Configuring VXLAN with IPv6 in the Underlay (VXLANv6)

This chapter contains the following sections:

Information About Configuring VXLAN with IPv6 in the Underlay (VXLANv6)

VXLAN BGP EVPN is deployed with IPv4 underlay and IPv4 VTEP. Hosts in the overlay can be IPv4 or IPv6. Support is added for VXLAN with IPv6 in the Underlay (VXLANv6) with an IPv6 VTEP. This requires IPv6 versions of the unicast routing protocols and utilizing ingress replication for multi-destination traffic (BUM) in the underlay.

This solution is targeted for deployments where the VTEP is IPv6 only and the underlay is IPv6. The BGP sessions between the leaf and spine are also IPv6. The overlay hosts can be either IPv4 or IPv6.

VXLANv6 feature supports BGP unnumbered peering in the underlay.

The following protocols are supported in the underlay:

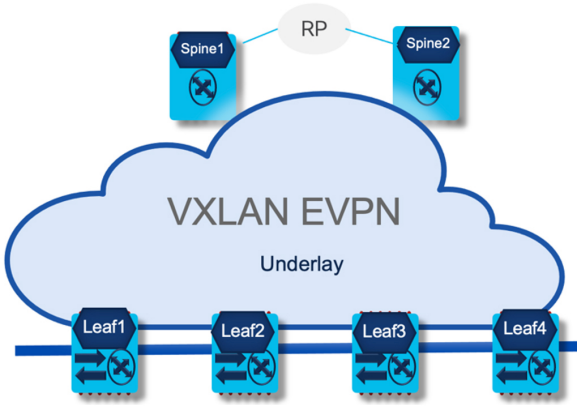
- IS-IS

- OSPFv3
- eBGP

Information About Configuring VXLAN EVPN and TRM with IPv6 Underlay

Beginning with Cisco NX-OS Release 10.4(2)F, the support is provided for VXLAN with IPv6 Multicast in the Underlay. Hosts in the overlay can be IPv4 or IPv6. This requires IPv6 versions of the unicast routing protocols and using IPv6 multicast in the underlay (PIMv6). Any multi-destination overlay traffic (such as TRM, BUM) can use the IPv6 multicast underlay.

Figure 1: Topology - VXLAN EVPN with IPv6 Multicast Underlay



The above topology shows four leafs and two spines in a VXLAN EVPN fabric. The underlay is an IPv6 Multicast running PIMv6. RP is positioned in the spine with anycast RP.

Beginning with Cisco NX-OS Release 10.4(3)F, the combination of PIMv6 underlay on the fabric side and Ingress Replication (IPv6) on Data Center Interconnect (DCI) side is supported on Cisco Nexus 9300-FX/FX2/FX3/GX/GX2/H2R/H1 ToR switches and 9500 switches with X9716D-GX and X9736C-FX line cards.

Information About vPC and VXLAN with IPv6 in the Underlay (VXLANv6)

vPC VTEPs use vMAC (virtual MAC) with the VIP/PIP feature. vMAC is used with VIP and the system MAC is used with PIP.

In the IPv4 underlay, vMAC is derived from the IPv4 VIP address:

VMAC = 0x02 + 4 bytes IPv4 VIP address.

In the IPv6 underlay, VIP is IPv6 (128 bits) which cannot be used to generate a conflict free unique vMAC (48 bits). The default method is to autogenerate the vMAC by picking the last 48 bits from the IPv6 VIP:

Autogenerated vMAC = 0x06 + the last 4 bytes of the IPv6 VIP address.

If there are two vPC complexes which have different VIPs but the same last 4 bytes of IPv6 address in the VIP, both autogenerate the same vMAC. For a remote VTEP, it sees vMAC flopping between two different VIPs. This is not an issue for Cisco Nexus 9000 Series switches which support VXLAN IPv6.

For other vendor boxes, if this is an issue for interoperability reasons, the vMAC can be manually configured on Cisco Nexus 9000 Series switches to override the autogenerated vMAC. The default behavior for VXLAN with IPv6 in the Underlay (VXLANv6) is to autogenerate the VMAC. If a VMAC is configured manually, the manually configured VMAC takes precedence.

```
interface nve1
    virtual-rmac <48-bit mac address>
```

The VMAC must be managed by the administrator just like the VIP/PIP and must be unique in the fabric. All the preceding behavior is for VXLAN with IPv6 in the Underlay (VXLANv6) only and nothing changes about VMAC creation and advertisement for VXLAN IPv4 in the underlay.

