

VXLAN and BGP EVPN Configuration Guide for OS10 Enterprise Edition

Release 10.4.3.0

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.

© 2019 Dell Inc. or its subsidiaries. All rights reserved. Dell, EMC, and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners.

Contents

1 VXLAN	6
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	14
DHCP relay on VTEPs	15
View VXLAN configuration.....	16
VXLAN MAC addresses.....	18
VXLAN commands.....	20
hardware overlay-routing-profile.....	20
interface virtual-network.....	21
ip virtual-router address.....	22
ip virtual-router mac-address.....	22
member-interface.....	22
nve.....	23
remote-vtep.....	23
show hardware overlay-routing-profile mode.....	24
show interface virtual-network.....	24
show nve remote-vtep.....	25
show nve remote-vtep counters.....	26
show nve vxlan-vni.....	26
show virtual-network.....	26
show virtual-network counters.....	27
show virtual-network interface counters.....	27
show virtual-network interface.....	28
show virtual-network vlan.....	29
show vlan (virtual network).....	29
source-interface loopback.....	29
virtual-network.....	30
virtual-network untagged-vlan.....	30
vxlan-vni.....	31
VXLAN MAC commands.....	31
clear mac address-table dynamic nve remote-vtep.....	31
clear mac address-table dynamic virtual-network.....	32

show mac address-table count extended.....	32
show mac address-table count nve.....	33
show mac address-table count virtual-network.....	34
show mac address-table extended.....	34
show mac address-table nve.....	35
show mac address-table virtual-network.....	36
Example: VXLAN with static VTEP.....	37
VTEP 1 Leaf Switch.....	38
VTEP 2 Leaf Switch.....	40
VTEP 3 Leaf Switch.....	43
VTEP 4 Leaf Switch.....	45
Spine Switch 1.....	48
Spine Switch 2.....	49
2 Controller-provisioned VXLAN.....	50
Configure the NSX controller.....	50
Configure an OVSDB controller.....	50
Assign an interface.....	51
BFD in VXLAN service nodes.....	52
View replicators.....	53
Configure and control VXLAN from VMware vCenter.....	53
Example: VXLAN with a controller configuration.....	56
Configure the controller and interface.....	57
Verify the controller configuration.....	57
VXLAN Controller commands.....	58
controller ovsdb.....	58
ip port ssl.....	59
max-backoff.....	59
nve-controller.....	60
nve controller ssl-key-generate.....	60
show nve controller.....	60
show nve controller ssl-certificate.....	61
show nve replicators.....	62
show ovsdb-tables mac-local-ucast.....	62
show ovsdb-tables mac-remote-ucast.....	63
show ovsdb-tables manager.....	63
show ovsdb-tables tunnel.....	64
3 BGP EVPN for VXLAN.....	65
BGP EVPN compared to static VXLAN.....	65
VXLAN BGP EVPN operation.....	66
Configure BGP EVPN for VXLAN.....	69
VXLAN BGP EVPN routing.....	72
BGP EVPN with VLT.....	73
VXLAN BGP commands.....	74
activate (l2vpn evpn).....	74

address-family l2vpn evpn.....	75
allowas-in.....	75
sender-side-loop-detection.....	76
show ip bgp l2vpn evpn.....	76
VXLAN EVPN commands.....	77
auto-evi.....	77
evi.....	78
evpn.....	78
rd.....	79
route-target.....	79
show evpn evi.....	80
show evpn mac.....	80
show evpn mac-ip.....	81
show evpn vrf.....	82
show evpn vxlan-vni.....	83
vni.....	83
Example: VXLAN with BGP EVPN.....	84
VTEP 1 Leaf Switch.....	85
VTEP 2 Leaf Switch.....	89
VTEP 3 Leaf Switch.....	93
VTEP 4 Leaf Switch.....	96
Spine Switch 1.....	100
Spine Switch 2.....	102
4 Support resources.....	105

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 multi-tenant 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.

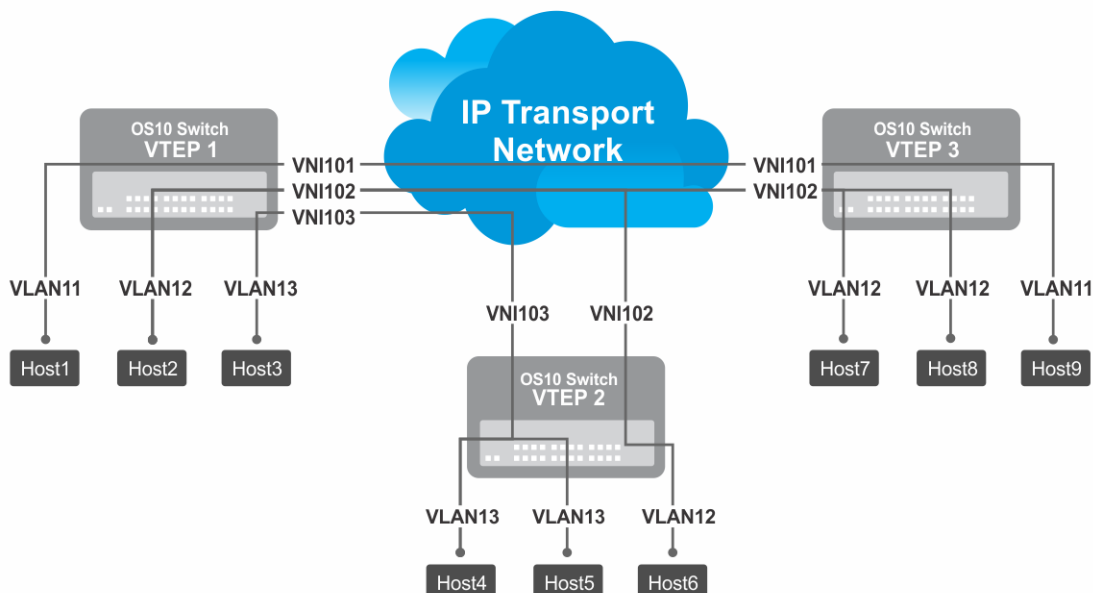


Figure 1. VXLAN topology

Topics:

- [VXLAN concepts](#)
- [VXLAN as NVO solution](#)
- [Configure VXLAN](#)
- [L3 VXLAN route scaling](#)
- [DHCP relay on VTEPs](#)
- [View VXLAN configuration](#)
- [VXLAN MAC addresses](#)
- [VXLAN commands](#)
- [VXLAN MAC commands](#)

VXLAN concepts

Network virtualization overlay (NVO)	<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">· The NVO overlay network uses a separate L2 bridge domain (virtual network), which is independent of legacy VLAN forwarding.· The NVO underlay network operates in the default VRF using the existing L3 infrastructure and routing protocols.
Virtual extensible LAN (VXLAN)	<p>A type of network virtualization overlay that encapsulates a tenant payload into IP UDP packets for transport across the IP underlay network.</p>
VXLAN network identifier (VNI)	<p>A 24-bit ID number that identifies a tenant segment and transmits in a VXLAN-encapsulated packet.</p>
VXLAN tunnel endpoint (VTEP)	<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>
Bridge domain	<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">· Simple VLAN: A bridge domain a VLAN ID represents. Traffic on all member ports is assigned with the same VLAN ID.· 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>.
Distributed routing	<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">· Asymmetric routing: All VTEPs can perform routing. Routing decisions are made only on ingress VTEPs. Egress VTEPs perform bridging.· Symmetric routing: All VTEPs perform routing. Routing decisions are made on both ingress and egress VTEPs.
Virtual network	<p>In OS10, each L2 flooding domain in the overlay network is represented as a <i>virtual network</i>.</p>
Virtual network identifier (VNID)	<p>A 16-bit ID number that identifies a virtual network in OS10.</p>
Virtual-network interface	<p>A router interface that connects a virtual network bridge to a tenant VRF routing instance.</p>
Access port	<p>A port on a VTEP switch that connects to an end host and is part of the overlay network.</p>
Network port	<p>A port on a VTEP switch that connects to the underlay network.</p>
Switch-scoped VLAN	<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> <ul style="list-style-type: none">· 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	<p>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:</p> <ul style="list-style-type: none">· 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.

- 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.

```
vxlan-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.

- 4 Return to VIRTUAL-NETWORK mode.

```
exit
```

- 5 Return to CONFIGURATION mode.

```
exit
```

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.

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

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

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

- b 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-5)# 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:

- a Configure interfaces as trunk members in Interface mode.

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

- b 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.

- c Repeat Steps a) and b) to assign additional member Port,VLAN pairs to the virtual network.
 - 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.

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:

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

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

- b 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:

- a 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
```

- b 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
```

- c 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 described how a VTEP switches traffic between hosts within the same L2 tenant segment, the 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 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.

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 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, 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
```

- 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 EMC 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.
- Routing protocols, such as Open Shortest Path First (OSPF) and BGP, are not supported on the virtual-network interface in the overlay network. However, 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.
- When you add a static route in the overlay, any next-hop IP address that belongs to a virtual-network subnet must be the only next-hop for that route and cannot be one of multiple Electronic Commerce Messaging Protocol (ECMP) next-hops. For example, if you enter the following configuration commands one after the other, where 10.250.0.0/16 is a virtual-network subnet, only the first next-hop is active on the switch.

```
OS10(config)# ip route 0.0.0.0/0 10.250.0.101
OS10(config)# ip route 0.0.0.0/0 10.250.0.102
```

If the next-hop is a pair of dual-homed VTEPs in a VLT domain, a workaround is to configure the same anycast gateway IP address on both VTEPs and use this address as the next-hop IP address.

- 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.
- IP routing of incoming VXLAN encapsulated traffic in the overlay after VXLAN termination is not supported.

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

Virtual network	VTEP	Anycast gateway MAC address
	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

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
```

L3 VXLAN route scaling

The S4100-ON series, S5200-ON series, S4048T-ON, 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:	—	—	—	—

OS10 Switch	Overlay next-hop entries	Underlay next-hop entries	Overlay L3 RIF entries	Underlay L3 RIF entries
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# 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 Underlay Next-hop Overlay L3 RIF Underlay L3 RIF
Entries Entries Entries 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 on hosts in the overlay communicate with a DHCP server using a DHCP relay on the VTEP switch. In OS10, DHCP relay is supported on VTEPs only if you locate the DHCP server in the underlay network. To work seamlessly, VTEP DHCP relay transmits the virtual-network IP address of the relay interface to the DHCP server.

By default, DHCP uses the `giaddr` packet field to carry these addresses to the server. In a VxLAN, which has overlay and underlay subnets in the same default VRF, DHCP relay on VTEPs operates without user intervention. However, in a VxLAN in which the underlay

and overlay are in different VRFs, the default DHCP method is not successful. The IP tenant subnet is in the overlay address space. The IP address where the VTEP is reachable is in the underlay address space. To transmit the IP subnet of the client separately from the IP address where the VTEP is reachable, you must configure an additional DHCP sub-option (5 or 151) in DHCP relay agent option 82.

Because OS10 does not support the required sub-options in DHCP relay agent option 82, the `giaddr` packet field must contain the virtual-network IP address of the relay interface, and this IP address must be reachable from the DHCP server in the underlay. Each VTEP that acts as a DHCP relay must have its virtual-network IP address installed using a route leaking mechanism as a route to the underlay and advertised to all underlay routers, including the spine switches.

Similarly, the DHCP server in the underlay VRF must be reachable from the client tenant VRF in the overlay. Configure a static route for the DHCP server subnet in the underlay default VRF, and leak the static route to the client tenant VRF in the overlay. This configuration sets up a bi-directional communication between the client and DHCP server across the virtual networks. The route-leaking configuration is not required if the VxLAN overlay subnet and underlay subnet are in same default VRF.

Configure DHCP relay on VTEPs

- 1 Configure the IP address of the virtual-network relay interface in the non-default tenant VRF as a static route in the default VRF.

