> and <code>sysadmin</code> user roles.
<b>Example</b>	<pre>OS10-Relay(conf-if-eth1/1/1)# ipv6 dhcp-relay ? interface-id Enable interface-id option source-interface Ipv6 Source interface for dhcp relay  OS10-Relay(conf-if-eth1/1/1)# ipv6 dhcp-relay source-interface loopback1 OS10-Relay(conf-if-eth1/1/1)# ipv6 dhcp-relay source-interface ? ethernet Ethernet interface type loopback Loopback interface type port-channel Port-channel interface type vlan Vlan interface type virtual-network Virtual network type</pre>
<b>Supported Releases</b>	10.5.3.2 or later

### show ipv6 dhcp-relay

Displays the DHCPv6 relay information about the client interfaces.

<b>Syntax</b>	<code>show ipv6 dhcp-relay interface [{ethernet <i>node/slot/port</i>   port-channel <i>id-number</i>}   vlan <i>vlan-id</i> [{ethernet <i>node/slot/port</i>   port-channel <i>id-number</i>}]   virtual-network <i>vnid</i>]</code>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ethernet <i>node/slot/port</i></code>—Enter the interface information.</li> <li>• <code>port-channel <i>id-number</i></code>—Enter the port channel ID, from 1 to 999 or 1001 to 2000.</li> <li>• <code>vlan <i>vlan-id</i></code>—Enter the VLAN ID number.</li> </ul>

- `virtual-network vnid`—Enter a virtual-network ID.

**Defaults**

None

**Command Mode**

EXEC

**Usage Information**

The `show ipv6 dhcp-relay` command displays the following information:

```
Interface name
Per-interface interface-id field value
Per-interface remote-id field value
Per-interface source-interface field value
```

`show ipv6 dhcp relay interface <interface>` - This command displays the relay information for the requested interface.

If the interface-id option is enabled, the show command displays the option-18 value which is sent in packet Format:- `interface-id` (or) `prefix:interface-id` (if prefix is configured)

If the remote-id option is enabled, the show command displays the option-37 value which is sent in packet format:- `remote-id` (or) `prefix:remote-id` (if prefix is configured)

If interface-id option is disabled, the corresponding show command field displays `Disabled`.

If remote-id option is disabled, the corresponding show command field displays `Disabled`.

If source interface is configured with IP address, the corresponding show command field displays `source-interface IP address`.

If the source interface is configured without any IP address, the corresponding show command field displays that the `source-interface IP address` is not configured.

If source interface is not configured, the corresponding show command field displays that the `source-interface` is not configured.

If an interface is associated to a VLAN, then it is mandatory to give both VLAN and port information in the show command. For example, `show ipv6 dhcp-relay vlan 10 ethernet 1/1/1`. Otherwise, the show command displays an empty output.

**NOTE:** If an interface is associated with a VLAN, then you must specify both the VLAN and port information in the show command. For example, `show ipv6 dhcp-relay vlan 10 ethernet 1/1/1`. Otherwise, the show command displays empty output.

**NOTE:** This command is accessible to all user roles.

**Example**

```
OS10(conf)#ipv6 dhcp-relay remote-id
OS10(conf)#ipv6 dhcp-relay interface-id
OS10-Relay(config)# interface loopback 1
OS10-Relay(config)# interface loopback 1
OS10-Relay(conf-if-lo-1)# show configuration
!
interface loopback1
no shutdown
ipv6 address 55::1/64
OS10-Relay(conf-if-lo-1)#

OS10-Relay(conf-if-eth1/1/1)# ipv6 dhcp-relay source-interface loopback 1
Relay(config)# show ipv6 dhcp-relay interface ethernet 1/1/1
Interface : ethernet1/1/1
Interface-id[option-18] : Enabled
Interface-id value : ethernet1/1/1
Remote-id[option-37] : Enabled
Enterprise-number : 674
Remote-id value : 0003000126b6775dfb76
Source-ip : 55::1 (loopback1)
Relay(config)#

Relay(conf-if-eth1/1/1)# no ipv6 dhcp-relay source-interface
Relay(config)# show ipv6 dhcp-relay interface ethernet 1/1/1
Interface : ethernet1/1/1
```

```

Interface-id[option-18] : Enabled
Interface-id value : ethernet1/1/1
Remote-id[option-37] : Enabled
Enterprise-number : 674
Remote-id value : 0003000126b6775dfb76
Source-ip : Not Configured
Relay(config)#

```

**Supported Releases** 10.5.2.1 or later

## show ip dhcp-relay

Displays the DHCP relay information corresponding to the client interfaces.

**Syntax** `show ip dhcp-relay interface [vrf vrf-name] {ethernet slot/port | port-channel port-channel-id | vlan vlan-id | virtual-network vn-id}`

- Parameters**
- `vrf vrf-name`—Enter *vrf* then the name of the VRF.
  - `ethernet slot/port`—Displays information corresponding to the Ethernet interface that you specify.
  - `port-channel port-channel-id`—Displays information corresponding to the port channel that you specify. Valid values are from 1 to 999 or 1001 to 2000.
  - `vlan vlan-id`—Displays information corresponding to the VLAN interface that you specify.
  - `virtual-network vn-id`—Displays information corresponding to the virtual network that you specify.

**Defaults** None

**Command Mode** EXEC

**Usage Information** This command displays the global-level status of Option-82 and the interface-level Option-82 status.

The `show ip dhcp-relay interface` command displays the relay information corresponding to the requested interface enabled with the helper address. If you enable the Option-82 configuration, the Option-82 status appears as `Enabled (Default)`. If you disable the Option-82 configuration, the Option-82 status appears as `Disabled`.

The command displays the following information:

- Per-interface `giaddr` field value.
- Per-interface suboption 5 field value.
- Per-interface suboption 151 field value.
- Per-interface suboption 11 field value.
- Configured list of the DHCP server IP addresses with the reachable VRF name.
- Per-interface source-interface field value.

`show ip dhcp relay interface interface` – This command displays the relay information for the requested interface that is enabled with the helper address.

`show ip dhcp relay vrf vrf-name` – This command displays the relay information for all the interfaces that are enabled with the helper address and associated with the `vrf_name`.

- If the `Giaddr = 0`, it displays an additional message in the `GiAddr` field `None - Relay cannot work without gateway IP address`.
- If the link selection field does not have an interface or anycast gateway IP, the link-selection field shows the value `None`.
- If the `server_override` option is enabled, the server-override field displays `Enabled (<anycast gateway IP >)`.
- If the `server_override` option is enabled, and anycast gateway IP is not present, the server-override field displays `Enabled (None)`.
- If the `server_override` option is disabled at the Globally or Interface level, the server-override field displays `Disabled`.
- If the VSS option is disabled, the VSS field displays `Disabled`.

- If the VSS option is enabled Globally without any interface `vss-info`, the VSS field displays Enabled (None).
- If the VSS option is enabled Globally with interface `vss-info`, the VSS field displays Enabled (type <0/1/> value <value>).
- If source interface is configured with IP address, then the show command displays the source-interface IP address.
- If source interface is configured without any IP address, then the show command displays that the source-interface IP address is not configured.
- If source interface is not configured, then the show command displays that the source interface is not configured.

### Example (Interface)

```
OS10# show ip dhcp-relay interface vlan 3033

Interface                : Vlan3033
Gateway Address          : 3.0.17.1
Option 82 Status        : Enabled (Default)
Link Selection [option-5] : Enabled (192.168.33.1)
VSS Info [option-151]    : Enabled (type 0 Red)
Server ID override [option-11] : Enabled (192.168.33.254)
DHCP Helper Address List :

Helper-Address          VRF
-----
192.172.2.3             Red
```

### Example (source-interface)

```
OS10(conf)#ip dhcp-relay server-override
OS10(conf)#ip dhcp-relay vss

interface loopback1
no shutdown
ip address 11.1.1.1/32

interface ethernet1/1/1
no shutdown
no switchport
ip address 6.1.1.1/24
flowcontrol receive on
ip dhcp-relay source interface loopback 1
ip helper-address 3.1.1.2
ip helper-address 4.1.1.2
ip dhcp-relay vss-info type 255

interface ethernet1/1/2
no shutdown
no switchport
ip address 2.1.1.1/24
flowcontrol receive on
ip dhcp-relay source interface loopback 1
ip helper-address 3.1.1.2
ip helper-address 4.1.1.2
ip dhcp-relay vss-info type 255

interface vlan 100
no shutdown
no switchport
ip vrf forwarding vrf_red
ip address 100.1.1.1/24
flowcontrol receive on
ip dhcp-relay source interface loopback 1
ip helper-address 7.1.1.2 vrf vrf_red
ip helper-address 8.1.1.2 vrf vrf_red
ip dhcp-relay vss-info type 0 value vrf_red
```

```

OS10#show ip dhcp-relay interface Ethernet1/1/1

Interface          : Ethernet1/1/1
Gateway Address    : 11.1.1.1
Link selection Address [option-5] : Enabled (6.1.1.1)
VSS Info[option-151] : Enabled (type 255)
Server ID override[option-11]    : Enabled (6.1.1.1)
Source-ip : 10.10.1.1 (loopback1)
DHCP Helper Address list          :

Helper-address      VRF
-----
3.1.1.2             default
4.1.1.2             default

OS10#show ip dhcp-relay interface Vlan 100

Interface          : Vlan 100
Gateway Address    : 11.1.1.1
Link selection Address [option-5] : Enabled (100.1.1.1)
VSS Info[option-151] : Enabled (type 0 vrf_red)
Server ID override[option-11]    : Enabled (100.1.1.1)
Source-ip : 10.10.1.1 (loopback1)
DHCP Helper Address list          :

Helper-address      VRF
-----
7.1.1.2             vrf_red
8.1.1.2             vrf_red

OS10#show ip dhcp-relay vrf default

Interface          : Ethernet1/1/1
Gateway Address    : 11.1.1.1
Link selection Address [option-5] : Enabled (6.1.1.1)
VSS Info[option-151] : Enabled (type 255)
Server ID override[option-11]    : Enabled (6.1.1.1)
Source-ip : 10.10.1.1 (loopback1)
DHCP Helper Address list          :

Helper-address      VRF
-----
3.1.1.2             default
4.1.1.2             default

Interface          : Ethernet1/1/2
Gateway Address    : 11.1.1.1
Link selection Address [option-5] : Enabled (2.1.1.1)
VSS Info[option-151] : Enabled (type 255)
Server ID override[option-11]    : Enabled (2.1.1.1)
Source-ip : 10.10.1.1 (loopback1)
DHCP Helper Address list          :

Helper-address      VRF
-----
3.1.1.2             default
4.1.1.2             default

OS10#show ip dhcp-relay vrf vrf_red

Interface          : Vlan 100
Gateway Address    : 11.1.1.1
Link selection Address [option-5] : Enabled (100.1.1.1)
VSS Info[option-151] : Enabled (type 0 vrf_red)
Server ID override[option-11]    : Enabled (100.1.1.1)
Source-ip : 10.10.1.1 (loopback1)
DHCP Helper Address list          :

Helper-address      VRF
-----

```

7.1.1.2	vrf_red
8.1.1.2	vrf_red

**Supported Releases** 10.5.2.0 or later

## show vlt mismatch dhcpv6-relay

Displays the mismatch (if any), between the VLT peer for the DHCPv6 relay options configuration on the Global level, VLT port-channel, and VLANs and VxLANs spanned across the VLT peers.