The default behavior is that vMAC is autogenerated from the configured VIP and advertised. There is no need to use the **virtual-rmac** command as previously described except for interoperability cases. There is no need to use the existing **advertise virtual-rmac** command for VXLAN with IPv6 in the Underlay (VXLANv6).

Information About vPC Peer Keepalive and VXLAN with IPv6 in the Underlay (VXLANv6)

The modification for vPC is to allow IPv6 addresses to be used for the peer-keepalive link. The link can be on the management interface or any other interface. The keepalive link becomes operational only when both peers are configured correctly either with the IPv4 or IPv6 address and those addresses are reachable from each peer. Peer-keepalive can be configured on in-band and out-of-band interfaces.

Note peer-keepalive must be a global unicast address.

The configuration command for **peer-keepalive** accepts an IPv6 address

```
vpc domain 1
peer-keepalive destination 001:002::003:004 source 001:002::003:005 vrf management
```

Guidelines and Limitations for VXLAN with IPv6 in the Underlay (VXLANv6)

VXLAN with IPv6 in the Underlay (VXLANv6) has the following guidelines and limitations:

- Dual Stack (IPv4 and IPv6) is not supported for VXLAN underlay. It should either be IPv4 or IPv6, not both.
- NVE Source interface loopback for VTEP can either be IPv4 (VXLANv4) or IPv6 (VXLANv6), and not both.

- Next hop address in overlay (in bgp l2vpn evpn address family updates) should be resolved in underlay URIB to the same address family. For example, the use of VTEP (NVE source loopback) IPv4 addresses in fabric should only have BGP l2vpn evpn peering over IPv4 addresses.
- Usage of IPv6 LLA requires the TCAM Region for **ing-sup** to be re-carved from the default value of 512 to 768. This step requires a copy run start and reload

The following Cisco Nexus platforms are supported to provide the VTEP function (leaf and border). The BGP route reflector can be provided by any Cisco Nexus platform that supports the EVPN **address-family** command over an IPv6 MP-BGP peering.

- Cisco Nexus 9332C
- Cisco Nexus 9364C
- Cisco Nexus 9300-EX
- Cisco Nexus 9300-FX
- Cisco Nexus 9300-FX2
- Cisco Nexus 9300-FX3
- Cisco Nexus 9300-FXP
- Cisco Nexus 9300-GX
- Cisco Nexus 9300-GX2
- Cisco Nexus 9332D-H2R
- Cisco Nexus 93400LD-H1
- Cisco Nexus 9364C-H1

VXLAN with IPv6 in the Underlay (VXLANv6) supports the following features:

- Address Resolution Protocol (ARP) suppression in the overlay
- Access Control List (ACL) Quality of Service (QoS)
- Border Node with VRF-Lite
- Dynamic Host Configuration Protocol (DHCP)
- Guestshell support
- Internet Group Management Protocol (IGMP) Snooping in the overlay
- Virtual Extensible Local Area Network (VXLAN) Operation, Administration, and Maintenance (OAM)
- Storm Control for host ports (Access Side)
- Virtual Port Channel (vPC) with VIP and PIP support
- VXLAN Policy-Based Routing (PBR)
- vPC Fabric Peering
- VXLAN Access Features

- Private VLAN (PVLAN)
- 802.1x
- Port security
- Port VLAN translation
- QinVNI
- SelQinVNI
- QinQ QinVNI

VXLAN with IPv6 in the Underlay (VXLANv6) does not support the following features:

- Downstream VNI
- Bidirectional Forwarding Detection (BFD)
- Centralized Route Leak
- Cisco Data Center Network Manager (DCNM) integration
- Cross Connect
- EVPN Multi-homing with Ethernet Segment (ES)
- Fabric Extender (FEX) attached to a VXLAN-enabled switch.
- VXLAN Flood and Learn
- MACsec
- Multiprotocol Label Switching (MPLS) and Locator/ID Separation Protocol (LISP) handoff
- Multicast underlay (PIM-BiDir, Protocol Independent Multicast (PIM) Any Source Multicast (ASM), Snooping)
- NetFlow
- Overlay IGMP Snooping
- **peer vtep** command
- Sampled Flow (sFLOW)
- Static ingress replication (IR)
- Tenant Routed Multicast (TRM)
- Virtual Network Functions (VNF) Multipath
- VXLAN Multi-Site

Beginning with Cisco NX-OS Release 10.1(1), IPv6 Underlay is supported for N9K-C9316D-GX, N9K-C93600CD-GX, and N9K-C9364C-GX TOR switches.