```
OS10(config)# ip route 10.10.0.2/32 interface virtual-network 10
```

- 2 Configure a static IP route to the DHCP server interface in the tenant VRF.

```
OS10(config)# ip route vrf tenant01 40.1.1.0/24 interface vlan40
```

- 3 Configure DHCP relay on the virtual-network interface of the tenant VRF.

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

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
```

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)
```

View the VXLAN statistics on the remote VTEPs

```
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
```

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
<code>show mac address-table virtual-network [vn-id local remote static dynamic address mac-address interface {ethernet node/slot/port:subport port-channel number}]</code>	Displays all MAC addresses learned on all or a specified virtual network. <i>vn-id</i> : Displays only information about the specified virtual network. <i>local</i> : Displays only locally-learned MAC addresses. <i>remote</i> : Displays only remote MAC addresses.

Command**Description**

```
show mac address-table extended [address mac-address | interface {ethernet node/slot/port:subport | port-channel number} | static | dynamic]
```

`static`: Displays only static MAC addresses.

`dynamic`: Displays only dynamic MAC addresses.

`address mac-address`: Displays only information about the specified MAC address.

`interface ethernet node/slot/port:subport`: Displays only MAC addresses learned on the specified interface.

`interface port-channel number`: Displays only MAC addresses learned on the specified port channel.

Displays MAC addresses learned on all VLANs and VXLANs (default).

`address mac-address`: Displays only information about the specified MAC address.

`interface ethernet node/slot/port:subport`: Displays only MAC addresses learned on the specified interface.

`interface port-channel number`: Displays only MAC addresses learned on the specified port channel.

`static`: Displays only static MAC addresses.

`dynamic`: Displays only dynamic MAC addresses.

```
show mac address-table nve {vxlan-vni vn-id | remote-vtep ip-address}
```

`vxlan-vni vn-id`: Displays MAC addresses learned on NVE from the specified VXLAN virtual-network ID.

`remote-vtep ip-address`: Displays MAC addresses learned on NVE from the specified remote VTEP.

```
show mac address-table count virtual-network [dynamic | local | remote | static | interface {ethernet node/slot/port:subport | port-channel number} | vn-id]
```

Displays the number of MAC addresses learned on all virtual networks (default).

`dynamic`: Displays the number of dynamic MAC addresses learned on all or a specified virtual network.

`local`: Displays the number of locally-learned MAC addresses.

`remote`: Displays the number of remote MAC addresses learned on all or a specified virtual network.

`static`: Displays the number of static MAC addresses learned on all or a specified virtual network.

`interface ethernet node/slot/port:subport`: Displays the number of MAC addresses learned on the specified interface.

`interface port-channel number`: Displays the number of MAC addresses learned on the specified port channel.

`vn-id`: Displays the number of MAC addresses learned on the specified virtual network.

```
show mac address-table count nve {remote-vtep ip-address | vxlan-vni vn-id}
```

Displays the number of MAC addresses learned for a virtual network or from a remote VTEP.

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

VXLAN commands

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.

Syntax

`interface virtual-network vn-id`

Parameters

**virtual-
network *vn-id*** Enter a virtual-network ID, from 1 to 65535.

Default

Not configured

Command mode

CONFIGURATION

Usage information

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.

Example

```
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
```

Supported releases

10.4.3.0 or later

ip virtual-router address

Configures an anycast gateway IP address for a VXLAN virtual network.

Syntax	<code>ip virtual-router address ip-address</code>
Parameters	address ip-address Enter the IP address of the anycast L3 gateway.
Default	Not configured
Command mode	INTERFACE-VIRTUAL-NETWORK
Usage information	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.
Example	<pre>OS10(config)# interface virtual-network 10000 OS10(config-if-vn-10000)# ip virtual-router address 10.1.0.100</pre>
Supported releases	10.4.3.0 or later

ip virtual-router mac-address

Configures the MAC address of an anycast L3 gateway for VXLAN routing.

Syntax	<code>ip virtual-router mac-address mac-address</code>
Parameters	mac-address mac-address Enter the MAC address of the anycast L3 gateway.
Default	Not configured
Command mode	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	<pre>OS10(config)# ip virtual-router mac-address 00:01:01:01:01:01</pre>
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	<code>member-interface {ethernet node/slot/port[:subport] port-channel number} {vlan-tag vlan-id untagged}</code>
---------------	---

Parameters	<p>ethernet node/ slot/ port[:subport] Assign the specified interface to a virtual network.</p> <p>port-channel number Assign the specified port channel to a virtual network.</p> <p>untagged Assign untagged traffic on an interface or port channel to a virtual network.</p> <p>vlan-tag vlan-id Assign tagged traffic on the specified VLAN to a virtual network.</p>
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 <code>no</code> version of this command removes the configured value.
Example	<pre>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</pre>
Supported releases	10.4.2.0 or later

nve

Enters network virtualization edge (NVE) configuration mode to configure the source VXLAN tunnel endpoint.

Syntax	<code>nve</code>
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	<pre>OS10# nve OS10(config-nve)#</pre>
Supported releases	10.4.2.0 or later

remote-vtep

Configures the IP address of a remote tunnel endpoint in a VXLAN network.

Syntax	<code>remote-vtep ip-address</code>
Parameters	<code>ip-address</code> — 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 Next-hop Underlay Next-hop Overlay Underlay
Entries Entries L3 RIF L3 RIF
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 Next-hop Underlay Overlay Underlay
Entries Entries L3 RIF L3 RIF
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.

Syntax `show interface 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 virtual-network IP address used for routing traffic in a virtual network. Traffic counters also display.

Example

```
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.

Syntax `show nve remote-vtep [ip-address | summary | counters]`

Parameters

ip-address	Display detailed information about a specified remote VTEP.
summary	Display summary information about remote VTEPs.
counters	Display statistics on remote VTEP traffic.

Default Not configured

Command mode EXEC

Usage information 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.

Example

```
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	<i>vn-id</i> 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	<pre>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)</pre>
Supported releases	10.4.2.0 or later

show virtual-network counters

Displays packet statistics for virtual networks.

Syntax	<code>show virtual-network [vn-id] counters</code>
Parameters	<i>vn-id</i> 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 <code>clear virtual-network counters</code> command to clear virtual-network counters.
Example	<pre>OS10# show virtual-network counters Virtual-Network Input (Packets/Bytes) Output (Packets/Bytes) 1000 857/8570 257/23709 2000 457/3570 277/13709</pre>
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	<code>show virtual-network interface {ethernet <i>node/slot/port:subport</i> port-channel <i>number</i>} [vlan <i>vlan-id</i>] counters</code>
Parameters	<i>interface</i> Enter the port information for an Ethernet interface. <i>ethernet node/</i>

```

slot/
port[:subport]

interface          Enter a port-channel number, from 1 to 128.
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 128.
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`

Parameters `vlan vlan-id` Enter a VLAN ID, from 1 to 4093.

Default Not configured

Command mode EXEC

Usage information Use this command to verify the VXLAN virtual networks where a VLAN is assigned, including the port members connected to downstream servers.

Example

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

Supported releases 10.4.2.0 or later

show vlan (virtual network)

Displays the VLANs assigned to virtual networks.

Syntax `show vlan`

Parameters None

Default Not configured

Command mode EXEC

Usage information Use this command to display the VLAN port interfaces that transmit VXLAN packets over a virtual network.

Example

```
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           A  Eth1/1/1-1/1/48
@ 100  up      T           T  Eth1/1/2,Eth1/1/3
           A           A  Eth1/1/1
@ 101  up      T           T  port-channel5
  200  up      T           T  Eth1/1/11-1/1/15
```

Supported releases 10.4.2.0 or later

source-interface loopback

Configures a dedicated Loopback interface as the source VTEP.

Syntax `source-interface loopback number`

Parameters	loopback number	Enter the Loopback interface used as the source interface of a VXLAN virtual tunnel, from 0 to 16383.
Default	Not configured	
Command mode	NVE-INSTANCE	
Usage information	<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"> • The Loopback interface must have an IP address configured. The Loopback IP address must be reachable from the remote VTEP. • You cannot change the source interface if at least one VXLAN virtual network ID (VNID) is configured for the NVE instance. <p>Use this command in NVE mode to override a previously configured value and reconfigure the source IP address. The <code>no</code> version of this command removes the configured value.</p>	
Examples	<pre>OS10(config-nve)# source-interface loopback 1</pre>	
Supported releases	10.4.2.0 or later	

virtual-network

Creates a virtual network for VXLAN tunneling.

Syntax	<code>virtual-network vn-id</code>	
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 <code>vxlan-vni</code> command. The <code>no</code> version of this command removes the configured virtual network.	
Example	<pre>OS10(config)# virtual-network 1000 OS10(config-vn)#</pre>	
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	<code>virtual-network untagged-vlan vlan-id</code>	
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 <code>no</code> version of this command removes the configured value.
Example	<pre>OS10(config)# virtual-network untagged-vlan 10</pre>
Supported releases	10.4.2.0 or later

vxlan-vni

Assigns a VXLAN ID to a virtual network.

Syntax	<code>vxlan-vni vni</code>
Parameters	vni Enter the VXLAN ID for a virtual network, from 1 to 16,777,215.
Default	Not configured
Command mode	VIRTUAL-NETWORK
Usage information	This command associates a VXLAN ID number with a virtual network. The <code>no</code> version of this command removes the configured ID.
Example	<pre>OS10(conf-vn-100)# vxlan-vni 100 OS10(config-vn-vxlan-vni)#</pre>
Supported releases	10.4.2.0 or later

VXLAN MAC commands

clear mac address-table dynamic nve remote-vtep

Clears all MAC addresses learned from a remote VTEP.

Syntax	<code>clear mac address-table dynamic nve remote-vtep ip-address</code>
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 <code>show mac address-table nve remote-vtep</code> command. Use this command to delete all MAC address entries learned from a remote VTEP.
Example	<pre>OS10# clear mac address-table dynamic nve remote-vtep 32.1.1.1</pre>
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 ethernet node/slot/ slot/ port[:subport]	Clear all MAC addresses learned on the specified interface.
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 <i>EEEE.EEEE.EEEE</i> format.

Default Not configured

Command mode EXEC

Usage information 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 `show mac address-table virtual-network` command to display the MAC addresses learned on a virtual network.