- Syntax** `show vlt vlt-domain mismatch dhcpv6-relay`
- Parameters** None.
- Defaults** None
- Command Mode** EXEC
- Security and access** This command is restricted to the `netadmin`, `secadmin`, and `sysadmin` roles.
- Usage Information** This command shows the mismatch in the following parameters:

- dhcpv6 hostname.
- global interface-id and remote-id enable/disable status.
- prefix configuration of both interface-id and remote-id.
- remote-id customized description configuration.
- interface-id customized description configuration for vlt port-channel, spanned vlan and vxlans.
- Status of the source-interface configuration; whether present or not present.

This command is restricted to `netadmin`, `secadmin`, and `sysadmin` role users.

### Example (Interface)

```
OS10(conf-if-po-20)# do show vlt 100 mismatch dhcpv6-relay
Global relay Configuration Mismatch
-----
DHCPv6 Hostname Mismatch:
VLT Unit ID          Hostname
-----
* 1                   Present (DELL)
  2                   Not Present

Remote-id Mismatch:
VLT Unit ID  Status          prefix          description
-----
* 1          -          hostname        default
  2          -          hostname-vrfname custom(force10)

Interface-id Mismatch:
VLT Unit ID  Status          prefix
-----
* 1          -          hostname
  2          -          custom(DELL)

Interface Relay Configuration Mismatch
-----
VLAN: 10
VLT Unit ID  description          Source-Interface
-----
* 1          default            -
  2          custom(santaclara) -
```

```
VNI: 20
VLT Unit ID      description      Source-Interface
-----
* 1              custom(force10)  Present
  2              default          Not Present

VLT-PORTCHANNEL: 100
VLT Unit ID      description      Source-Interface
-----
* 1              custom(force10)  Present
  2              custom(santaclara)  Not Present
Note : - Represent no mismatch.
```

**Supported Releases**

10.5.2.0 or later

## show vlt mismatch dhcp-relay

Displays the mismatch (if any), between the VLT peer for the DHCP relay options configuration on the Global level, VRF levels, and VLANs spanned across the VLT peers.

**Syntax** `show vlt vlt-domain mismatch dhcp-relay`

**Parameters** None.

**Defaults** None

**Command Mode** EXEC

**Usage Information** This command shows the mismatch in the global `ip dhcp-relay vss` and `ip dhcp-relay server-override` commands.

This command shows the codes for the mismatch between the VLT peers for the server-override configuration and `vss-info` values for the spanned VLANs or virtual-networks. This command also displays the presence or absence of VRF or interface level source-interface configurations whenever the interface IP address on the client-facing interface is not present.

This command is restricted to `netadmin` and `sysadmin` role users.

**Example (Interface)**

```
OS10(conf-if-po-20)# do show vlt 100 mismatch dhcp-relay

Global relay Configuration Mismatch
-----
VLT Unit ID      Link-Selection      Server-Override      VSS
-----
* 1              enabled             -                    disabled
  2              disabled           -                    enabled

VRF relay Configuration Mismatch
-----
VRF : VRF RED
VLT Unit ID      Source-Interface
-----
* 1              Present
  2              Not Present

Interface Relay Configuration Mismatch
-----
VLAN: 10
VLT Unit ID      Server-Override      VSS                    Source-Interface
-----
* 1              enabled             type-0 (Red)          -
  2              disabled           type-0 (Blue)        -
VNI: 20
VLT Unit ID      Server-Override      VSS                    Source-Interface
-----
```

* 1	-	type-0 (Red)	Present
2	-	type-1 (ABC:1234)	Not Present

**Supported Releases** 10.5.2 or later

## VXLAN MAC commands

### clear mac address-table dynamic nve remote-vtep

Clears all MAC addresses learned from a remote VTEP.

**Syntax** `clear mac address-table dynamic nve remote-vtep ip-address`

**Parameters** **remote-vtep** Clear MAC addresses learned from the specified remote VTEP.  
**ip-address**

**Default** Not configured

**Command mode** EXEC

**Usage information** To display the MAC addresses learned from a remote VTEP, use the `show mac address-table nve remote-vtep` command. Use this command to delete all MAC address entries learned from a remote VTEP.

**Example**

```
OS10# clear mac address-table dynamic nve remote-vtep 32.1.1.1
```

**Supported releases** 10.4.2.0 or later

### clear mac address-table dynamic virtual-network

Clears MAC addresses learned on all or a specified VXLAN virtual network.

**Syntax** `clear mac address-table dynamic virtual-network [interface {ethernet node/slot/port:subport | port-channel number} | local | vn-id [address mac-address | local]]`

**Parameters**

- interface** Clear all MAC addresses learned on the specified interface.
- ethernet node/slot/port[:subport]**
- interface port-channel number** Clear all MAC addresses learned on the specified port channel.
- virtual-network vn-id** Clear all MAC addresses learned on the specified virtual network, from 1 to 65535.
- local** Clear only locally-learned MAC addresses.
- vn-id** Clear learned MAC addresses on the specified virtual network, from 1 to 65535.
- vn-id local** Clear locally learned MAC addresses on the specified virtual network, from 1 to 65535.
- vn-id address mac-address** Clear only the MAC address entry learned in the specified virtual network. Enter the MAC address in `EEEE.EEEE.EEEE` format.

**Default** Not configured

<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command with no optional parameters to delete all dynamic MAC address entries that are learned only on virtual-network bridges from the MAC address table. This command does not delete MAC address entries learned on simple VLAN bridges. Use the <code>show mac address-table virtual-network</code> command to display the MAC addresses learned on a virtual network.
<b>Example</b>	<pre>OS10# clear mac address-table dynamic virtual-network</pre>
<b>Supported releases</b>	10.4.2.0 or later

## show mac address-table count extended

Displays the number of MAC addresses learned on all VLANs and VXLAN virtual networks.

<b>Syntax</b>	<code>show mac address-table count extended [interface {ethernet <i>node/slot/port:subport</i>   port-channel <i>number</i>}]</code>	
<b>Parameters</b>	<b>interface</b>	Display the number of MAC addresses learned on all VLANs and VXLANs on the specified interface.
	<b>ethernet</b>	
	<b>node/slot/</b>	
	<b>port[:subport]</b>	
	<b>]</b>	
	<b>interface</b>	Display the number of MAC addresses learned on all VLANs and VXLANs on the specified port channel.
	<b>port-channel</b>	
	<b>number</b>	
<b>Default</b>	Not configured	
<b>Command mode</b>	EXEC	
<b>Usage information</b>	Use this command to display the number of MAC address entries learned on all VLANs and VXLAN virtual networks.	
<b>Example</b>	<pre>OS10# show mac address-table count extended MAC Entries for all vlans : Dynamic Address Count :           10 Static Address (User-defined) Count :    2 Total MAC Addresses in Use:         12</pre>	
<b>Supported releases</b>	10.4.2.0 or later	

## show mac address-table count nve

Displays the number of MAC addresses learned on a VXLAN virtual network or from a remote VXLAN tunnel endpoint.

<b>Syntax</b>	<code>show mac address-table count nve {vxlan-vni <i>vni</i>   remote-vtep <i>ip-address</i>}</code>	
<b>Parameters</b>	<b>vxlan-vni <i>vni</i></b>	Display MAC addresses learned on the specified VXLAN virtual network, from 1 to 16,777,215.
	<b>remote-vtep</b>	Display MAC addresses learned from the specified remote VTEP.
	<b>ip-address</b>	
<b>Default</b>	Not configured	
<b>Command mode</b>	EXEC	
<b>Usage information</b>	Use the <code>clear mac address-table dynamic nve remote-vtep</code> command to delete all MAC address entries learned from a remote VTEP. Use the <code>clear mac address-table dynamic</code>	

`virtual-network vn-id` command to delete all dynamic MAC address entries learned on a virtual-network bridge.

### Example

```
OS10# show mac address-table count nve vxlan-vni 1001
MAC Entries for all vlans :
Dynamic Address Count :           1
Static Address (User-defined) Count : 0
Total MAC Addresses in Use:       1

OS10# show mac address-table count nve remote-vtep 32.1.1.1
MAC Entries for all vlans :
Dynamic Address Count :           2
Static Address (User-defined) Count : 0
Total MAC Addresses in Use:       2
```

### Supported releases

10.4.2.0 or later

## show mac address-table count virtual-network

Displays the number of MAC addresses learned on virtual networks.

### Syntax

```
show mac address-table count virtual-network [dynamic | local | remote |
static | interface {ethernet node/slot/port:subport | port-channel number}
| vn-id]
```

### Parameters

<b>dynamic</b>	Display the number of local dynamically-learned MAC addresses.
<b>local</b>	Display the number of local MAC addresses.
<b>remote</b>	Display the number of MAC addresses learned from remote VTEPs.
<b>static</b>	Display the number of local statically-configured MAC addresses.
<b>interface</b> <b>ethernet</b> <b><i>node/slot/</i></b> <b><i>port[:subport]</i></b> <b><i>]</i></b>	Display the number of MAC addresses learned on the specified interface.
<b>interface</b> <b>port-channel</b> <b><i>number</i></b>	Display the number of MAC addresses learned on the specified port channel.
<b><i>vn-id</i></b>	Display the number of MAC addresses learned on the specified virtual network, from 1 to 65535.

### Default

Not configured

### Command mode

EXEC

### Usage information

Use this command to display the number of MAC address entries learned on virtual networks in the MAC address table.

### Example

```
OS10# show mac address-table count virtual-network
MAC Entries for all vlans :
Dynamic Address Count :           8
Static Address (User-defined) Count : 0
Total MAC Addresses in Use:       8
```

### Supported releases

10.4.2.0 or later

## show mac address-table extended

Displays MAC addresses learned on all VLANs and VXLANs.