Beginning with Cisco NX-OS Release 10.2(3)F, IPv6 Underlay is supported on Cisco Nexus 9700-EX/FX/GX line cards.

Beginning with Cisco NX-OS Release 10.3(2)F, vPC fabric peering with IPv6 underlay is supported on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 switches.

Beginning with Cisco NX-OS Release 10.4(1)F, vPC fabric peering with IPv6 underlay is supported on Cisco Nexus 9332D-H2R switches.

Beginning with Cisco NX-OS Release 10.4(2)F, vPC fabric peering with IPv6 underlay is supported on Cisco Nexus 93400LD-H1 switches.

Beginning with Cisco NX-OS Release 10.4(3)F, vPC fabric peering with IPv6 underlay is supported on Cisco Nexus 9364C-H1 switches.

Beginning with Cisco NX-OS Release 10.2(3)F, the VTEP function (leaf and border) is supported on Cisco Nexus 9300-GX2 platform switches.

Beginning with Cisco NX-OS Release 10.4(1)F, the VTEP function (leaf and border) is supported on Cisco Nexus 9332D-H2R switches.

Beginning with Cisco NX-OS Release 10.4(2)F, the VTEP function (leaf and border) is supported on Cisco Nexus 93400LD-H1 switches.

Beginning with Cisco NX-OS Release 10.4(3)F, the VTEP function (leaf and border) is supported on Cisco Nexus 9364C-H1 switches.

Beginning with Cisco NX-OS Release 10.2(3)F, VXLAN PBR is supported with VXLAN v6 underlay on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2 platforms, N9K-C9364C, and N9K-C9332C ToR switches.

Beginning with Cisco NX-OS Release 10.4(1)F, VXLAN PBR is supported with VXLAN v6 underlay on Cisco Nexus 9332D-H2R switches.

Beginning with Cisco NX-OS Release 10.4(2)F, VXLAN PBR is supported with VXLAN v6 underlay on Cisco Nexus 93400LD-H1 switches.

Beginning with Cisco NX-OS Release 10.4(3)F, VXLAN PBR is supported with VXLAN v6 underlay on Cisco Nexus 9364C-H1 switches.

Beginning with Cisco NX-OS Release 10.2(3)F, IPv6 Underlay is supported on Cisco Nexus 9300-GX2 switches.

Beginning with Cisco NX-OS Release 10.4(1)F, IPv6 Underlay is supported on Cisco Nexus 9332D-H2R switches.

Beginning with Cisco NX-OS Release 10.4(2)F, IPv6 Underlay is supported on Cisco Nexus 93400LD-H1 switches.

Beginning with Cisco NX-OS Release 10.4(3)F, IPv6 Underlay is supported on Cisco Nexus 9364C-H1 switches.

The IPv6 Underlay is supported on the following features for VXLAN EVPN:

- Private VLAN (PVLAN) on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1, and Cisco Nexus 9500 switches with Nexus 9700-EX/FX/GX line cards.
- 802.1x on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1, and Cisco Nexus 9500 switches with Nexus 9700-EX/FX/GX line cards.
- Port security on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1, and Cisco Nexus 9500 switches with Nexus 9700-EX/FX/GX line cards.
- Port VLAN translation on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1, and Cisco Nexus 9500 switches with Nexus 9700-EX/FX/GX line cards.

- QinVNI on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1 platform switches.
- SelQinVNI on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1 platform switches.
- QinQ-QinVNI on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2/H2R/H1 platform switches.

Other guidelines and limitations:

- VXLAN/Fibre Channel co-existence

Guidelines and Limitations for VXLAN EVPN and TRM with IPv6 in the Multicast Underlay

VXLAN EVPN and TRM with IPv6 Multicast Underlay has the following guidelines and limitations:

- Spine-based static RP is supported in underlay.
- Cisco Nexus 9300-FX, FX2, FX3, GX, GX2, H2R, and H1 ToR switches are supported as the leaf VTEP.
- Cisco Nexus X9716D-GX and X9736C-FX line cards are supported only on the spine (EoR).