Example `OS10# clear mac address-table dynamic virtual-network`

Supported releases 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.

Syntax `show mac address-table count extended [interface {ethernet node/slot/port:subport | port-channel number}]`

Parameters

interface ethernet node/slot/ slot/ port[:subport]	Display the number of MAC addresses learned on all VLANs and VXLANs on the specified interface.
---	---

interface Display the number of MAC addresses learned on all VLANs and VXLANs on the specified
port-channel port channel.
number

Default Not configured

Command mode EXEC

Usage information Use this command to display the number of MAC address entries learned on all VLANs and VXLAN virtual networks.

Example

```
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
```

Supported releases 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.

Syntax show mac address-table count 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` 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 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

dynamic	Display the number of local dynamically-learned MAC addresses.
local	Display the number of local MAC addresses.
remote	Display the number of MAC addresses learned from remote VTEPs.
static	Display the number of local statically-configured MAC addresses.
interface ethernet node/ slot/ port[:subport]	Display the number of MAC addresses learned on the specified interface.
interface port-channel number	Display the number of MAC addresses learned on the specified port channel.
vn-id	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 Display only MAC addresses learned on the specified port channel.

port-channel
number

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           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           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           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` 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

- vn-id** Display only information about the specified virtual network.
- local** Display only locally learned MAC addresses.
- remote** Display only remote MAC addresses.
- static** Display only static MAC addresses.
- dynamic** Display only dynamic MAC addresses.
- address *mac-address*** Display only information about the specified MAC address. Enter the MAC address in *EEEE.EEEE.EEEE* format.
- 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.

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           9999  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           100    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           300    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

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.

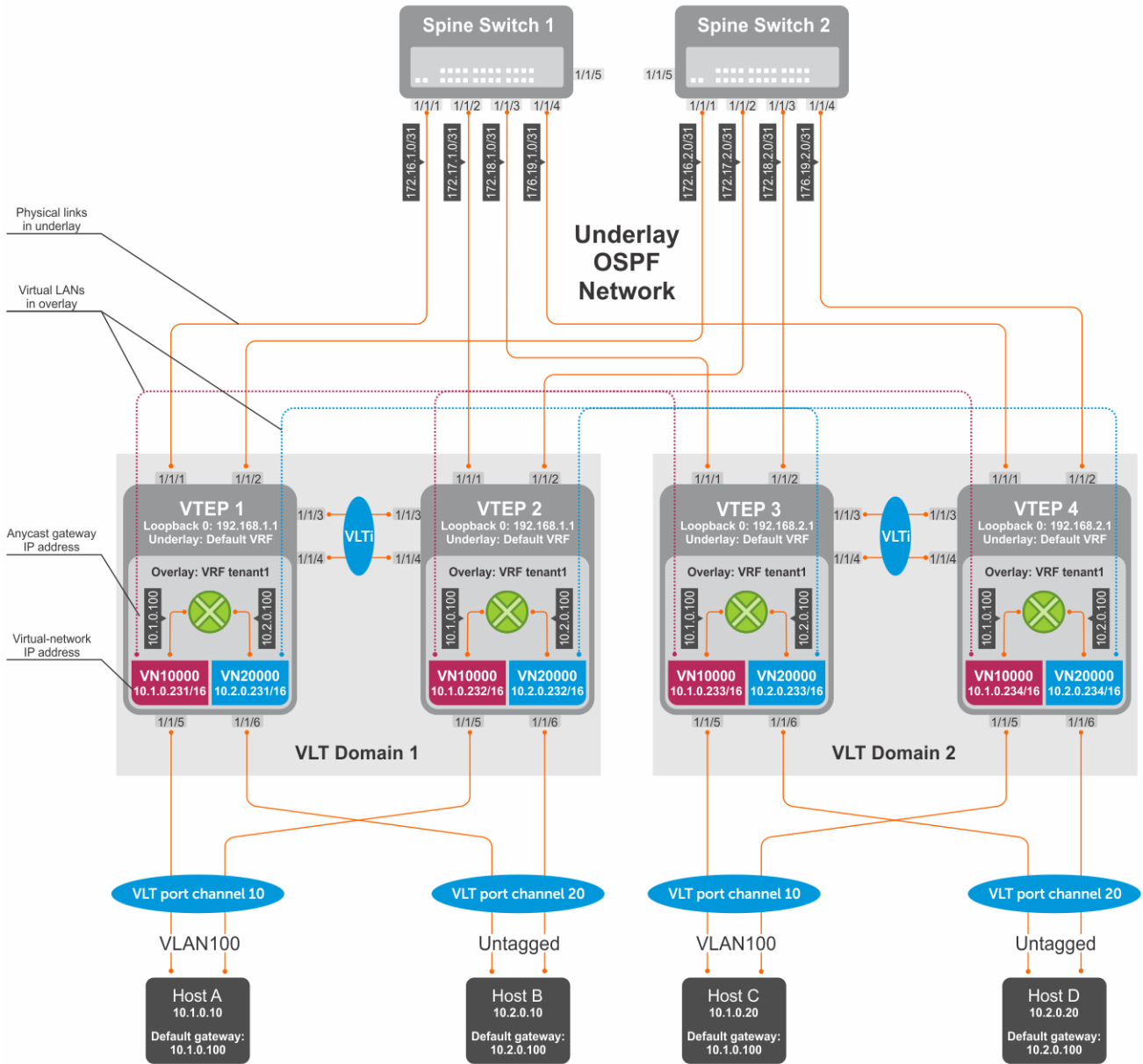


Figure 2. Static VXLAN use case

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

```
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)# upstream port-channel20
OS10(conf-uplink-state-group-1)# exit
```

9. Configure overlay IP routing

Create the tenant VRF

```
OS10(config)# ip vrf tenant1
OS10(conf-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(conf-if-po-10)# switchport access vlan 200
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 access
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 172.17.1.0/31
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(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/2)# ip address 172.17.2.0/31
OS10(conf-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(conf-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(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
```

9. Configure overlay IP routing

Create a tenant VRF

```
OS10(config)# ip vrf tenant1
OS10(config-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(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 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(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 172.18.1.0/31
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 172.18.2.0/31
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(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 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(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 172.19.1.0/31
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(conf-if-eth1/1/2)# no switchport
OS10(conf-if-eth1/1/1)# mtu 1650
OS10(conf-if-eth1/1/2)# ip address 172.19.2.0/31
OS10(conf-if-eth1/1/2)# ip ospf 1 area 0.0.0.0
OS10(conf-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(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 a 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:dd:cc: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
```

10. Configure overlay IP routing

Create a tenant VRF

```
OS10(config)# ip vrf tenant1
OS10(conf-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(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)# 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(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)# ip ospf 1 area 0.0.0.0
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)# ip ospf 1 area 0.0.0.0
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(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 controller-based virtual extensible LAN (VXLAN) implementation using a VMware NSX controller, which handles configurations and control plane operations. Controller-based VXLAN deployments are solutions for Layer 2 (L2) extensions within and between data centers. These deployments are sometimes a requirement for cross-data center (DC) connectivity for disaster recovery, storage replication, and general LANs that stretch over long distances for virtual appliances and workload mobility.

Unlike controller-based VXLAN implementations, in a static VXLAN, an OS10 switch operates as a VXLAN tunnel endpoint (VTEP) and also acts as a VXLAN gateway. The switch performs encapsulation or decapsulation of L2 frames that are sent to the underlay network; a Layer 3 (L3) network. The VTEP classifies the frames based on manually configured parameters. For more information about static VXLAN configuration, see [Example: VXLAN with static VTEP](#).

NOTE: Controller-based VXLAN is not supported on the S5148F-ON and S3048-ON switches.

Topics:

- [Configure the NSX controller](#)
- [Configure and control VXLAN from VMware vCenter](#)
- [Example: VXLAN with a controller configuration](#)
- [VXLAN Controller commands](#)

Configure the NSX controller

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

- 1 Create a Loopback interface and assign an IPv4 address.
- 2 Create a NVE instance and configure the NVE source interface.
- 3 Enable an NSX controller.
- 4 Assign interfaces to expose to the controller.
- 5 (Optional) Implement Bidirectional Forwarding Detection (BFD) for NSX in the VXLAN network.

For more information about creating a Loopback interface and configuring an NVE source interface, see the *Configure source IP address on VTEP* section in the *OS10 Enterprise Edition User Guide*.

Configure an OVSDB controller

In OS10 VTEP, the controller configuration command initializes a connection to an OVSDB-based controller. The OVSDB controller command also starts the OVSDB server service in the specific OS10 VTEP. OS10 supports only one controller connection at a time.

NOTE: The only supported OVSDB controller is NSX.

To configure an OVSDB controller on the OS10 VTEP:

- 1 Enable VXLAN in CONFIGURATION mode.

```
OS10(config)# nve
```
- 2 Enable the controller in CONFIGURATION-NVE mode.

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

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

The range is from 0 to 65535. Configure the port-number as 6640 and the connection type as SSL.

- (Optional) Specify the time interval, in milliseconds (ms), that 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 an interface

For the VXLAN setup, explicitly assign interfaces for an OVSDb controller to manage.

Before you assign the interface, consider the following:

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

If you do not meet these conditions when assigning the interface for controller mode, the system returns error messages.

When the interface is in controller mode, you cannot:

- remove the interface from Switchport Trunk mode
- tag or untag the interface to any VLAN
- remove the interface from the controller configuration if the interface has active port-scoped VLAN (Port,VLAN) pairs the controller configured

To assign an interface to controller mode:

- 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)

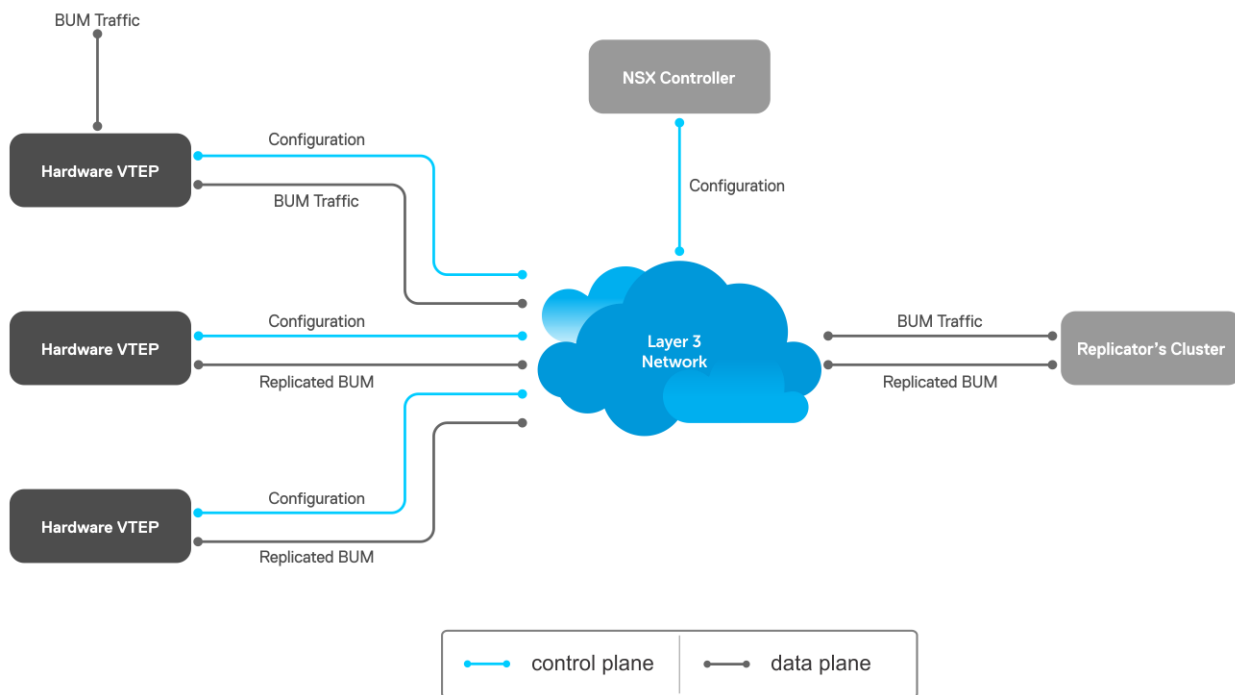
Controller Cluster
IP      Port    Protocol  Connected  State      Max-Backoff
10.16.140.173 6640    ssl      true       ACTIVE     1000
10.16.140.171 6640    ssl      false      BACKOFF    1000
10.16.140.172 6640    ssl      true       ACTIVE     1000
```