**Syntax** `show mac address-table extended [address mac-address | interface {ethernet node/slot/port:subport | port-channel number} | static | dynamic]`

**Parameters**

- address *mac-address*** Display only information about the specified MAC address.
- interface *ethernet node/slot/port[:subport]*** Display only MAC addresses learned on the specified interface.
- interface *port-channel number*** Display only MAC addresses learned on the specified port channel.
- static** Display only static MAC addresses.
- dynamic** Display only dynamic MAC addresses.

**Default** Not configured

**Command mode** EXEC

**Usage information** By default, MAC learning from a remote VTEP is enabled. Use this command to verify the MAC addresses learned both on VXLAN virtual networks and VLANs on the switch. The `show mac address-table` command displays the MAC addresses learned only on LAN port and VLAN interfaces.

### Example

```
OS10# show mac address-table extended
Virtual-Network  VlanId  MAC Address           Type           Interface/Remote-VTEP
-----
-                500     00:00:00:00:11:11    dynamic        ethernet1/1/31:1
-                500     00:00:00:00:44:44    dynamic        port-channel1000
-                1        aa:bb:cc:dd:f0:03    static         port-channel1000
-                500     aa:bb:cc:dd:f0:03    static         port-channel1000
-                4000    aa:bb:cc:dd:f0:03    static         port-channel1000
10000            10000   00:00:00:00:00:11    dynamic        ethernet1/1/31:1
10000            10000   00:00:00:00:00:44    dynamic        port-channel1000
10000            10000   00:00:00:00:00:55    dynamic        port-channel10
10000            10000   00:00:00:00:00:77    dynamic        VxLAN(32.1.1.1)
20000            300     00:00:00:00:00:22    dynamic        port-channel100
20000            300     00:00:00:00:00:33    dynamic        port-channel1000
20000            300     00:00:00:00:00:66    dynamic        port-channel10
20000            20000   00:00:00:00:00:88    dynamic        VxLAN(32.1.1.1)
```

**Supported releases** 10.4.2.0 or later

## show mac address-table nve

Displays MAC addresses learned on a VXLAN virtual network or from a remote VXLAN tunnel endpoint.

**Syntax** `show mac address-table nve {vxlan-vni vni | remote-vtep ip-address}`

**Parameters**

- vxlan-vni vni*** Display MAC addresses learned on the specified VXLAN virtual network, from 1 to 16,777,215.
- remote-vtep ip-address*** Display MAC addresses learned from the specified remote VTEP.

**Default** Not configured

**Command mode** EXEC

## Usage information

Use the `clear mac address-table dynamic nve remote-vtep 32.1.1.1` command to delete all MAC address entries learned from a remote VTEP. Use the `clear mac address-table dynamic virtual-network vn-id` command to delete all dynamic MAC address entries learned on a virtual-network bridge.

## Example

```
OS10# show mac address-table nve remote-vtep 32.1.1.1
Virtual-Network VNI      MAC Address      Type      Remote-VTEP
-----
10000           9999  00:00:00:00:00:77  dynamic  VxLAN (32.1.1.1)
20000           19999 00:00:00:00:00:88  dynamic  VxLAN (32.1.1.1)

OS10# show mac address-table nve vxlan-vni 9999
Virtual-Network VNI      MAC Address      Type      Remote-VTEP
-----
10000           9999  00:00:00:00:00:77  dynamic  VxLAN (32.1.1.1)
```

## Supported releases

10.4.2.0 or later

# show mac address-table virtual-network

Displays the MAC addresses learned on all or a specified virtual network.

## Syntax

```
show mac address-table virtual-network [vn-id | local | remote | static |
dynamic | address mac-address | interface {ethernet node/slot/port:subport
| port-channel number}]
```

## Parameters

<b>vn-id</b>	Display only information about the specified virtual network.
<b>local</b>	Display only locally learned MAC addresses.
<b>remote</b>	Display only remote MAC addresses.
<b>static</b>	Display only static MAC addresses.
<b>dynamic</b>	Display only dynamic MAC addresses.
<b>address mac-address</b>	Display only information about the specified MAC address. Enter the MAC address in <i>EEEE.EEEE.EEEE</i> format.
<b>interface ethernet node/slot/port[:subport]</b>	Display only MAC addresses learned on the specified interface.
<b>interface port-channel number</b>	Display only MAC addresses learned on the specified port channel.

## Default

Not configured

## Command mode

EXEC

## Usage information

Use this command to verify the MAC addresses learned on VXLAN virtual networks. By default, MAC learning from a remote VTEP is enabled.

## Example

```
OS10# show mac address-table virtual-network
Virtual-Network VlanId  MAC Address      Type      Interface/Remote-VTEP
-----
10000           10000  00:00:00:00:00:11  dynamic  ethernet1/1/31:1
10000           100    00:00:00:00:00:44  dynamic  port-channel1000
10000           100    00:00:00:00:00:55  dynamic  port-channel10
10000           10000  00:00:00:00:00:77  dynamic  VxLAN (32.1.1.1)
10000           100    34:a0:a0:a1:a2:f6  dynamic  port-channel10
20000           300    00:00:00:00:00:22  dynamic  port-channel100
20000           300    00:00:00:00:00:33  dynamic  port-channel1000
```

20000	300	00:00:00:00:00:66	dynamic	port-channel10
20000		00:00:00:00:00:88	dynamic	VxLAN (32.1.1.1)
20000	300	34:a0:a0:a1:a2:f6	dynamic	port-channel10

**Supported releases** 10.4.2.0 or later

## VXLAN BGP commands

### arp-nd-suppression

Enables or disables ARP-suppression globally on Layer2 and Layer 3 virtual networks.

**Syntax** [no] arp-nd-suppression disable

**Parameters** None.

**Default** Enabled by default after EVPN is enabled.

**Command Mode** EVPN

**Security and access** sysadmin, netadmin, netoperator

**Usage Information** You can use this command to enable or disable ARP-suppression feature globally. This configuration is applied on all Layer 2 Layer 3 VXLAN bridges.

The no arp-nd-suppression disable command enables ARP-suppression.

**Example (EVPN)** ARP-suppression is enabled internally by default, after EVPN is enabled:

```
OS10 (conf) # evpn
OS10 (conf-evpn) #
```

**Example (disable)** To disable ARP-suppression, use the following command:

```
OS10 (conf) # evpn
OS10 (conf-evpn) # arp-nd-suppression disable
```

**Example (enable)** To enable ARP-suppression globally, use the following command:

```
OS10 (conf) # evpn
OS10 (conf-evpn) # no arp-nd-suppression disable
```

**Supported Releases** 10.5.3 or later

### activate (l2vpn evpn)

Enables the exchange of L2 VPN EVPN address family information with a BGP neighbor or peer group.

**Syntax** activate

**Parameters** None

**Default** Not configured

**Command Mode** ROUTER-BGP-NEIGHBOR-AF

**Usage Information** Use this command to exchange L2 VPN EVPN address information for VXLAN host-based routing with a BGP neighbor. The IPv4 unicast address family is enabled by default. Use the `no activate` command to disable an address family with a neighbor.

**Example**

```
OS10(config-router-neighbor)# address-family l2vpn evpn unicast
OS10(config-router-bgp-neighbor-af)# activate
```

**Supported Releases** 10.2.0E or later

## address-family l2vpn evpn

Configures the L2 VPN EVPN address family for VXLAN host-based routing to a BGP neighbor.

**Syntax** `address-family l2vpn evpn`

**Parameters** None

**Default** Not configured

**Command mode** ROUTER-NEIGHBOR

**Usage information** To use BGP EVPN service in a VXLAN, you must configure and enable the L2VPN EVPN address family on a VTEP to support host-based routing to each BGP neighbor.

**Example**

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# neighbor 45.0.0.1
OS10(config-router-neighbor)# address-family l2vpn evpn
```

**Supported releases** 10.4.2.0 or later

## allowas-in

Configures the number of times the local AS number can appear in the BGP AS\_PATH path attribute before the switch rejects the route.

**Syntax** `allowas-in as-number`

**Parameters** `as-number`—Enter the number of occurrences for a local AS number, from 1 to 10.

**Default** Disabled

**Command Mode** ROUTER-BGP-NEIGHBOR-AF

**Usage Information** Use this command to enable the BGP speaker to accept a route with the local AS number in updates received from a peer for the specified number of times. The `no` version of this command resets the value to the default.

**Example (IPv4)**

```
OS10(config-router-neighbor)# address-family ipv4 unicast
OS10(config-router-bgp-neighbor-af)# allowas-in 5
```

**Example (IPv6)**

```
OS10(config-router-template)# address-family ipv6 unicast
OS10(config-router-bgp-template-af)# allowas-in 5
```

**Example (l2vpn)**

```
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-bgp-neighbor-af)# allowas-in 3
```

**Supported Releases** 10.3.0E or later

## sender-side-loop-detection

Enables the sender-side loop detection process for a BGP neighbor.

**Syntax** `sender-side-loop-detection`

**Parameters** None

**Default** Enabled

**Command Mode** ROUTER-BGP-NEIGHBOR-AF

**Usage Information** This command helps detect routing loops, based on the AS path before it starts advertising routes. To configure a neighbor to accept routes use the `neighbor allowas-in` command. The `no` version of this command disables sender-side loop detection for that neighbor.

### Example (IPv4)

```
OS10(conf-router-bgp-102)# neighbor 3.3.3.1
OS10(conf-router-neighbor)# address-family ipv4 unicast
OS10(conf-router-bgp-neighbor-af)# sender-side-loop-detection
```

### Example (IPv6)

```
OS10(conf-router-bgp-102)# neighbor 32::1
OS10(conf-router-neighbor)# address-family ipv6 unicast
OS10(conf-router-bgp-neighbor-af)# no sender-side-loop-detection
```

**Supported Releases** 10.3.0E or later

## show evpn arp-nd-suppression

Shows the global ARP-suppression enabled or disabled status.

**Syntax** `show evpn arp-nd-suppression`

**Parameters** None.

**Default** None.

**Command Mode** EXEC

**Security and access** sysadmin, netadmin, netoperator

**Usage Information** Use this command to display the global ARP-suppression enabled or disabled status.

### Example

```
OS10# show evpn arp-nd-suppression
ARP-ND Suppression Status : Enabled

OS10# show evpn arp-nd-suppression
ARP-ND Suppression Status : Disabled
```

**Supported Releases** 10.5.3 or later

## show ip bgp l2vpn evpn

Displays the internal BGP routes in the L2VPN EVPN address family in EVPN instances.

**Syntax** `show ip bgp l2vpn evpn [summary | neighbors [ip-address | interface interface-type]]`

## Parameters

<b>summary</b>	Displays a summary of the BGP routes in the L2VPN address family that exchange with remote VTEPs.
<b>neighbors</b>	Display the remote VTEPs with whom BGP routes in the L2VPN address family exchange.
<b>ip-address</b>	Displays information about a specific neighbor.
<b>interface</b> <b>interface-</b> <b>type</b>	Displays BGP information that is learned through an unnumbered neighbor.

## Default

Not configured

## Command mode

EXEC

## Usage information

Use this command to display the BGP routes used for the L2VPN EVPN address family in EVPN instances on the switch.

## Examples