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

BFD in VXLAN service nodes

Service nodes (SN) handle L2 broadcast, unknown-unicast, and multicast (BUM) traffic that originates in a controller-based VXLAN environment. In SN replication, the NSX suite deploys hypervisors that replicate BUM traffic across VTEPs. These hypervisors are called *service node replicators*, which are deployed as a cluster. The NSX controller notifies the OVSDB server in the OS10 VTEP about the replication cluster through the OVSBX tables. A service node is also a VTEP, but the controller fully manages the service node. The SN is the central replication engine for flooded packets. In VXLAN, the BUM traffic replicates to all the other remote VTEPs in the same VXLAN.

The following shows BUM traffic replication in the controller environment:



A source VTEP forwards BUM traffic to the replicator, which in turn replicates the frames and transmits them to all other VTEPs through the underlay IP route.

Use BFD to detect the health of the service nodes. BFD runs between the VTEP and the service nodes. The NSX controller provides parameters, such as the minimum TX and RX interval, and the multiplier, to initiate the BFD session. To establish a BFD session, enable the BFD on both the controller and the VTEP.

NOTE: When establishing a BFD session, use the controller-provided parameters instead of the VTEP-configured parameters.

NOTE: If you do not enable BFD, the controller does not trigger BFD session updates and the choice of the service node does not depend on the BFD state.

If you disable BFD in the VTEP and enable BFD at the NSX controller, or if you disable BFD in the VTEP after being enabled, the service node is selected based on next-hop reachability. After you enable BFD, a new service node with BFD enabled is selected from the list.

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.

- Example without a VNID

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

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

- Example with a 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
```

Configure and control VXLAN from VMware vCenter

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

- 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-----
MIIDgDCCAmgCAQMwDQYJKoZIhvcNAQENBQAwgYExCzAJBgNVBAYTA1VTMQswCQYD
VQQIDAJDQTEVMBMGAlUECgMT3B1biB2U3dpdGNoMREwDwYDVQQLEAhzd210Y2hj
YTE7MDkGA1UEAwYtT1ZTIHN3aXRjaGNIENBIENlcnRpZmljYXRlICgyMDE4IFNl
cCAyMyAwMzozMDZyMzIwMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYy
iTELMakGA1UEBhMCVVMwCzAJBgNVBAGMAkNBMRUwEwYDVQQKDAxPcGVuIHZTd210
Y2gxHzAdBgNVBAsMFk9wZW4gdln3aXRjaCBjaXZJ0aWZpZXIiXNTAaZBgNVBAMMLGRl
bGwgaWQ6MGVlZmUyYmVtNGJjOC00MmVmLTkzOTEtN2RlMmMwY2JmMTJjMIIBIjAN
BgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsMlD4c4fWwY+5t6VScjizlkFsNzE
BOK5PJyI3B6ReRK/J14Fdxio1YmzG0YObjxiwjpUYEsqPL3Nvh0f10KMqwgJVbdf
6sXWHUVw+9A7cIfRh0aRI+HIYyUC4YD48GlnVnaCqhxYaA0tcMzJm4r2k7AjwJU1
0pDXiqS3uJwGmfxlhvmFio8EeHM/Z79DkBRD6FUMwacAnb3yCIKZH50AWq7qRmmG
NZOgYUT+8oaj5t0/hEQFDYuv32E5z4d3FhiBJMFT86T4YvpJYyJkiKmaQWInkthL
V3VxEMXi5vJQc1MhwYbKfPB4hh3+qdS5o+uVco76CvrcWi7r03XmsBkbnQIDAQAB
MA0GCSqGSIb3DQEBDQUAA4IBAQAuFVD20GcHD8zdpYf0YaP4b6TuonUzF0jwoV+
Qr9b4k0jEBGuoPdevX3AeV/dvAa2Q6o1iOBM5z74NgHizhr067pFP841Nv7DAVb7
cPHHSSTSeeJjIVMh0kv0KkVefsYuI4r1jqJxu0GZgBinqehXxVKlceouLvwbbhl
MFYXN31cE2AXR746q1VIC6stNkxf3nr1OpSDz3P4VOnbAnIrY+SvUVmAT0tdrowH
99y2AzoAxUHOdWsh8EjFch7VilmCVVhyghXdfyl6lv/F6vMRwjc343BpBW3QsGj
68ROX0ILrtOz/2q5oUb/rpJd15KFFN3itT/xYBfZ1ZdLYd5F
-----END CERTIFICATE-----
```

- Create a VXLAN gateway.

Open a browser window, enter the vCenter IP address, and log in to VMware vCenter. To create a service node, provide the IP address and SSL certificate. Select **Home > Networking and Security > Service Definition > Hardware Devices**. Under the **Hardware Devices** tab, click the **Add (+)** button. The **Add Hardware Device** dialog window opens. Enter a name and copy the certificate generated in the OS10 switch and paste it in the **Certificate** box and click **OK**. This step configures the L2 gateway switch for the VXLAN virtual network.

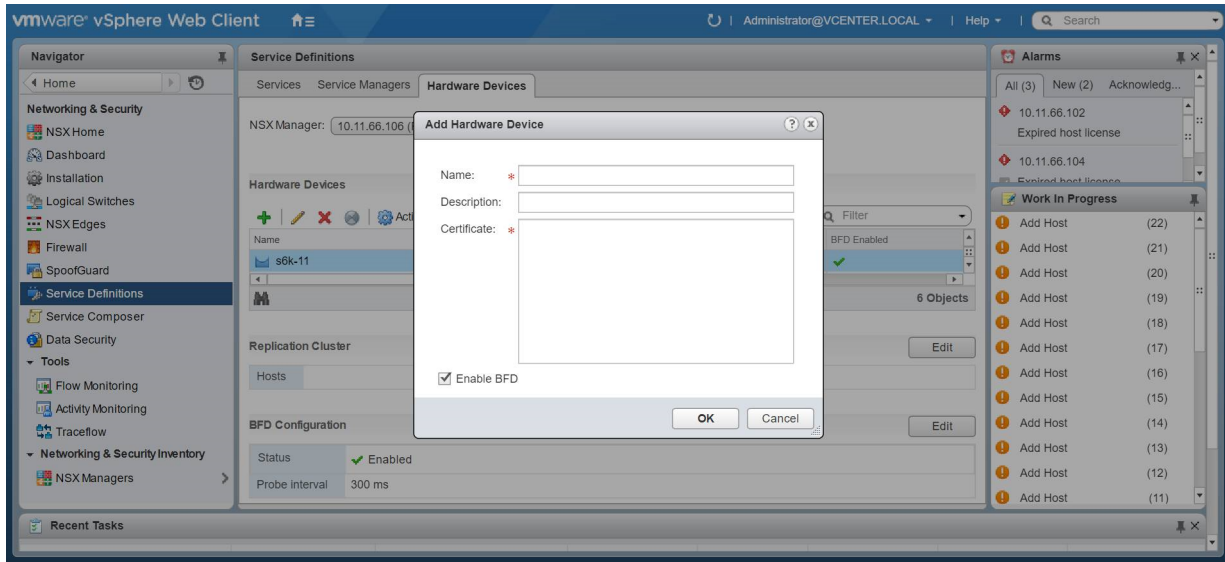
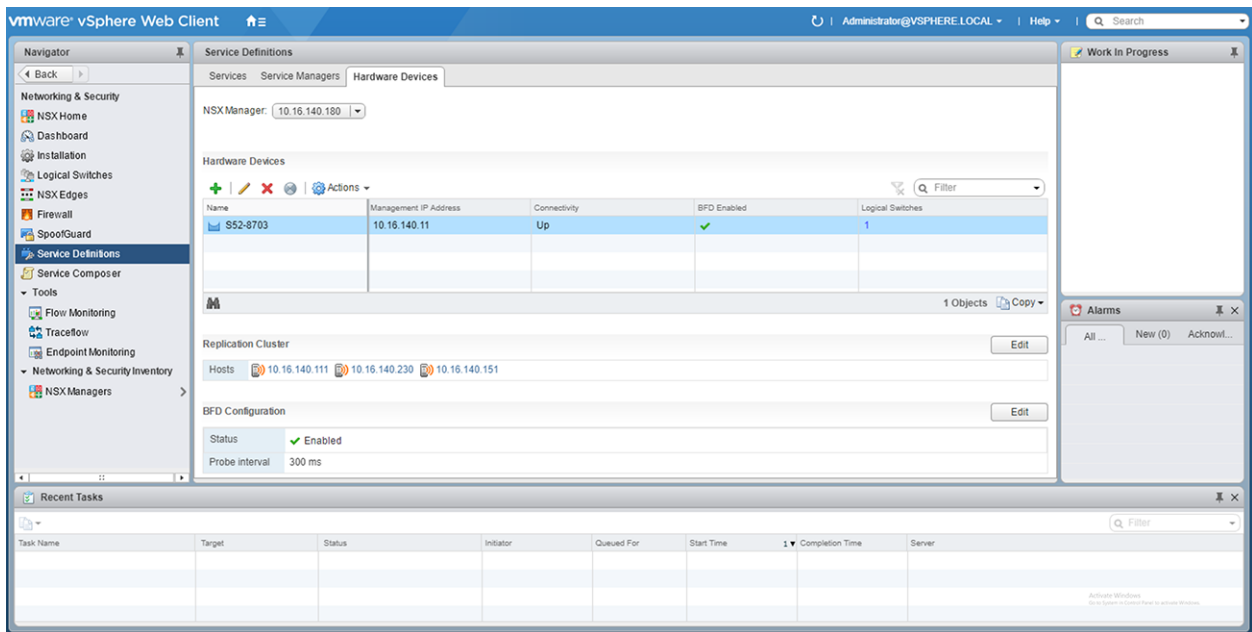


Figure 3. Add hardware device

To create a VXLAN L2 Gateway, you need the IP address of the Gateway. After connectivity establishes between the VTEP and the NSX controller, the management IP address and the connectivity status populate.



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.

Select **Home > Logical Switches > New Logical Switch** and click **Add**. A New Logical Switch dialog window opens. Enter a name and select **Unicast** as the replicate mode and click **OK**. This step configures the logical switch for NSX controller management.

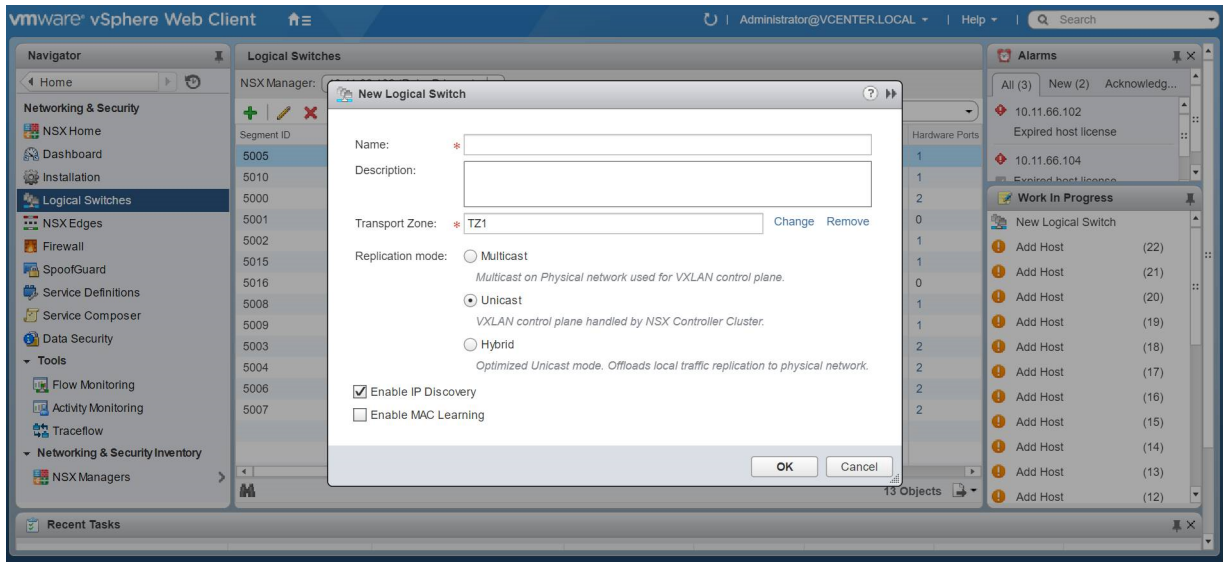
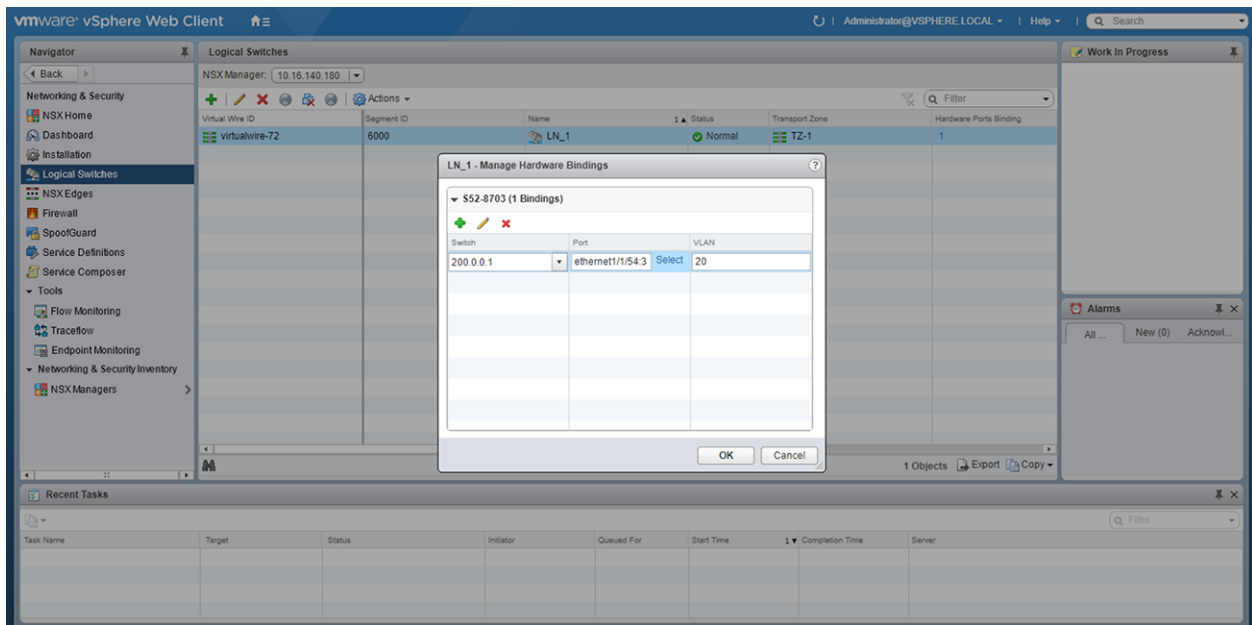


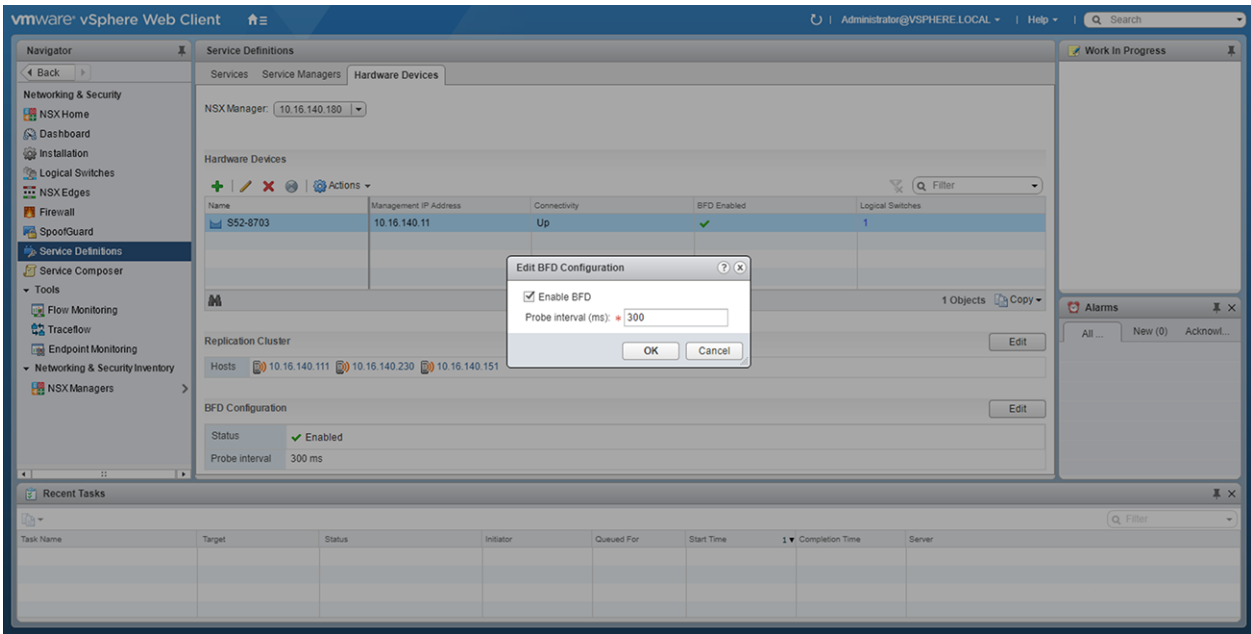
Figure 4. New logical switch

- 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. The logical switch port binds the virtual access ports in the gateway to the logical VXLAN network and server VLAN. Select the **Logical Switches** and from the Manage Hardware Bindings dialog window, expand a VTEP and click **Add**. Select the switch from the Switch column. Enter the hardware port under the port field, and enter the VLAN ID under the VLAN column and click **OK**.



- 5 (Optional) Enable or disable BFD globally.

Go to **Hardware Devices > BFD Configuration**, and click the **Edit** button. The Edit BFD Configuration windows opens. Check or clear the **Enable BFD** check box. You can also change the probe interval if required.

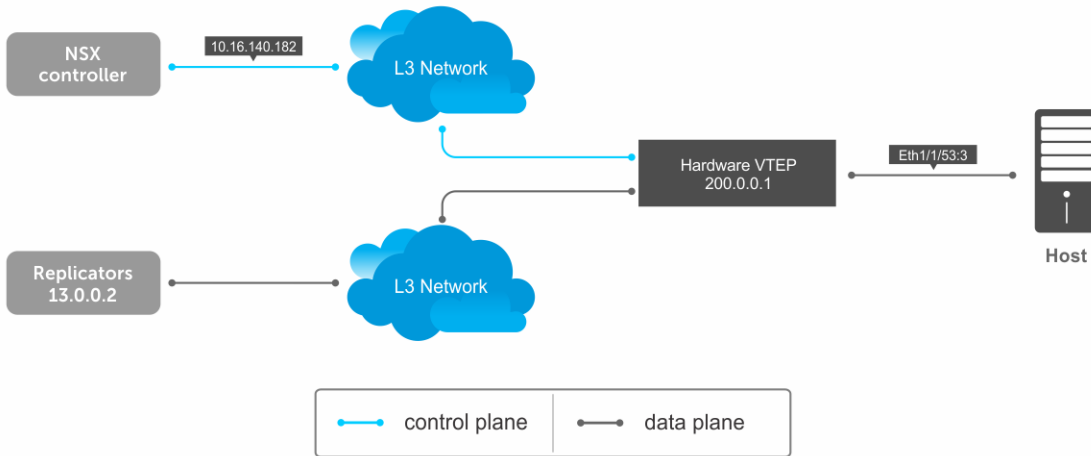


After you configure a VMware NSX controller on a server VM, connect to the controller from the VXLAN gateway switch.

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 OS10 VTEP deployed in VXLAN environment.



To configure an NSX controller-based VXLAN:

- Configure the controller and an interface the controller manages, in the OS10 VTEP
- 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](#).

Configure the controller and interface

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.

- 1 Configure the underlay OSPF protocol.

```
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 a Loopback interface as the VXLAN source tunnel interface.

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

- 4 Configure an OVSDB-based controller (NSX).

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

- 5 Assign an interface that the controller manages.

```
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
```

Verify the controller configuration

To view controller-based information on the VTEP, 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
```

VXLAN Controller commands

controller ovssdb

Enables Controller mode to configure the controller parameters.

Syntax `controller ovssdb`

Parameters None

Default None

Command mode CONFIGURATION-NVE

Usage information The controller configuration initiates the OVSSDB service on the OS10 switch. This populates the controller-managed ports in the OVSSDB server.