```
OS10# show ip bgp l2vpn evpn
BGP local RIB : Routes to be Added , Replaced , Withdrawn
BGP local router ID is 110.111.170.102
Status codes: s suppressed, S stale, d dampened, h history, * valid, >
best
Path source: I - internal, a - aggregate, c - confed-external,
r - redistributed/network, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network                               Next Hop           Metric   LocPrf
Weight  Path
*>r      Route distinguisher: 110.111.170.102:65447
[3]:[0]:[32]:[110.111.170.102]/152         110.111.170.102   0         100
32768   ?
*>      Route distinguisher: 110.111.170.107:64536
[3]:[0]:[32]:[110.111.170.107]/152         110.111.170.107   0         100
0       100 101 ?
```

```
OS10# show ip bgp l2vpn evpn summary
BGP router identifier 2.2.2.2 local AS number 4294967295
Neighbor AS      MsgRcvd      MsgSent      Up/Down
State/Pfx
3.3.3.3  4294967295  2831         9130         05:57:27      504
4.4.4.4  4294967295  2364         9586         05:56:43      504
5.5.5.5  4294967295  4947         8399         01:10:39     11514
6.6.6.6  4294967295  2413         7310         05:51:56      504
```

```
OS10# show ip bgp l2vpn evpn neighbors
BGP neighbor is 3.3.3.3, remote AS 4294967295, local AS 4294967295
internal link

BGP version 4, remote router ID 3.3.3.3
BGP state ESTABLISHED, in this state for 06:21:55
Last read 00:37:43 seconds
Hold time is 180, keepalive interval is 60 seconds
Configured hold time is 180, keepalive interval is 60 seconds
Fall-over disabled
Route reflector client

Received 2860 messages
  1 opens, 0 notifications, 2422 updates
  437 keepalives, 0 route refresh requests
Sent 32996 messages
  1 opens, 0 notifications, 32565 updates
  430 keepalives, 0 route refresh requests
Minimum time between advertisement runs is 30 seconds
Minimum time before advertisements start is 0 seconds

Capabilities received from neighbor for IPv4 Unicast:
  ROUTE_REFRESH(2)
  CISCO_ROUTE_REFRESH(128)
```

```
4 OCTET AS(65)
MP_L2VPN_EVPN(1)
Capabilities advertised to neighbor for IPv4 Unicast:
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
4 OCTET AS(65)
MP_L2VPN_EVPN(1)
Prefixes accepted 504, Prefixes advertised 13012
Connections established 1; dropped 0
Last reset never
Local host: 2.2.2.2, Local port: 37853
Foreign host: 3.3.3.3, Foreign port: 179
<Output Truncated>
```

```
OS10# show ip bgp l2vpn evpn neighbors interface vlan 30

BGP neighbor is fe80::76e6:e2ff:fef6:99a9 via vlan30, remote AS 100,
local AS 200 external link

BGP version 4, remote router ID 125.12.57.117
BGP state ESTABLISHED, in this state for 00:15:52
Last read 00:21:08 seconds
Hold time is 180, keepalive interval is 60 seconds
Configured hold time is 180, keepalive interval is 60 seconds
Fall-over disabled

Received 20 messages
    1 opens, 0 notifications, 0 updates
    19 keepalives, 0 route refresh requests
Sent 20 messages
    1 opens, 1 notifications, 0 updates
    18 keepalives, 0 route refresh requests
Minimum time between advertisement runs is 30 seconds
Minimum time before advertisements start is 0 seconds

Capabilities received from neighbor for IPv4 Unicast:
MULTIPROTO_EXT(1)
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
4_OCTET_AS(65)
MP_L2VPN_EVPN(1)
Extended Next Hop Encoding (5)

Capabilities advertised to neighbor for IPv4 Unicast:
MULTIPROTO_EXT(1)
ROUTE_REFRESH(2)
CISCO_ROUTE_REFRESH(128)
```

```

4_OCTET_AS (65)
MP_L2VPN_EVPN (1)
Extended Next Hop Encoding (5)
Prefixes accepted 0, Prefixes advertised 0
Connections established 1; dropped 0
Last reset never
Prefixes ignored due to:
Martian address 0, Our own AS in AS-PATH 0
Invalid Nexthop 0, Invalid AS-PATH length 0
Wellknown community 0, Locally originated 0

Local host: fe80::76e6:e2ff:fef5:a43e, Local port: 45926
Foreign host: fe80::76e6:e2ff:fef6:99a9, Foreign port: 179

```

```

OS10# show ip bgp l2vpn evpn summary
BGP router identifier 89.101.17.125 local AS number 100
Neighbor                AS                MsgRcvd
  MsgSent      Up/Down          State/Pfx      200
ethernet1/1/1      19              00:15:34      0              19

```

**Supported releases** 10.4.2.0 or later

## VXLAN commands

### arp-nd-suppression

Enables or disables ARP-suppression globally on Layer2 and Layer 3 virtual networks.

**Syntax** [no] arp-nd-suppression disable

**Parameters** None.

**Default** Enabled by default after EVPN is enabled.

**Command Mode** EVPN

**Security and access** sysadmin, netadmin, netoperator

**Usage Information** You can use this command to enable or disable ARP-suppression feature globally. This configuration is applied on all Layer 2 Layer 3 VXLAN bridges.

The no arp-nd-suppression disable command enables ARP-suppression.

**Example (EVPN)** ARP-suppression is enabled internally by default, after EVPN is enabled:

```

OS10 (conf) # evpn
OS10 (conf-evpn) #

```

**Example (disable)**

To disable ARP-suppression, use the following command:

```
OS10 (conf) # evpn
OS10 (conf-evpn) # arp-nd-suppression disable
```

**Example (enable)** To enable ARP-suppression globally, use the following command:

```
OS10 (conf) # evpn
OS10 (conf-evpn) # no arp-nd-suppression disable
```

**Supported Releases**

10.5.3 or later

## hardware overlay-routing-profile

Configures the number of reserved ARP table entries for VXLAN overlay routing.

**Syntax**

```
hardware overlay-routing-profile {balanced-overlay-routing | scaled-  
overlay-routing | disable-overlay-routing}
```

**Parameters****balanced-  
overlay-  
routing**

Reserve routing entries for balanced VXLAN tenant routing:

- S4048T-ON and S6010-ON: 24576 entries
- S4100-ON series: 16384 entries
- S5200-ON series switches: 32768 entries

**scaled-  
overlay-  
routing**

Reserve routing entries for scaled VXLAN tenant routing:

- S4048T-ON and S6010-ON: 36864 entries
- S4100-ON series: 24576 entries
- S5200-ON series switches: 53248 entries

**disable-  
overlay-  
routing**

Allocate 0 next-hop entries for overlay routing and all next-hop entries for underlay routing.

**Default**

S4048T-ON and S6010-ON switches reserve 8192 ARP table entries.

S4100-ON series switches reserve 4096 ARP table entries.

S5200-ON series switches reserve 8192 ARP table entries.

**Command mode**

CONFIGURATION

**Usage information**

The number of reserved table entries in a profile varies according to the OS10 switch. To view the available overlay routing profiles for a switch, use the `show hardware overlay-routing-profile mode all` command. After you configure a profile, reload the switch to activate the profile. The `no` form of the command disables the configured profile and restores the default number of reserved ARP table entries.

**Example**

```
OS10(config)# hardware overlay-routing-profile balanced-overlay-routing
OS10(config)# exit
OS10# write memory
OS10# reload
```

**Supported releases**

10.4.3.0 or later

## interface virtual-network

Configures a virtual-network router interface.

<b>Syntax</b>	<code>interface virtual-network <i>vn-id</i></code>
<b>Parameters</b>	<b>virtual-network <i>vn-id</i></b> Enter a virtual-network ID, from 1 to 65535.
<b>Default</b>	Not configured
<b>Command mode</b>	CONFIGURATION
<b>Usage information</b>	Configure a virtual-network router interface to enable hosts connected to a virtual network to route traffic to hosts on another virtual network in the same VRF. The virtual-network IP address must be unique on each VTEP, including VTEPs in VLT pairs.
<b>Example</b>	<pre>OS10(config)# interface virtual-network 10000 OS10(config-if-vn-10000)# ip vrf forwarding tenant1 OS10(config-if-vn-10000)# ip address 10.1.0.1/16 OS10(config-if-vn-10000)# no shutdown</pre>
<b>Supported releases</b>	10.4.3.0 or later

## ip virtual-router address

Configures an anycast gateway IP address for a VXLAN virtual network.

<b>Syntax</b>	<code>ip virtual-router address <i>ip-address</i></code>
<b>Parameters</b>	<b>address <i>ip-address</i></b> Enter the IP address of the anycast L3 gateway.
<b>Default</b>	Not configured
<b>Command mode</b>	INTERFACE-VIRTUAL-NETWORK
<b>Usage information</b>	Configure the same anycast gateway IP address on all VTEPs in a VXLAN virtual network. Use the anycast gateway IP address as the default gateway IP address if the host VMs move from one VTEP to another in a VXLAN. The anycast gateway IP address must be in the same subnet as the IP address of the virtual-network router interface.
<b>Example</b>	<pre>OS10(config)# interface virtual-network 10000 OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100</pre>
<b>Supported releases</b>	10.4.3.0 or later

## ip virtual-router mac-address

Configures the MAC address of an anycast L3 gateway for VXLAN routing.

<b>Syntax</b>	<code>ip virtual-router mac-address <i>mac-address</i></code>
<b>Parameters</b>	<b>mac-address <i>mac-address</i></b> Enter the MAC address of the anycast L3 gateway.
<b>Default</b>	Not configured
<b>Command mode</b>	CONFIGURATION

**Usage information** Configure the same MAC address on all VTEPs so that the anycast gateway MAC address remains the same if a VM migrates to a different VTEP. Because the configured MAC address is automatically used for all VXLAN virtual networks, configure it in global Configuration mode.

**Example**

```
OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01
```

**Supported releases** 10.4.3.0 or later

## member-interface

Assigns untagged or tagged VLAN traffic on a member interface to a virtual network.

**Syntax** `member-interface {ethernet node/slot/port[:subport] | port-channel number} {vlan-tag vlan-id | untagged}`

**Parameters**

<b>ethernet</b> <i>node/slot/port[:subport]</i>	Assign the specified interface to a virtual network.
<b>port-channel</b> <i>number</i>	Assign the specified port channel to a virtual network.
<b>untagged</b>	Assign untagged traffic on an interface or port channel to a virtual network.
<b>vlan-tag</b> <i>vlan-id</i>	Assign tagged traffic on the specified VLAN to a virtual network.

**Default** Not configured

**Command mode** VIRTUAL-NETWORK

**Usage information** Use this command to assign traffic on the same VLAN or interface to different virtual networks. The no version of this command removes the configured value.

**Example**

```
OS10(config)# virtual-network 10000
OS10(config-vn)# member-interface port-channel 10 vlan-tag 200
OS10(config-vn)# member-interface port-channel 20 untagged
```

**Supported releases** 10.4.2.0 or later

## nve

Enters network virtualization edge (NVE) configuration mode to configure the source VXLAN tunnel endpoint.

**Syntax** `nve`

**Parameters** None

**Default** None

**Command mode** CONFIGURATION

**Usage information** In NVE mode, configure the source tunnel endpoint for all virtual networks on the switch.

**Example**

```
OS10# nve
OS10(config-nve)#
```

**Supported releases** 10.4.2.0 or later

## remote-vtep

Configures the IP address of a remote tunnel endpoint in a VXLAN network.

- Syntax** `remote-vtep ip-address`
- Parameters** `ip-address` — Enter the IP address of a remote virtual tunnel endpoint (VTEP).
- Default** Not configured
- Command mode** VIRTUAL-NETWORK VXLAN-VNI
- Usage information** After you configure the remote VTEP, the VXLAN virtual network is enabled to start sending server traffic. You can configure multiple remote VTEPs. All broadcast, multicast, and unknown unicast (BUM) traffic received on an access interface is replicated on remote VTEPs. The `no` version of this command removes the configured value.

**Example**

```
OS10(config-vn-vxlan-vni)# remote-vtep 20.20.20.1
OS10(config-vn-vxlan-vni-remote-vtep)# exit
OS10(config-vn-vxlan-vni)# remote-vtep 30.20.20.1
```

**Supported releases** 10.4.2.0 or later

## show hardware overlay-routing-profile mode

Displays the number of hardware resources available for overlay routing in different profiles.

- Syntax** `show hardware overlay-routing-profile mode [all]`
- Parameters** `all` View the number of tenant entries available in each hardware partition for overlay routing profiles.
- Default** Not configured
- Command mode** EXEC
- Usage information** On S4100-ON series, S5200-ON series, S4048T-ON, S4248-ON, and S6010-ON switches, L3 VXLAN overlay routing requires reserved hardware resources. The number of reserved table entries in a profile varies according to the OS10 switch.

**Example (S5200-ON series)**

```
OS10# show hardware overlay-routing-profile mode all
Mode                Overlay Underlay  Overlay Underlay
                   Next-hop Next-hop L3 RIF  L3 RIF
                   Entries  Entries  Entries Entries
default-overlay-routing 8192    57344    2048    14336
disable-overlay-routing 0        65536    0        16384
balanced-overlay-routing 32768   32768    8192    8192
scaled-overlay-routing 53248   12288    12288   4096

show hardware overlay-routing-profile mode
Setting  Mode                Overlay Underlay  Overlay Underlay
                   Next-hop Next-hop L3 RIF  L3 RIF
                   Entries  Entries  Entries Entries
Current  default-overlay-routing 8192    57344    2048    14336
Next-boot default-overlay-routing 8192    57344    2048    14336
```

**Supported releases** 10.4.3.0 or later

## show interface virtual-network

Displays the configuration of virtual-network router interfaces and packet statistics.

<b>Syntax</b>	<code>show interface virtual-network [vn-id]</code>
<b>Parameters</b>	<b>vn-id</b> Enter a virtual-network ID, from 1 to 65535.
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command to display the virtual-network IP address used for routing traffic in a virtual network. Traffic counters also display.
<b>Example</b>	

```
show interface virtual-network 102
Virtual-network 102 is up, line protocol is up
Address is 14:18:77:25:6f:84, Current address is 14:18:77:25:6f:84
Interface index is 66
Internet address is 12.12.12.2/24
Mode of IPv4 Address Assignment: MANUAL
Interface IPv6 oper status: Enabled
Link local IPv6 address: fe80::1618:77ff:fe25:6eb9/64
MTU 1532 bytes, IP MTU 1500 bytes
ARP type: ARPA, ARP Timeout: 60
Last clearing of "show interface" counters: 10:24:21
Queuing strategy: fifo
Input statistics:
    89 packets, 10056 octets
Output statistics:
    207 packets, 7376 octets
Time since last interface status change: 10:23:21
```

**Supported releases** 10.4.3.0 or later

## show nve remote-vtep

Displays information about remote VXLAN tunnel endpoints.

<b>Syntax</b>	<code>show nve remote-vtep [ip-address   summary   counters]</code>
<b>Parameters</b>	<b>ip-address</b> Display detailed information about a specified remote VTEP. <b>summary</b> Display summary information about remote VTEPs. <b>counters</b> Display statistics on remote VTEP traffic.
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command to display the IP address, operational state, and configured VXLANs for each remote VTEP. The remote MAC learning and unknown unicast drop settings used for each VXLAN ID (VNI) also display.
<b>Example</b>	

```
OS10# show nve remote-vtep summary
Remote-VTEP      State
-----
2.2.2.2          up
```

```
OS10# show nve remote-vtep
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD -
Unknown-Unicast-Drop
```

```
IP Address: 2.2.2.2, State: up, Encap: VxLAN
VNI list: 10000 (DP), 200 (DP), 300 (DP)
```

**Supported releases** 10.4.2.0 or later

## show nve remote-vtep counters

Displays VXLAN packet statistics for a remote VTEP.

**Syntax** `show nve remote-vtep [ip-address] counters`

**Parameters** • `ip-address` — Enter IP address of a remote VTEP.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to display input and output statistics for VXLAN traffic on a remote VTEP. A VTEP is identified by its IP address.

Use the `clear nve remote-vtep [ip-address] counters` command to clear VXLAN packet statistics.

### Example

```
OS10# show nve remote-vtep counters
Peer          Input (Packets/Bytes)  Output (Packets/Bytes)
10.10.10.10   857/8570              257/23709
20.20.20.20   457/3570              277/13709
```

**Supported releases** 10.4.2.0 or later

## show nve vxlan-vni

Displays information about the VXLAN virtual networks on the switch.

**Syntax** `show nve vxlan-vni`

**Parameters** None

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to display information about configured VXLAN virtual networks. Each VXLAN virtual network is identified by its virtual-network ID.

### Example

```
OS10# show nve vxlan-vni
VNI          Virtual-Network  Source-IP  Remote-VTEPs
-----
10000        1                1.1.1.1    2.2.2.2
200          2                1.1.1.1    2.2.2.2
300          300              1.1.1.1    2.2.2.2
```

**Supported releases** 10.4.2.0 or later

## show virtual-network

Displays a virtual-network configuration, including all VXLAN configurations.

**Syntax** `show virtual-network [vn-id]`

**Parameters** `vn-id` Enter a virtual-network ID, from 1 to 65535.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to display the VNID, port members, source interface, and remote tunnel endpoints of a VXLAN virtual network.

**Example**

```
OS10# show virtual-network
Codes: DP - MAC-learn Dataplane, CP - MAC-learn Controlplane, UUD -
Unknown-Unicast-Drop
Un-tagged VLAN: 888
Virtual Network: 60000
  VLTi-VLAN: 2500
  Members:
    VLAN 1000: port-channell, ethernet1/1/9, ethernet1/1/10
    VLAN 2500: port-channell000
  VxLAN Virtual Network Identifier: 16775000
  Source Interface: loopback100(222.222.222.222)
  Remote-VTEPs (flood-list): 55.55.55.55(DP),77.1.1.1(DP)
```

**Supported releases** 10.4.2.0 or later

## show virtual-network counters

Displays packet statistics for virtual networks.

**Syntax** `show virtual-network [vn-id] counters`

**Parameters** `vn-id` Enter a virtual-network ID, from 1 to 65535.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to monitor the packet throughput on virtual networks, including VXLANs. Use the `clear virtual-network counters` command to clear virtual-network counters.

**Example**

```
OS10# show virtual-network counters
Virtual-Network      Input (Packets/Bytes)      Output (Packets/Bytes)
1000                  857/8570                   257/23709
2000                  457/3570                   277/13709
```

**Supported releases** 10.4.2.0 or later

## show virtual-network interface counters

Displays packet statistics for a member port, port channel, or VLAN in VXLAN virtual networks.

**Syntax** `show virtual-network interface {ethernet node/slot/port:subport | port-channel number} [vlan vlan-id] counters`

**Parameters** `interface` Enter the port information for an Ethernet interface.  
`ethernet`  
`node/slot/`  
`port[:subport]`  
`]`

**interface**            Enter a port-channel number, from 1 to 999 or 1001 to 2000.  
**port-channel**  
**number**

**vlan** *vlan-id*    (Optional) Enter a VLAN ID, from 1 to 4093.

**Default**            Not configured

**Command mode**    EXEC

**Usage information**    Use this command to monitor the packet throughput on a port interface that is a member of a VXLAN virtual network. Assign a VLAN member interface to only one virtual network. To clear VXLAN packet counters on a member port or VLAN members of a virtual network, use the `clear virtual-network interface {ethernet node/slot/port:subport | port-channel number} [vlan vlan-id] counters` command.

**Example**

```
OS10# show virtual-network interface 1/1/3 vlan 100 counters
Virtual-Network      Input (Packets/Bytes)      Output (Packets/Bytes)
2000                  457/3570                   277/13709
```

**Supported releases**    10.4.2.0 or later

## show virtual-network interface

Displays the VXLAN virtual networks and server VLANs where a port is assigned.

**Syntax**            `show virtual-network interface {ethernet node/slot/port:subport | port-channel number}`

**Parameters**

**interface**            Enter the port information for an Ethernet interface.  
**ethernet**  
**node/slot/**  
**port[:subport**  
**]**

**interface**            Enter a port-channel number, from 1 to 999 or 1001 to 2000.  
**port-channel**  
**number**

**Default**            Not configured

**Command mode**    EXEC

**Usage information**    Use this command to verify the VXLAN VLANs where an Ethernet port connected to downstream servers is a member.

**Example**