The `no` version of this command stops the OVSSDB 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 ovbdb
```

Supported releases 10.4.3.0 or later

ip port ssl

Adds an OVSDDB controller connection, 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 NSX controller to connect with.
- *port-number* — Specify the port number through which the connection to the NSX controller is made.

Default

For an OVSDDB-based controller, configure the following:

- Port number as 6640
- Connection type as SSL

Command mode CONFIGURATION-NVE-OVSDDB

Usage information

The only OVSDDB controller supported is the NSX controller. OS10 supports only one OVSDDB controller connection at a time. You can configure only one set of connection parameters for a controller. This command adds the controller configuration to OVSDDB. The local OVSDDB server initiates the connection to the NSX controller. The `no` version of this command removes the controller configuration from the local OVSDDB server, which in turn removes the connection to the NSX controller.

Example

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

Supported releases 10.4.3.0 or later

max-backoff

Configures the time interval, in milliseconds (ms), that the switch waits between the connection attempts to the controller.

Syntax `max-backoff interval`

Parameters

interval—Enter the amount of time, in ms, that the switch waits between the connection attempts to the controller, from 1000 to 180000 ms.

Default 8000 ms

Command Mode CONFIGURATION-NVE-OVSDDB

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 ovbdb
OS10 (config-nve-ovbdb)# max-backoff 40000
```

Supported Releases 10.4.3.0 or later

nve-controller

Assigns an controller-managed interface for VXLAN configurations.

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 following error message displays:

```
% 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

Usage information This command is available only for the `sysadmin` and `secadmin` roles. This command generates the SSL certificate and restarts the OVSDB server to pick up the newly generated certificate.

NOTE: During an OVSDB server restart, any updates to the server are missed.

Example

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

Supported releases 10.4.3.0 or later

show nve controller

Displays information about the controller and the controller-managed interfaces.

Syntax `show nve controller`

Parameters None
Default None
Command mode EXEC

```
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
```

Supported releases 10.4.3.0 or later

show nve controller ssl-certificate

Displays the SSL certificate generated in the system.

Syntax show nve controller ssl-certificate
Parameters None
Default None
Command mode EXEC
Usage information This command is available only for sysadmin and secadmin roles.

```
OS10# show nve controller
-----BEGIN CERTIFICATE-----
MIIDgDCCAmgCAQMwDQYJKoZIhvcNAQENBQAwgYExCzAJBgNVBAYTAlVTMQswCQYD
VQQIDAJDQTEVMBMGAlUECgwMT3BlbiB2U3dpdGNoMREwDwYDVQQLEDAhd210Y2hj
YTE7MDkGA1UEAwwyT1ZTIHN3aXRjaGNIENBIENlcnRpZmljYXRlICgyMDE4IFNl
cCAyMyAwMzo0NzoyMCKwHhcNMjgwOTI0MTYzMDUyWhcNMjgwOTIxMTYzMDUyWjCB
iTELMakGA1UEBhMCVVMxCzAJBgNVBAGMAkNBMRUwEwYDVQQKDAxPcGVuIHZTd210
Y2g4ZAdBgNVBAAsMFk9wZW4gd1N3aXRjaCBjZlZlZlZlZlZlZlZlZlZlZlZlZlZl
bGwgaWQ6MGVlZmUyWmVhZmVhZmVhZmVhZmVhZmVhZmVhZmVhZmVhZmVhZmVhZmVh
BqkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsM1D4c4fWwy+5t6VScjizlkfSNzE
BOK5PJyI3B6ReRK/J14Fdxio1YmzG0YObjxiwjpUYEsqPL3Nvh0f10KMqwqJVbdf
6sXWHUvW+9A7cIfRh0aRI+HIYyUC4YD48GlnVnaCqhxYaA0tcMzJm4r2k7AjwJU1
0pDXiqS3uJwGmfxlhvmFio8EeHM/Z79DkBRD6FUMwacAnb3yCIKZH50AWq7qRmmG
NZOgYUT+8oaj5t0/hEQfDYuv32E5z4d3FhiBJMFT86T4YvpJYyJkiKmaQWInkthL
V3VxEMXI5vJQclMhwYbKfPB4hh3+qdS5o+uVco76CVrcWi7rO3XmsBkbnQIDAQAB
MA0GCSqGSIB3DQEBDQUAA4IBAQTuFVD20GcHD8zdpYf0YaP4b6TuonUzF0jwoV+
Qr9b4k0jEBGuoPdevX3AeV/dvAa2Q6o1iOBM5z74NgHizhr067pFP841Nv7DAVb7
cPHHSSTTSeeJjIVMh0kv0KkVefsYuI4r1jqJxu0GZgBinqehXxVKlceouLvwbbh1
MFYXN3lce2AXR746q1Vlc6stNkxf3nrlOpSDz3P4VOnbAnIrY+SvUVWAt0tdrowH
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 the active replicator selected for that VNID based on the load-balancing logic.

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
```

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
-----
"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 each OVSSDB manager and parameters including connection status and a list of physical switch names.

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 the physical switch creates.

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_config_local  bfd_params
  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-
fbl1a7bba88fd
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 5ee586b-
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.

Topics:

- [BGP EVPN compared to static VXLAN](#)
- [VXLAN BGP EVPN operation](#)
- [Configure BGP EVPN for VXLAN](#)
- [VXLAN BGP EVPN routing](#)
- [BGP EVPN with VLT](#)
- [VXLAN BGP commands](#)
- [VXLAN EVPN commands](#)
- [Example: VXLAN with BGP EVPN](#)

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.

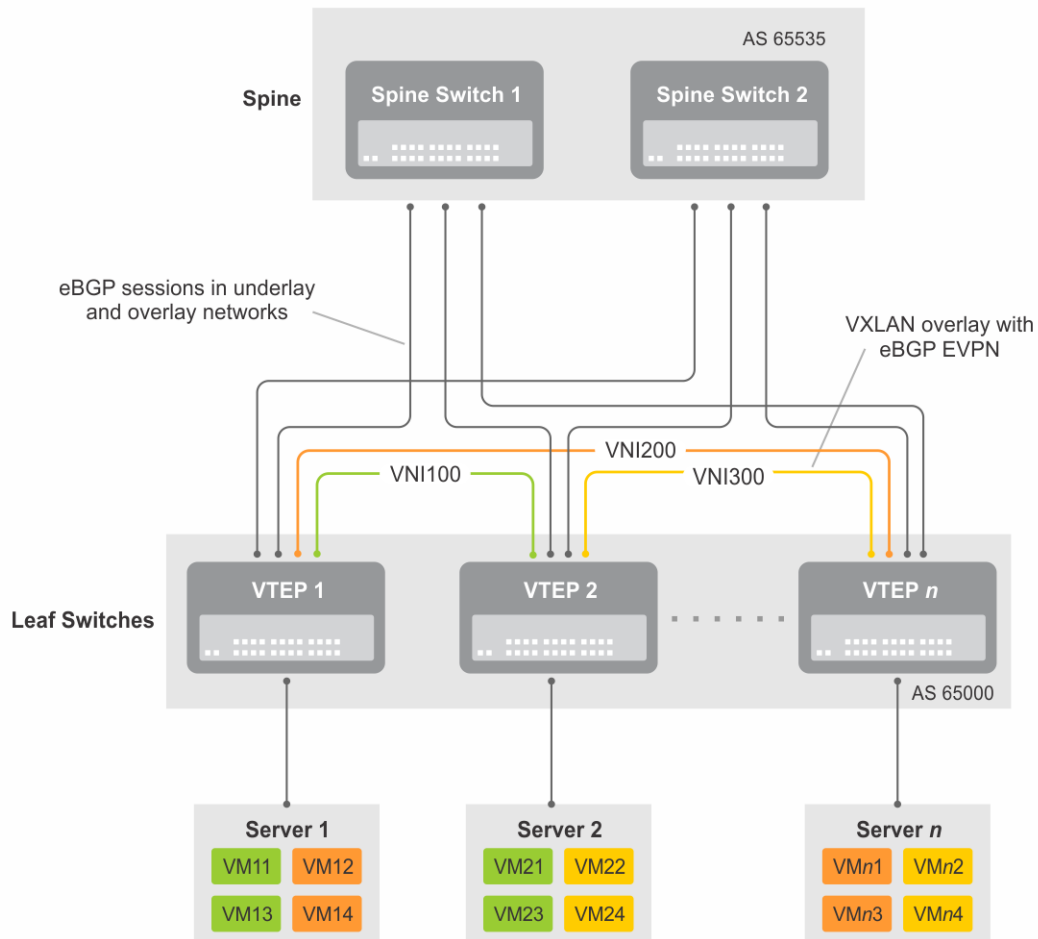


Figure 5. BGP EVPN topology

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 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 auto-generates 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 auto-generates 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 auto-generates 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 auto-configured.

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 auto-generate or manually configure a RD for each EVI. In auto-EVI mode, the RD is auto-generated. In manual EVI configuration mode, you can auto-generate 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 auto-generated 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 auto-generate or manually configure the RT import and export for each EVI. In auto-EVI mode, RT auto-generates. In manual EVI configuration mode, you can auto-generate or manually configure the RT.

The RT consists of a 2-octet *type* and a 6-octet *value*. If you auto-configure 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 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, OS10 can auto-configure RTs for both 2-byte and 4-byte ASNs. 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). Therefore, in 4-byte ASN deployment, OS10 supports RT auto-configuration if the VNID-to-VNI mapping is the same on all VTEPs.

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.

Usage guidelines

- Only L2 gateway EVPN bridging functionality is supported.
- Only EVPN route types 2 and 3 are supported.
- Only asymmetric IRB is supported.

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

- 1 Enable the EVPN control plane in CONFIGURATION mode.

```
evpn
```

- 2 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

- 1 Enable the EVPN control plane in CONFIGURATION mode.

```
evpn
```

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

```
evi id
```

- 3 Configure the Route Distinguisher in EVPN EVI mode.

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

Where:

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

- 4 Configure the RT values in EVPN EVI mode.

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

Where:

- `route-target auto` auto-configures an import and export value for EVPN routes.
- `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. You must 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

VXLAN BGP EVPN routing

[Configure BGP EVPN for VXLAN](#) describes how EVPN facilitates traffic switching within the same L2 tenant segment virtual network on a VTEP for virtual networks that associate with EVIs. 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.

As described in [Configure VXLAN — Enable overlay routing between virtual networks](#), 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.

For more information, see [Configure VXLAN — Enable overlay routing between virtual networks](#).

EVPN supports different types of IRB routing for tenants, VMs and servers, that connect to each VTEP in a tenant network.

- Asymmetric routing: IP routing is performed on ingress VTEPs. L2 bridging is performed on egress VTEPs. You must configure an ingress VTEP with a virtual network even for destination IP subnets that have no locally attached hosts. EVPN asymmetric IRB installs ARP entries to associate each tenant VM IP address with its corresponding VM MAC address in the overlay.
- Symmetric routing: IP routing is performed on both ingress and egress VTEPs. You do not have to configure an ingress VTEP with a virtual network for destination IP subnets that have no locally attached hosts. EVPN symmetric IRB installs ARP entries to associate each tenant VM destination IP address with the MAC address of the VTEP where the VM is located, reducing the number of required hardware next-hop routing resources.

NOTE: In release 10.4.3.0, OS10 supports only distributed asymmetric routing mode.

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

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 EMC 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 LAGs 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.

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

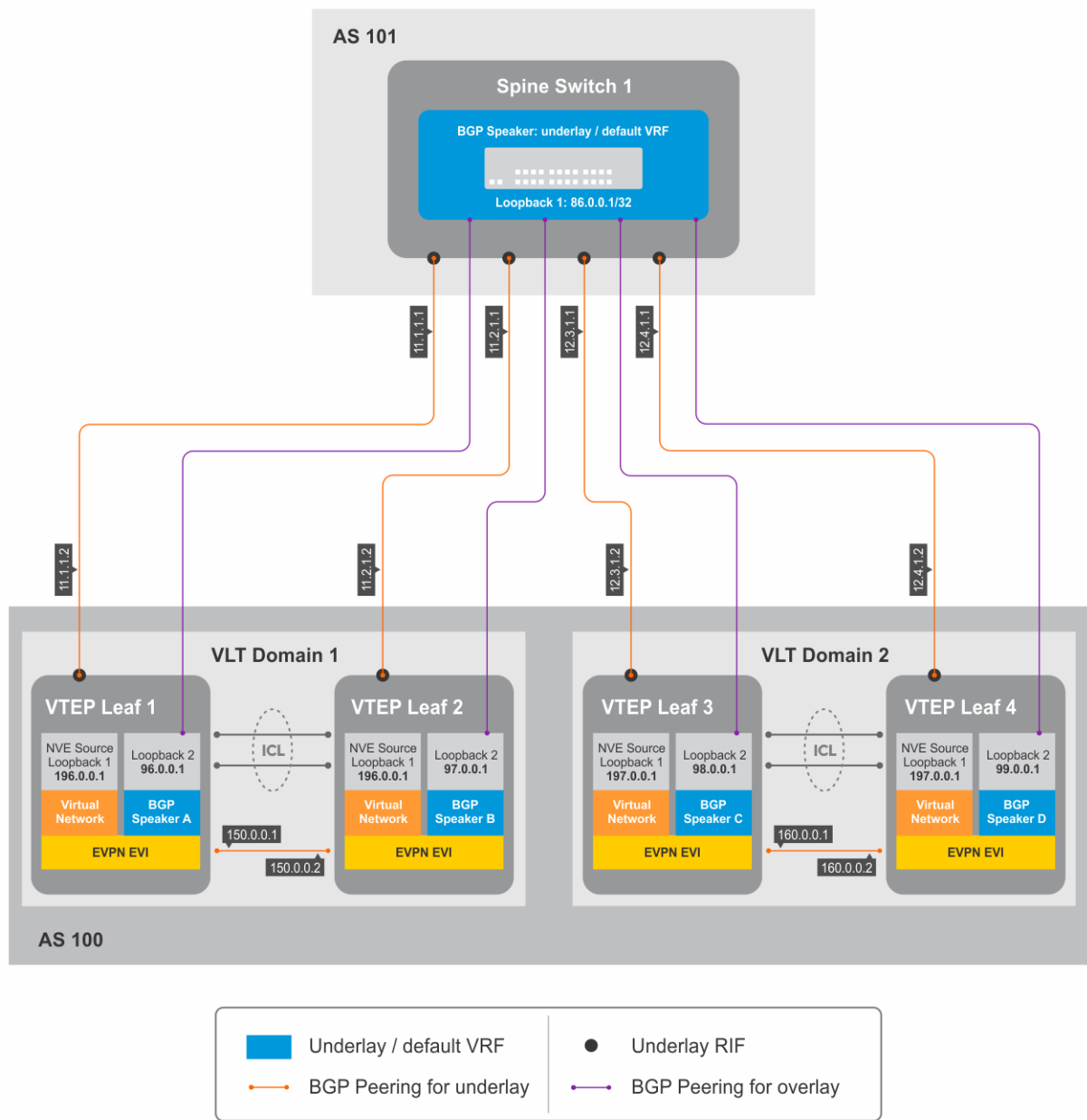


Figure 6. BGP EVPN in VLT domain

VXLAN BGP commands

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 <i>no</i> version of this command to disable an address family with a neighbor.
Example	<pre>OS10 (conf-router-neighbor) # address-family l2vpn evpn unicast OS10 (conf-router-bgp-neighbor-af) # activate</pre>
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	<code>address-family l2vpn evpn</code>
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	<pre>OS10 (config) # router bgp 100 OS10 (config-router-bgp-100) # neighbor 45.0.0.1 OS10 (config-router-neighbor) # address-family l2vpn evpn</pre>
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	<code>allowas-in as-number</code>
Parameters	<i>as-number</i> —Enter the number of occurrences for a local AS number, from 1 to 10.
Default	Disabled
Command Mode	ROUTER-BPG-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 <i>no</i> version of this command resets the value to the default.
Example (IPv4)	<pre>OS10 (config-router-neighbor) # address-family ipv4 unicast OS10 (conf-router-bgp-neighbor-af) # allowas-in 5</pre>
Example (IPv6)	<pre>OS10 (conf-router-template) # address-family ipv6 unicast OS10 (conf-router-bgp-template-af) # allowas-in 5</pre>
Supported Releases	10.3.0E or later

sender-side-loop-detection

Enables the sender-side loop detection process for a BGP neighbor.

Syntax	<code>sender-side-loop-detection</code>
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 <code>neighbor allowas-in</code> command. The <code>no</code> version of this command disables sender-side loop detection for that neighbor.
Example (IPv4)	<pre>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</pre>
Example (IPv6)	<pre>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</pre>
Supported Releases	10.3.0E or later

show ip bgp l2vpn evpn

Displays the internal BGP routes in the L2VPN EVPN address family in EVPN instances.

Syntax	<code>show ip bgp l2vpn evpn [summary neighbors]</code>				
Parameters	<table><tr><td>summary</td><td>Display a summary of the BGP routes in the L2VPN address family that exchange with remote VTEPs.</td></tr><tr><td>neighbors</td><td>Display the remote VTEPs with whom BGP routes in the L2VPN address family exchange.</td></tr></table>	summary	Display a summary of the BGP routes in the L2VPN address family that exchange with remote VTEPs.	neighbors	Display the remote VTEPs with whom BGP routes in the L2VPN address family exchange.
summary	Display a summary of the BGP routes in the L2VPN address family that exchange with remote VTEPs.				
neighbors	Display the remote VTEPs with whom BGP routes in the L2VPN address family exchange.				
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	<pre>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</pre>				

```
[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
...
```

Supported releases 10.4.2.0 or later

VXLAN EVPN commands

auto-evi

Creates an EVPN instance automatically, including Route Distinguisher (RD) and Route Target (RT) values.

Syntax	auto-evi
Parameters	None
Default	Not configured

Command mode	EVPN
Usage information	<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"> • For a 2-byte autonomous system: <ul style="list-style-type: none"> – The RD auto-configures as Type 1 from the overlay network source IP address and the auto-generated EVI index. – The RT auto-configures as Type 0 from the 2-byte AS and the 3-byte VNI—Type encoded as 0x0002. • For a 4-byte autonomous system: <ul style="list-style-type: none"> – The RD auto-configures as Type 1 from the overlay network source IP address and the auto-generated EVI index. – The RT auto-configures as Type 2 from the 4-byte AS and the 2-byte EVI—Type encoded as 0x0202.
Example	<pre>OS10 (config)# evpn OS10 (config-evpn)# auto-evi</pre>
Supported releases	10.4.2.0 or later

evi

Creates an EVPN instance (EVI) in EVPN mode.

Syntax	<code>evi id</code>
Parameters	<p>id Enter the EVPN instance ID, from 1 to 65535.</p>
Default	Not configured
Command mode	EVPN
Usage information	<p>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.</p>
Example	<pre>OS10 (config)# evpn OS10 (config-evpn)# evi 10 OS10 (config-evpn-evi)#</pre>
Supported releases	10.4.2.0 or later

evpn

Enables the EVPN control plane for VXLAN.

Syntax	<code>evpn</code>
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 EVPN routes use.

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

Parameters

A.B.C.D: [1-65535]	Manually configure the RD with a 4-octet IPv4 address then a 2-octet-number, from 1-65535.
auto	Configure the RD to automatically generate.

Default Not configured

Command mode EVPN-EVI

Usage information A RD maintains the uniqueness of an EVPN route between different EVPN instances. The RD auto-configures as Type 1 from the overlay network source IP address and the auto-generated EVPN instance ID.

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
```

Supported releases 10.4.2.0 or later

route-target

Configures the Route Target (RT) values EVPN routes use.

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

Parameters

value {import export both}	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">• The <i>2-octet</i> ASN or number is 1 to 65535.• The <i>4-octet</i> ASN or number is 1 to 4294967295.
auto	Configure the RT import and export values to automatically generate.
asn4	(Optional) Advertises a 4-byte AS number in RT values.

Default Not configured

Command mode EVPN-EVI

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.

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
```

Supported releases 10.4.2.0 or later

show evpn evi

Displays the configuration settings of EVPN instances.

Syntax `show evpn evi [id]`

Parameters *id* — (Optional) Enter the EVPN instance ID, from 1 to 65535.

Default Not configured

Command mode EXEC

Usage information 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.

Example

```
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)
```

Supported releases 10.4.2.0 or later

show evpn mac

Displays BGP EVPN routes for host MAC addresses.

Syntax `show evpn mac {count | mac-address nn.nn.nn.nn | evi id [mac-address nn.nn.nn.nn | count | next-hop ip-address count]}`

Parameters

- *count* — Displays the total number of local and remote host MAC addresses in EVPN instances.
- *mac-address nn.nn.nn.nn* — Displays the BGP EVPN routes for a specific 48-bit host MAC address.

- `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.

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.

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

Supported releases 10.4.3.0 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 vxlan-vni

Displays the VXLAN overlay network for EVPN instances.

Syntax `show evpn vxlan-vni [vni]`

Parameters *vni* — (Optional) Enter the VXLAN virtual-network ID, from 1 to 16,777,215.

Default Not configured

Command mode EXEC

Usage information 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 network ID.

Syntax `vni vni`

Parameters *vni* Enter the virtual-network ID, from 1 to 16,777,215.

Default Not configured

Command mode EVPN-EVI

Usage information Use this command in EVPN-EVI mode to configure an EVPN instance with RD and RT values to an overlay VXLAN virtual network.

Example