```
OS10# show virtual-network interface ethernet 1/1/1
Interface      Vlan      Virtual-network
ethernet1/1/1  100       1000
ethernet1/1/1  200       2000
ethernet1/1/1  300       3000
```

**Supported releases**    10.4.2.0 or later

## show virtual-network vlan

Displays the VXLAN virtual networks where a VLAN is assigned.

**Syntax**            `show virtual-network vlan vlan-id`

<b>Parameters</b>	<b>vlan <i>vlan-id</i></b> Enter a VLAN ID, from 1 to 4093.
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command to verify the VXLAN virtual networks where a VLAN is assigned, including the port members connected to downstream servers.
<b>Example</b>	<pre>OS10# show show virtual-network 100 Vlan  Virtual-network  Interface 100    1000                ethernet1/1/1,ethernet1/1/2</pre>
<b>Supported releases</b>	10.4.2.0 or later

## show vlan (virtual network)

Displays the VLANs assigned to virtual networks.

<b>Syntax</b>	show vlan
<b>Parameters</b>	None
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command to display the VLAN port interfaces that transmit VXLAN packets over a virtual network.
<b>Example</b>	<pre>OS10# show vlan Codes: * - Default VLAN, M - Management VLAN, R - Remote Port Mirroring VLANs, @ - Attached to Virtual Network Q: A - Access (Untagged), T - Tagged    NUM  Status  Description  Q  Ports *  1    up       Eth1/1/1-1/1/48  A  Eth1/1/1-1/1/48 @ 100  up       Eth1/1/2,Eth1/1/3  T  Eth1/1/2,Eth1/1/3 @ 101  up       Eth1/1/1          A  Eth1/1/1   200  up       port-channel5     T  port-channel5   200  up       Eth1/1/11-1/1/15  T  Eth1/1/11-1/1/15</pre>
<b>Supported releases</b>	10.4.2.0 or later

## source-interface loopback

Configures a dedicated Loopback interface as the source VTEP.

<b>Syntax</b>	source-interface loopback <i>number</i>
<b>Parameters</b>	<b>loopback <i>number</i></b> Enter the Loopback interface used as the source interface of a VXLAN virtual tunnel, from 0 to 16383.
<b>Default</b>	Not configured
<b>Command mode</b>	NVE-INSTANCE
<b>Usage information</b>	<p>The IP address of the Loopback interface serves as the source IP address in encapsulated packets transmitted from the switch as an NVE VTEP.</p> <ul style="list-style-type: none"> <li>The Loopback interface must have an IP address configured. The Loopback IP address must be reachable from the remote VTEP.</li> </ul>

- You cannot change the source interface if at least one VXLAN virtual network ID (VNID) is configured for the NVE instance.

Use this command in NVE mode to override a previously configured value and reconfigure the source IP address. The `no` version of this command removes the configured value.

#### Examples

```
OS10(config-nve)# source-interface loopback 1
```

#### Supported releases

10.4.2.0 or later

## virtual-network

Creates a virtual network for VXLAN tunneling.

#### Syntax

```
virtual-network vn-id
```

#### Parameters

***vn-id*** Enter the virtual-network ID, from 1 to 65535.

#### Default

Not configured

#### Command mode

CONFIGURATION

#### Usage information

The virtual network operates as a L2 bridging domain. To add a VXLAN to the virtual network, use the `vxlan-vni` command. The `no` version of this command removes the configured virtual network.

#### Example

```
OS10(config)# virtual-network 1000
OS10(config-vn)#
```

#### Supported releases

10.4.2.0 or later

## virtual-network untagged-vlan

Configures a dedicated VLAN for internal use to transmit untagged traffic on member ports in virtual networks on the switch.

#### Syntax

```
virtual-network untagged-vlan vlan-id
```

#### Parameters

***id*** Enter the reserved untagged VLAN ID, from 1 to 4093.

#### Default

Not configured

#### Command mode

CONFIGURATION

#### Usage information

The untagged VLAN ID is used internally for all untagged member interfaces that belong to virtual networks. You cannot use the reserved untagged VLAN ID for a simple VLAN bridge or for tagged traffic on member interfaces of virtual networks. The `no` version of this command removes the configured value.

#### Example

```
OS10(config)# virtual-network untagged-vlan 10
```

#### Supported releases

10.4.2.0 or later

## vxlan-vni

Assigns a VXLAN ID to a virtual network.

#### Syntax

```
vxlan-vni vni
```

<b>Parameters</b>	<b>vni</b> Enter the VXLAN ID for a virtual network, from 1 to 16,777,215.
<b>Default</b>	Not configured
<b>Command mode</b>	VIRTUAL-NETWORK
<b>Usage information</b>	This command associates a VXLAN ID number with a virtual network. The <code>no</code> version of this command removes the configured ID.
<b>Example</b>	<pre>OS10(conf-vn-100)# vxlan-vni 100 OS10(config-vn-vxlan-vni)#</pre>
<b>Supported releases</b>	10.4.2.0 or later

## VXLAN EVPN commands

### advertise

Advertises the IP prefixes learned from external networks and directly connected neighbors into EVPN.

<b>Syntax</b>	<code>advertise {ipv4   ipv6} {connected   static   ospf   bgp} [route-map <i>map-name</i>]</code>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ipv4</code> — Advertise learned IPv4 routes.</li> <li>• <code>ipv6</code> — Advertise learned IPv6 routes.</li> <li>• <code>connected</code> — Advertise routes learned from directly connected neighbors.</li> <li>• <code>static</code> — Advertise manually configured routes.</li> <li>• <code>ospf</code> — Advertise OSPF routes into EVPN.</li> <li>• <code>bgp</code> — Advertise BGP learned external routes into EVPN.</li> <li>• <code>route-map <i>map-name</i></code> — (Optional) Filter EVPN Type-5 advertised routes using the specified route map. You can add the match rule <code>inactive-path-additive</code> to the route map to advertise inactive routes.</li> </ul>
<b>Default</b>	None
<b>Command Mode</b>	EVPN-VRF
<b>Usage Information</b>	EVPN uses Type 5 route advertisements. To specify the types of learned routes to use in EVPN Type 5 advertisements in a tenant VRF, use the <code>advertise</code> command. From Release 10.5.2.0 and beyond, the <code>advertise</code> command advertises only active routes. To advertise both the active and inactive routes, you must configure a route map with the <code>inactive-path-additive</code> rule and apply the route map to the <code>advertise</code> command.
<b>Example – advertise active routes</b>	<pre>OS10(config)# evpn OS10(config-evpn)# vrf vrf-blue OS10(config-evpn-vrf-vrf-blue)# advertise ipv4 connected route-map map-connected</pre>
<b>Example - advertise IPv4 static routes to L2VPN EVPN</b>	<pre>OS10# configure terminal OS10(config)# route-map redis-inactive-routes OS10(config-route-map)# match inactive-path-additive  OS10(config)# evpn OS10(config-evpn)# vrf vrf-blue OS10(config-evpn-vrf-vrf-blue)# advertise ipv4 static route-map redis-inactive-routes</pre>

### Example - advertise IPv6 OSPF routes to L2VPN EVPN

```
OS10# configure terminal
OS10(config)# route-map redis-inactive-routes
OS10(config-route-map)# match inactive-path-additive

OS10(config)# evpn
OS10(config-evpn)# vrf vrf-blue
OS10(config-evpn-vrf-vrf-blue)# advertise ipv6 ospf route-map redis-
inactive-routes
```

**Supported  
Releases** 10.5.1 or later

## auto-evi

Creates an EVPN instance automatically, including Route Distinguisher (RD) and Route Target (RT) values.

<b>Syntax</b>	auto-evi
<b>Parameters</b>	None
<b>Default</b>	Not configured
<b>Command mode</b>	EVPN
<b>Usage information</b>	<p>In deployments running BGP with 2-byte or 4-byte autonomous systems, auto-EVI automatically creates EVPN instances when you create a virtual network on a VTEP in the overlay network. In auto-EVI mode, the RD and RT values automatically generate:</p> <ul style="list-style-type: none"><li>• For a 2-byte autonomous system:<ul style="list-style-type: none"><li>◦ The RD auto-configures as Type 1 from the overlay network source IP address and the auto-generated EVI index.</li><li>◦ The RT auto-configures as Type 0 from the 2-byte AS and the 3-byte VNI—Type encoded as 0x0002.</li></ul></li><li>• For a 4-byte autonomous system:<ul style="list-style-type: none"><li>◦ The RD auto-configures as Type 1 from the overlay network source IP address and the auto-generated EVI index.</li><li>◦ The RT auto-configures as Type 2 from the 4-byte AS and the 2-byte EVI—Type encoded as 0x0202.</li></ul></li></ul>

### Example

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
```

**Supported  
releases** 10.4.2.0 or later

## disable-rt-asn

Sets the ASN value to 0 in auto-derived route targets.

<b>Syntax</b>	disable-rt-asn
<b>Parameters</b>	None
<b>Default</b>	Not configured
<b>Command mode</b>	EVPN
<b>Usage information</b>	<p>In a Clos leaf-spine topology, if you configure the leaf nodes (VTEPs) in separate ASNs, the system cannot use the route targets that are automatically generated using the <code>auto-evi</code> or <code>route-target auto</code> commands. The route target includes the ASN and the route targets derived on each of the leaf nodes differ from one another.</p> <p>In such eBGP EVPN scenarios, use the <code>disable-rt-asn</code> command to automatically provision route targets in the leaf nodes. When you use this command, the <code>export route-target</code> has the ASN value</p>

set to 0 and ensures that identical route targets are generated on all the leaf nodes. The leaf VTEPs can import EVPN routes only based on VNI, even though the leaf VTEPs are on different ASNs.

This command is applicable when you use the `auto-evi` or `route-target auto` commands for EVIs, symmetric IRB VRFs, or both.

Note: You must manually configure the route target and set the ASN value to 0 in other vendor switches that do not support the `disable-rt-asn` feature.

### Example 1

```
OS10(config)# evpn
OS10(config-evpn)# auto-evi
OS10(config-evpn)# disable-rt-asn
```

### Example 2

```
OS10(config)# evpn
OS10(config-evpn)# disable-rt-asn
OS10(config-evpn)# evi 1001
OS10(config-evpn-evi-1001)# route-target auto
OS10(config-evpn)# vrf BLUE
OS10(config-evpn-vrf-BLUE)# vni 64001
OS10(config-evpn-vrf-BLUE)# route-target auto
OS10(config-evpn-vrf-BLUE)#
```

**Supported releases** 10.5.1.0 or later

## evi

Creates an EVPN instance (EVI) in EVPN mode.

**Syntax** `evi id`

**Parameters** `id` Enter the EVPN instance ID, from 1 to 65535.

**Default** Not configured

**Command mode** EVPN

**Usage information** If an MP-BGP network uses 4-byte autonomous systems or to specify the RD and RT values, manually configure EVPN instances and associate each EVI with the overlay VXLAN virtual network. The EVI activates only when you configure the VXLAN network ID (VNI), RD, RT, and virtual network.

### Example

```
OS10(config)# evpn
OS10(config-evpn)# evi 10
OS10(config-evpn-evi)#
```

**Supported releases** 10.4.2.0 or later

## evpn

Enables the EVPN control plane for VXLAN.

**Syntax** `evpn`

**Parameters** None

**Default** Not configured

**Command mode** CONFIGURATION

**Usage information** Enabling EVPN triggers BGP to advertise EVPN capability with AFI=25 and SAFI=70 to all BGP peers in an autonomous system. The `no` version of this command disables EVPN on the switch.

## Example

```
OS10(config)# evpn
OS10(config-evpn)#
```

## Supported releases

10.4.2.0 or later

## rd

Configures the Route Distinguisher (RD) value that EVPN routes use.

### Syntax

```
rd {A.B.C.D:[1-65535] | auto}
```

### Parameters

**A.B.C.D:** Manually configure the RD with a 4-octet IPv4 address, then a 2-octet-number [1-65535] from 1 to 65535.

**auto** Configure the RD to automatically generate.

### Default

Not configured

### Command mode

EVPN-EVI and EVPN-VRF

### Usage information

A RD maintains the uniqueness of an EVPN route between different EVPN instances. Configure a route distinguisher in a tenant VRF used for EVPN symmetric IRB traffic. The RD auto-configures as Type 1 from the overlay network source IP address and the auto-generated EVPN instance ID.

The `rd auto` command is not supported in EVPN-VRF mode. When you create a VRF in EVPN mode, the RD is automatically generated. The `rd A.B.C.D:[1-65535]` command is supported in EVPN-VRF mode in 10.5.1 and later releases.

## Example

```
OS10(config)# evpn
OS10(config-evpn)# evi 10
OS10(config-evpn-evi)# vni 10000
OS10(config-evpn-evi)# rd 111.111.111.111:65535
```

```
OS10(config)# evpn
OS10(config-evpn)# vrf vrf-blue
OS10(config-evpn-vrf-vrf-blue)# rd 111.111.111.111:65000
```

## Supported releases

10.4.2.0 or later

## redistribute l2vpn evpn

Redistributes L2VPN EVPN routes into BGP and OSPF IPv4/IPv6 routes.

### Syntax

```
redistribute l2vpn evpn [route-map map name]
```

### Parameters

- `route-map map-name` — (Optional) Filter the L2VPN EVPN routes that are redistributed in BGP and OSPF.

### Default

None

### Command Mode

ROUTER-BGPv4-AF, ROUTER-BGPv6-AF, ROUTER-OSPF, or ROUTER-OSPFv6

### Usage Information

Use the `redistribute l2vpn evpn` command to redistribute the L2VPN EVPN routes learned in non-default tenant VRFs for BGP and or OSPF IPv4/IPv6 routing.

## Example

```
OS10(config)# router bgp 101
OS10(conf-router-bgp-101)# vrf blue
```

```
OS10(config-router-bgp-101-vrf)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute l2vpn evpn
```

```
OS10(config)# router ospf 1 vrf GREEN
OS10(config-router-ospf-1)# redistribute l2vpn evpn
```

```
OS10(config)# router ospfv3 2 vrf GREEN
OS10(config-router-ospfv3-2)# redistribute l2vpn evpn
```

**Supported Releases** 10.5.1 or later

## route-target

Configures the Route Target (RT) values that EVPN routes use.

**Syntax** `route-target {auto | value {import | export | both} [asn4]}`

**Parameters**

<b>value {import   export   both}</b>	Configure an RT import or export value, or both values in the format <i>2-octet-ASN:4-octet-number</i> or <i>4-octet-ASN:2-octet-number</i> . <ul style="list-style-type: none"><li>The <i>2-octet</i> ASN or number is 1 to 65535.</li><li>The <i>4-octet</i> ASN or number is 1 to 4294967295.</li></ul>
<b>auto</b>	Configure the RT import and export values to automatically generate.
<b>asn4</b>	(Optional) Advertises a 4-byte AS number in RT values.

**Default** Not configured

**Command mode** EVPN-EVI and EVPN-VRF

**Usage information** A RT determines how EVPN routes distribute among EVPN instances. Configure each RT with an import and export value. When the EVPN routes advertise, the RT export value configured for export attaches to each route. The receiving VTEP compares a route export value with the local RT import value. If the values match, the routes download and install on the VTEP.

- For 2-byte autonomous systems, the RT auto-configures as Type 0 from the 2-byte AS and the 3-byte VNI—Type encoded as 0x0002.
- For 4-byte autonomous systems, the RT auto-configures as Type 2 from the 4-byte AS and the 2-byte EVI—Type encoded as 0x0202.

Configure a route target in a tenant VRF used for EVPN symmetric IRB traffic. The `route-target` command is supported in EVPN-VRF mode in 10.5.1 and later releases. In EVPN-VRF command mode, the manual route-target configuration should be unique across VRFs.

### Example

```
OS10(config)# evpn
OS10(config-evpn)# evi 10
OS10(config-evpn-evi)# vni 10000
OS10(config-evpn-evi)# rd 111.111.111.111:65535
OS10(config-evpn-evi)# route-target 1:3 both
```

```
OS10(config)# evpn
OS10(config-evpn)# vrf vrf-blue
OS10(config-evpn-vrf-vrf-blue)# route-target auto
```

**Supported releases** 10.4.2.0 or later

## router-mac

Configure the local router MAC address that is used by remote VTEPs as the destination address in VXLAN encapsulated packets sent to the switch.

<b>Syntax</b>	<code>router-mac mac-address</code>
<b>Parameters</b>	<b>mac-address</b> Enter the MAC address in <code>nn:nn:nn:nn:nn:nn</code> format.
<b>Default</b>	Not configured
<b>Command mode</b>	EVPN
<b>Usage information</b>	The EVPN router MAC address is encoded in the router's MAC extended community in MAC/IP Type 2 and IP-prefix Type 5 route updates. It also serves as the destination MAC address in VXLAN encapsulated packets sent to the switch. In a VLT domain, configure the same router MAC address on both VLT VTEP peers.
<b>Example</b>	<pre>OS10(config-evpn)# router-mac 00:01:02:03:04:05</pre>
<b>Supported releases</b>	10.5.1 or later

## show evpn evi

Displays the configuration settings of EVPN instances.

<b>Syntax</b>	<code>show evpn evi [id]</code>
<b>Parameters</b>	<b>id</b> — (Optional) Enter the EVPN instance ID, from 1 to 65535.
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command to verify EVPN instance status, associated VXLAN virtual networks and the RD and RT values the BGP EVPN routes use in the EVI. The status of integrated routing and bridging (IRB) and the VRF used for EVPN traffic also display.
<b>Example</b>	<pre>OS10# show evpn evi 101 EVI : 101, State : up Bridge-Domain      : Virtual-Network 101, VNI 101 Route-Distinguisher : 1:95.0.0.4:101(auto) Route-Targets      : 0:101:268435556(auto) both Inclusive Multicast : 95.0.0.3 IRB                 : Enabled(VRF: default)</pre>
<b>Supported releases</b>	10.4.2.0 or later

## show evpn mac

Displays BGP EVPN routes for host MAC addresses.

<b>Syntax</b>	<code>show evpn mac {count   mac-address nn.nn.nn.nn   evi id [mac-address nn.nn.nn.nn   count   next-hop ip-address count]}</code>
<b>Parameters</b>	<ul style="list-style-type: none"><li><b>count</b> — Displays the total number of local and remote host MAC addresses in EVPN instances.</li><li><b>mac-address nn.nn.nn.nn</b> — Displays the BGP EVPN routes for a specific 48-bit host MAC address.</li></ul>

- `evi id` — Displays the host MAC addresses and next hops in a specified EVPN instance, from 1 to 65535. To filter the output, display information on the host MAC address count for an EVPN ID or for a next-hop IP address, and BGP routes for a specified MAC address.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to display the BGP routes for host MAC addresses in EVPN instances. The type 2 routes received from the remote VTEP is displayed only if there is a corresponding EVI configured locally.

**Examples**

```
OS10# show evpn mac
Type  -(lcl): Local  (rmt): remote

EVI  Mac-Address      Type  Seq-No  Interface/Next-Hop
50   00:00:00:aa:aa:aa  rmt   0       55.1.1.3
```

```
OS10# show evpn mac count

Total MAC Entries :
  Local MAC Address Count :    2
  Remote MAC Address Count :    5
```

```
OS10# show evpn mac evi 811 count

EVI 811 MAC Entries :
  Local MAC Address Count :    1
  Remote MAC Address Count :    2
```

```
OS10# show evpn mac evi 811 next-hop 80.80.1.8 count

EVI 811 next-hop 80.80.1.8 MAC Entries :
  Remote MAC Address Count :    2
```

**Supported releases** 10.4.2.0 or later

## show evpn mac-ip

Displays the BGP EVPN Type 2 routes used for host MAC-IP address binding.

**Syntax** `show evpn mac-ip [count | evi evi [mac-address mac-address] | mac-address mac-address | next-hop ip-address]`

- Parameters**
- `count` — Displays the total number of MAC addresses in EVPN MAC-IP address binding.
  - `evi evi` — Enter an EVPN instance ID, from 1 to 65535.
  - `host ip-address` — Enter the IP address of a host that communicates through EVPN routes.
  - `mac-address mac-address` — Enter the MAC address of a host that communicates through EVPN routes in the format `nn:nn:nn:nn:nn`.
  - `next-hop ip-address` — Enter the IP address of a next-hop switch.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to view the MAC-IP address binding for host communication in VXLAN tenant segments. The type 2 routes received from the remote VTEP displays only if there is a corresponding EVI configured locally.

Use this command to display the snooped MAC-IP binding (ARP entries) for Layer 2 VXLAN bridges. The functionality is extended to Layer 2 VXLAN bridges. Additionally, this command displays Layer 2 VXLAN related information also. Use this command to view snooped MAC-IP bindings of Layer 3 VXLAN bridges.

All existing filters of this command are supported including VRF. The `show ip arp summary` command is supported for Layer 2 VXLAN.