```
OS10(config)# evpn
OS10(config-evpn)# evi 10
OS10(config-evpn-evi)# vni 10000
```

Supported releases 10.4.2.0 or later

Example: VXLAN with BGP EVPN

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 65001. All leaf nodes are in another autonomous system—AS 65002.

- 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 VTEP 3: 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-EVI mode.
- On VTEP 4: Access ports are assigned to the virtual network using a port-scoped VLAN. EVPN is configured using manual configuration mode, including the EVPN instance, RD, and RT.

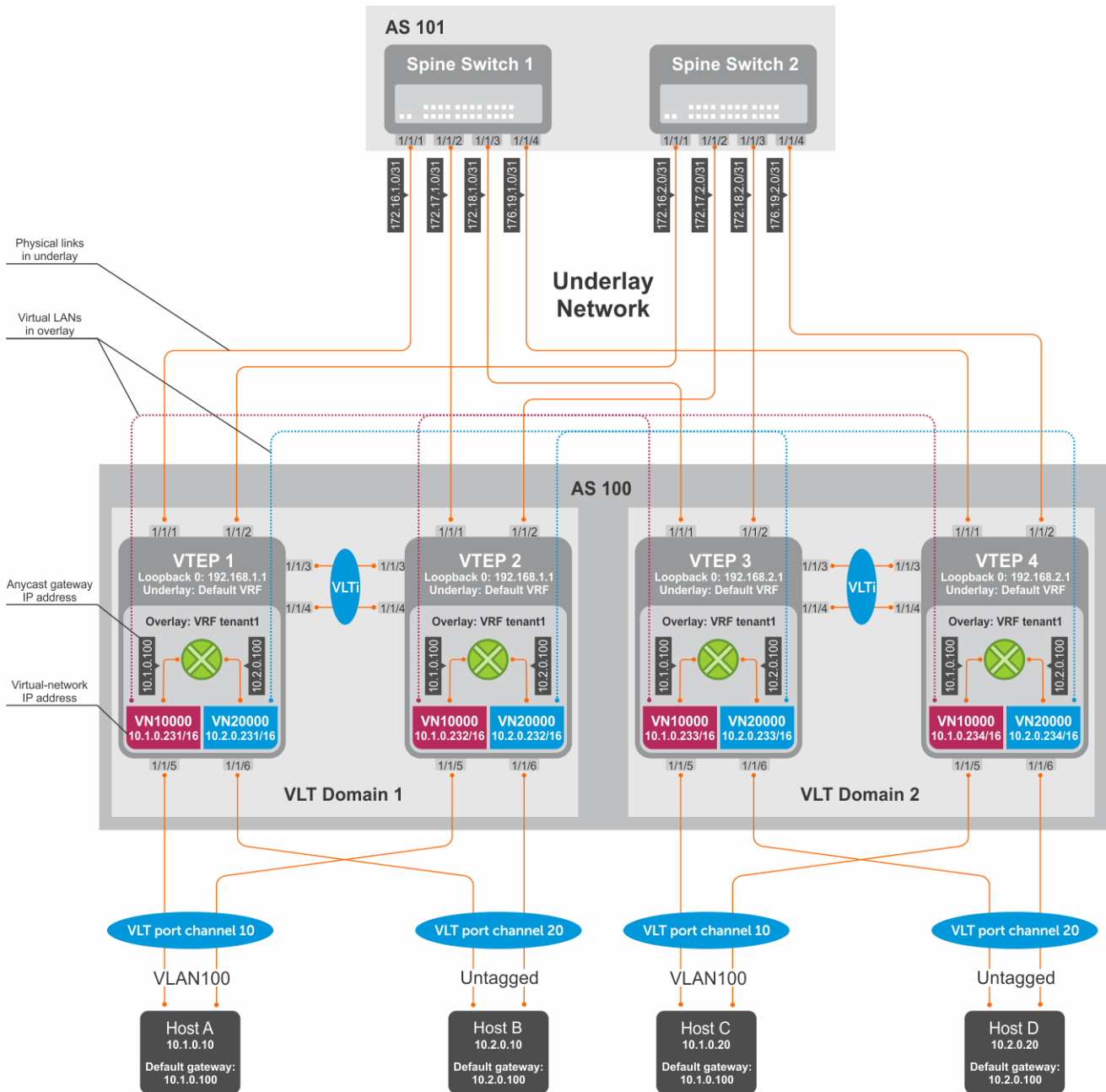


Figure 7. VXLAN BGP EVPN use case

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)# vxlan-vni 10000
OS10(config-vn-vxlan-vni-10000)# exit
OS10(config-vn)# exit
OS10(config)# virtual-network 20000
OS10(config-vn)# vxlan-vni 20000
OS10(config-vn-vxlan-vni-20000)# exit
OS10(config-vn)# 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(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 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 172.16.1.0/31
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 172.16.2.0/31
OS10(conf-if-eth1/1/2)# exit
```

7. Configure eBGP

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-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)# no shutdown
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-neighbor)# 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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
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-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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
OS10(config-router-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-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 172.16.250.0/30
OS10(config-if-vl-4000)# ip 1 area 0.0.0.0
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-network10000
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
```

```
OS10(config)# interface virtual-network20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.231/16
OS10(conf-if-vn-10000)# ip virtual-router address 10.2.0.100
OS10(conf-if-vn-20000)# no shutdown
OS10(conf-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(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 networks

```
OS10(config)# virtual-network 10000
OS10(config-vn)# vxlan-vni 10000
OS10(config-vn-vxlan-vni-10000)# exit
OS10(config-vn)# exit
OS10(config)# virtual-network 20000
OS10(config-vn)# vxlan-vni 20000
OS10(config-vn-vxlan-vni-20000)# exit
OS10(config-vn)# 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(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 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-eth1/1/1)# 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 100
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-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)# no shutdown
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-neighbor)# 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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
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-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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
OS10(config-router-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-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 172.16.250.1/30
OS10(config-if-vl-4000)# ip 1 area 0.0.0.0
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(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 172.16.250.0
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(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 10.1.0.232/16
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

OS10(config)# interface virtual-network20000
OS10(conf-if-vn-20000)# ip vrf forwarding tenant1
OS10(conf-if-vn-20000)# ip address 10.2.0.232/16
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 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)# vxlan-vni 10000
OS10(config-vn-vxlan-vni-10000)# exit
OS10(config-vn)# exit
```

```
OS10(config)# virtual-network 20000
OS10(config-vn)# vxlan-vni 20000
OS10(config-vn-vxlan-vni-20000)# exit
OS10(config-vn)# exit
```

4. Configure unused VLAN ID for untagged membership

```
OS10(config)# virtual-network untagged-vlan 4000
```

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)# 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)# 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 virtual networks

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

```
OS10(config)# virtual-network 20000
OS10(config-vn)# member-interface ethernet 1/1/6 untagged
OS10(config-vn)# 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)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-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)# 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-neighbor)# 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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
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-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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
OS10(config-router-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-neighbor)# exit
OS10(config-router-bgp-100)# exit

```

12. Configure EVPN

Configure the EVPN instance, RD, and RT using manual configuration mode:

```

OS10(config)# evpn
OS10(config-evpn)# evi 1000
OS10(config-evpn-evi)# vni 10000
OS10(config-evpn-evi)# rd auto
OS10(config-evpn-evi)# route-target both auto
OS10(config-evpn-evi)# exit

OS10(config-evpn)# evi 2000
OS10(config-evpn-evi)# vni 20000
OS10(config-evpn-evi)# rd auto
OS10(config-evpn-evi)# route-target both auto
OS10(config-evpn-evi)# exit
OS10(config-evpn)# exit

```

13. Configure VLT

Configure a VLTi VLAN for the virtual network

```

OS10(config)# virtual-network 10000
OS10(config-vn)# vlti-vlan 100
OS10(config-vn)# 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 172.16.250.10/30
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: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 172.16.250.11
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-10000)# 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)# vxlan-vni 10000
OS10(config-vn-vxlan-vni-10000)# exit
OS10(config-vn)# exit
```

```
OS10(config)# virtual-network 20000
OS10(config-vn)# vxlan-vni 20000
OS10(config-vn-vxlan-vni-20000)# exit
OS10(config-vn)# exit
```

4. Configure the unused VLAN ID for untagged membership

```
OS10(config)# virtual-network untagged-vlan 4000
```

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)# 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)# 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 networks

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

```
OS10(config)# virtual-network 20000
OS10(config-vn)# member-interface ethernet 1/1/6 untagged
OS10(config-vn)# 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 172.19.1.0/31
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 172.19.2.0/31
OS10(conf-if-eth1/1/2)# exit
```

8. Configure eBGP

```
OS10(config)# router bgp 100
OS10(config-router-bgp-100)# address-family ipv4 unicast
OS10(configure-router-bgpv4-af)# redistribute connected
OS10(configure-router-bgpv4-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)# 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-neighbor)# 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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
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-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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
OS10(config-router-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-neighbor)# exit
OS10(config-router-bgp-100)# exit
```

12. Configure EVPN

Configure the EVPN instance, RD, and RT using manual configuration mode:

```
OS10(config)# evpn
OS10(config-evpn)# evi 1000
OS10(config-evpn-evi)# vni 10000
OS10(config-evpn-evi)# rd auto
OS10(config-evpn-evi)# route-target both auto
OS10(config-evpn-evi)# exit

OS10(config-evpn)# evi 2000
OS10(config-evpn-evi)# vni 20000
OS10(config-evpn-evi)# rd auto
OS10(config-evpn-evi)# route-target both auto
OS10(config-evpn-evi)# exit
OS10(config-evpn)# exit
```

13. Configure VLT

Configure a VLTi VLAN for the virtual network

```
OS10(config)# virtual-network 10000
OS10(config-vn)# vlti-vlan 100
OS10(config-vn)# 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 172.16.250.11/30
OS10(config-if-vl-4000)# exit
```

Configure 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

```
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: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 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 networks

```
OS10(config)# interface virtual-network10000
OS10(conf-if-vn-10000)# ip vrf forwarding tenant1
OS10(conf-if-vn-10000)# ip address 10.1.0.234/16
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 10.2.0.234/16
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
```

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)# address-family ipv4 unicast
OS10(config-router-bgp-101)# redistribute connected
```

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(config-router-bgp-101)# neighbor 172.17.1.0
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(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)# 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(config-router-bgp-101)# neighbor 172.19.1.0
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(config-router-bgp-101)# 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 101
OS10(config-router-bgp-101)# neighbor 172.16.0.1
OS10(config-router-neighbor)# ebgp-multihop 4
OS10(config-router-neighbor)# remote-as 100
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-neighbor-af)# no activate
OS10(config-router-neighbor-af)# exit
OS10(config-router-neighbor)# address-family l2vpn evpn
OS10(config-router-neighbor-af)# no sender-side-loop-detection
OS10(config-router-neighbor-af)# activate
OS10(config-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)# address-family ipv4 unicast
OS10(config-router-bgp-101)# redistribute connected
```

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 172.202.0.1/32
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

```

Support resources

The Dell EMC Support site provides a range of documents and tools to assist you with effectively using Dell EMC devices. Through the support site you can obtain technical information regarding Dell EMC products, access software upgrades and patches, download available management software, and manage your open cases. The Dell EMC support site provides integrated, secure access to these services.

To access the Dell EMC Support site, go to 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/.

To search for drivers and downloads, see www.dell.com/drivers/.

To participate in Dell EMC community blogs and forums, see www.dell.com/community.