### Example

```
OS10# show evpn mac-ip

Type  -(lcl): Local  (rmt): remote

EVI    Mac-Address      Type  Seq-No  Host-IP      Interface/Next-Hop
101    14:18:77:0c:e5:a3  rmt   0       11.11.11.3   95.0.0.5
101    14:18:77:0c:e5:a3  rmt   0       2001:11::11:3 95.0.0.5
101    14:18:77:25:4e:84  rmt   0       55.55.55.1   95.0.0.3
101    14:18:77:25:6f:84  lcl   0       11.11.11.2
101    14:18:77:25:6f:84  lcl   0       2001:11::11:2
102    14:18:77:0c:e5:a4  rmt   0       12.12.12.3   95.0.0.5
102    14:18:77:0c:e5:a4  rmt   0       2001:12::12:3 95.0.0.5
102    14:18:77:25:4d:b9  rmt   0       12.12.12.1   95.0.0.3
102    14:18:77:25:6e:b9  lcl   0       12.12.12.2
103    14:18:77:25:4e:84  rmt   0       13.13.13.1   95.0.0.3
103    14:18:77:25:4e:84  rmt   0       2001:13::13:1 95.0.0.3
103    14:18:77:25:6f:84  lcl   0       13.13.13.2
103    14:18:77:25:6f:84  lcl   0       2001:13::13:2
104    14:18:77:25:4d:b9  rmt   0       14.14.14.1   95.0.0.3
104    14:18:77:25:4d:b9  rmt   0       2001:14::14:1 95.0.0.3
104    14:18:77:25:6e:b9  lcl   0       14.14.14.2
104    14:18:77:25:6e:b9  lcl   0       2001:14::14:2
105    14:18:77:25:4d:b9  rmt   0       15.15.15.1   95.0.0.3
105    14:18:77:25:4d:b9  rmt   0       2001:15::15:1 95.0.0.3
105    14:18:77:25:6e:b9  lcl   0       15.15.15.2
105    14:18:77:25:6e:b9  lcl   0       2001:15::15:2
106    14:18:77:25:4e:84  rmt   0       16.16.16.1   95.0.0.3
106    14:18:77:25:4e:84  rmt   0       2001:16::16:1 95.0.0.3
106    14:18:77:25:6f:84  lcl   0       16.16.16.2
106    14:18:77:25:6f:84  lcl   0       2001:16::16:2
```

```
OS10# show evpn mac-ip evi 104

Type  -(lcl): Local  (rmt): remote

EVI    Mac-Address      Type  Seq-No  Host-IP      Interface/Next-Hop
104    14:18:77:25:4d:b9  rmt   0       14.14.14.1   95.0.0.3
104    14:18:77:25:4d:b9  rmt   0       2001:14::14:1 95.0.0.3
104    14:18:77:25:6e:b9  lcl   0       14.14.14.2
104    14:18:77:25:6e:b9  lcl   0       2001:14::14:2
```

```
OS10# show evpn mac-ip evi 101 mac-address 14:18:77:0c:e5:a3

Type  -(lcl): Local  (rmt): remote

EVI    Mac-Address      Type  Seq-No  Host-IP      Interface/Next-Hop
101    14:18:77:0c:e5:a3  rmt   0       11.11.11.3   95.0.0.5
101    14:18:77:0c:e5:a3  rmt   0       2001:11::11:3 95.0.0.5
```

```
OS10# show evpn mac-ip mac-address 14:18:77:25:4e:84

Type  -(lcl): Local  (rmt): remote

EVI    Mac-Address      Type  Seq-No  Host-IP      Interface/Next-Hop
101    14:18:77:25:4e:84  rmt   0       55.55.55.1   95.0.0.3
103    14:18:77:25:4e:84  rmt   0       13.13.13.1   95.0.0.3
103    14:18:77:25:4e:84  rmt   0       2001:13::13:1 95.0.0.3
106    14:18:77:25:4e:84  rmt   0       16.16.16.1   95.0.0.3
106    14:18:77:25:4e:84  rmt   0       2001:16::16:1 95.0.0.3
```

### Example (ARP-suppression)

```
OS10# show evpn mac-ip

Type  -(lcl): Local  (rmt): remote
```

```
EVI Mac-Address Type Seq-No Host-IP Interface/Next-Hop
100 00:00:e7:dd:21:2c lcl 0 1.1.1.1 virtual-network100
100 00:00:e7:dd:3b:a9 lcl 0 1.1.1.2 virtual-network100
```

**Supported releases** 10.4.3.0 or later

## show evpn router-mac remote-vtep

Displays both the local and remote router MAC addresses used in symmetric IRB.

**Syntax** `show evpn router-mac {router-vtep [vtep-ip-address]}`

**Parameters** `vtep-ip-address` — (Optional) Enter the IP address of a remote VTEP.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use the `show evpn router-mac remote-vtep` command to display the router MAC address used on the switch and on specified remote VTEPs. Use the `router-mac` command to create a local router MAC address. The `show evpn router-mac` command displays the local router mac and router mac of all remote VTEPs. The `show evpn router-mac remote-vtep [vtep-ip-address]` command displays router mac of specified remote VTEP.

### Example

```
OS10# show evpn router-mac

Local Router MAC : 14:18:77:25:4e:4d

Remote-VTEP      Router's-MAC
4.4.4.4          14:18:77:25:6f:4d
5.5.5.5          00:00:01:00:a3:b4
```

**Supported releases** 10.5.1.0 or later

## show evpn vrf

Displays the VRF instances used to forward EVPN routes in VXLAN overlay networks.

**Syntax** `show evpn vrf [vrf-name]`

**Parameters** `vrf-name` — (Optional) Enter the name of a non-default tenant VRF instance.

**Default** Not configured

**Command mode** EXEC

**Usage information** Use this command to verify the tenant VRF instances used in EVPN instances to exchange BGP EVPN routes in VXLANs.

### Example

```
show evpn vrf

VXLAN-VNI   EVI   Virtual-Network-Instance  VRF-Name
102         102   102                       blue
103         103   103                       default
104         104   104                       blue
106         106   106                       default
105         105   105                       blue
101         101   101                       default
```

**Supported releases** 10.4.3.0 or later

## show evpn vrf l3-vni

Displays the configuration of the tenant VRF instances used for symmetric IRB.

<b>Syntax</b>	<code>show evpn vrf l3-vni [tenant-vrf-name]</code>
<b>Parameters</b>	<i>tenant-vrf-name</i> — (Optional) Enter the name of a non-default tenant VRF instance.
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use the <code>show evpn vrf l3-vni</code> command to display the configuration settings of each tenant VRF with its unique VXLAN VNI. Use the <code>show evpn vrf</code> command to display the tenant VRF instances used to exchange BGP EVPN routes in VXLANs.

### Example

```
OS10# show evpn vrf l3-vni

VRF : vrf_30, State : up
L3-VNI      : 3030
Route-Distinguisher : 1:80.80.1.1:3030(auto)
Route-Targets   : 0:200:268438486(auto) both
Remote VTEP    : 4.4.4.4

VRF : vrf_40, State : up
L3-VNI      : 4040
Route-Distinguisher : 1:80.80.1.1:4040(auto)
Route-Targets   : 0:200:268439496(auto) both
Remote VTEP    : 4.4.4.4

VRF : vrf_50, State : up
L3-VNI      : 5050
Route-Distinguisher : 1:80.80.1.1:5050(auto)
Route-Targets   : 0:200:268440506(auto) both
Remote VTEP    : 4.4.4.4
```

```
OS10# show evpn vrf
VXLAN-VNI  EVI  Virtual-Network-Instance  VRF-Name
30          30   30                          vrf_30
40          40   40                          vrf_40
```

```
OS10# show evpn vrf l3-vni vrf_30
VRF : vrf_30, State : up
L3-VNI      : 3030
Route-Distinguisher : 1:80.80.1.1:3030(auto)
Route-Targets   : 0:200:268435557(auto) both
Remote VTEP    : 4.4.4.4
```

**Supported releases** 10.5.1.0 or later

## show evpn vxlan-vni

Displays the VXLAN overlay network for EVPN instances.

<b>Syntax</b>	<code>show evpn vxlan-vni [vni]</code>
<b>Parameters</b>	<i>vni</i> — (Optional) Enter the VXLAN virtual-network ID, from 1 to 16,777,215.
<b>Default</b>	Not configured
<b>Command mode</b>	EXEC
<b>Usage information</b>	Use this command to verify the VXLAN virtual network and bridge domain used by an EVPN instance.

## Example

```
OS10# show evpn vxlan-vni

VXLAN-VNI   EVI       Bridge-Domain
100         65447     65447
```

**Supported releases** 10.4.2.0 or later

## vni

Associates an EVPN instance with a VXLAN VNI or configures a VXLAN VNI to use for L3 EVPN symmetric IRB traffic.

**Syntax** `vni vni`

**Parameters** `vni` Enter a VXLAN virtual-network ID, from 1 to 16,777,215.

**Default** Not configured

**Command mode** EVPN-EVI and EVPN-VRF

**Usage information** Use this command:

- In EVPN-EVI mode to configure an EVPN instance with RD and RT values for an overlay VXLAN virtual network.
- In EVPN-VRF mode to configure a unique VXLAN VNI for EVPN symmetric IRB traffic in a tenant VRF.

## Example

```
OS10(config)# evpn
OS10(config-evpn)# evi 10
OS10(config-evpn-evi)# vni 10000
```

```
OS10(config)# evpn
OS10(config-evpn)# vrf vrf-blue
OS10(config-evpn-vrf-vrf-blue)# vni 65536
```

**Supported releases** 10.5.1 or later

## vrf

Creates a non-default VRF instance for EVPN symmetric IRB traffic.

**Syntax** `vrf vrf-name`

**Parameters** `vrf-name` — Enter the name of a non-default tenant VRF; 32 characters maximum.

**Default** Not configured

**Command Mode** EVPN

**Usage Information** Configure a non-default VRF for symmetric IRB for each tenant VRF. The tenant VRF is created using the `ip vrf` command when you enable overlay routing with IRB; see [Enable overlay routing between virtual networks](#).

## Example

```
OS10(config)# evpn
OS10(config-evpn)# vrf vrf-blue
```

**Supported Releases** 10.5.1 or later

## Support resources

The Dell Technologies support site provides a range of documents and tools to assist you with effectively using Dell devices. Through the support site you can obtain technical information regarding Dell products, access software upgrades and patches, download available management software, and manage your open cases. The support site provides integrated, secure access to these services.

To access the support site, go to [www.dell.com/support/](http://www.dell.com/support/). To display information in your language, scroll down to the bottom of the page and select your country from the drop-down menu.

- To obtain product-specific information, enter the 7-character service tag or 11-digit express service code of your switch and click **Submit**.

To view the service tag or express service code, pull out the luggage tag on the chassis or enter the `show chassis` command from the CLI.

- To receive additional kinds of technical support, click **Contact Us**, then click **Technical Support**.

To access system documentation, see [www.dell.com/manuals/](http://www.dell.com/manuals/).

To search for drivers and downloads, see [www.dell.com/drivers/](http://www.dell.com/drivers/).

To participate in Dell Technologies community blogs and forums, see [www.dell.com/community](http://www.dell.com/community).

# Index

## B

bgp unnumbered [169](#)