Note EoR requires configuring a non-default template using one of the following commands in global configuration mode:

- **system routing template-multicast-heavy**
- **system routing template-multicast-ext-heavy**

- OSPFv3, eBGP underlay is supported.
- PIMv6 ASM (sparse mode) is supported in underlay.
- PIMv6 Anycast RP is supported in underlay as RP redundancy.
- Underlay IPv6 Multicast is supported.
- Underlay IPv6 Multicast is not supported on EOR platforms as a leaf.
- For overlay traffic, each Cisco Nexus 9000 leaf switch is an RP. External RP is also supported.
- EVPN TRMv4 and TRMv6 with IPv6 Multicast Underlay are supported on the Fabric.
- Fabric Peering and Multisite are not supported with IPv6 multicast underlay.
- GPO is not supported with IPv6 multicast underlay.
- For EVPN TRMv4 and TRMv6 with IPv6 Multicast Underlay, the TCAM region for ingress sup region must be carved to 768.
  - Check the ingress sup region using **show hardware access-list tcam region** command.
  - If the ingress sup region is not 768 or above, you must configure using the **hardware access-list tcam region ing-sup 768** command.

Note If you get an error, "Aggregate ingress TCAM allocation failure" while configuring ing-sup as 768, you must borrow the amount from other TCAM regions.

- Reload the device after this configuration.

Configuring the VTEP IP Address

SUMMARY STEPS

1. **configure terminal**
2. **interface nve1**
3. **source-interface loopback** *src-if*
4. **exit**
5. **interface loopback** *loopback_number*
6. **ipv6 address** *ipv6_format*
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter global configuration mode.
Step 2	**interface nve1** **Example:** `switch(config)# interface nve1`	Configure the NVE interface.
Step 3	**source-interface loopback** *src-if* **Example:** `switch(config-if-nve)# source interface loopback 1`	The source interface must be a loopback interface that is configured on the switch with a valid /128 IP address. This /128 IP address must be known by the intermediate devices in the transport network and the remote VTEPs. This is accomplished by advertising it through a dynamic routing protocol in the transport network. **Note** The IPv6 address on **loopback1** must be a /128 address. The VTEP IP address cannot be a link local IPv6 address.
Step 4	**exit** **Example:**	Exit configuration mode.

	Command or Action	Purpose
	`switch(config-if-nve)# exit`	
Step 5	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 1`	Configure the loopback interface.
Step 6	**ipv6 address** *ipv6_format* **Example:** `switch(config-if)# ipv6 address 2001:db8:0:0:1:0:0:1/128`	Configure IPv6 address on the interface.
Step 7	**exit** **Example:** `switch(config-if)# exit`	Exit configuration mode.

Configuring vPC for VXLAN with IPv6 in the Underlay (VXLANv6)

VXLAN with IPv4 in the underlay leveraged the concept of a secondary IP address (VIP) used in vPC. IPv6 does not have the concept of secondary addresses as does IPv4. However, multiple IPv6 global addresses can be configured on an interface, which are treated equally in priority.

The CLI for the VIP configuration has been extended to specify the loopback interface that carries the VIP if there is a VXLAN with IPv6 in the Underlay (VXLANv6) vPC. The IPv6 primary IP address (PIP) and VIP are in two separate loopback interfaces.

Similar to IPv4, if there are multiple IPv6 addresses specified on either loopback, the lowest IP is selected for each.

The following steps outline the configuration of a VTEP IP (VIP/PIP) required on a vPC setup.

Note MVPN VRI ID must be configured for TRM in a vPC. This same VRI id must be configured on both vPC nodes that are part of the same vPC complex. However, each VRI ID must be unique within the network. This implies that two different vPC pairs must have distinct VRI ID configurations to ensure correct routing and avoid any conflicts.

The **anycast loopback** command is used only for VXLAN with IPv6 in the Underlay (VXLANv6).

SUMMARY STEPS

1. **configure terminal**
2. **interface nve1**
3. **source-interface loopback** *src-if* **anycast loopback** *any-if*
4. **exit**
5. **interface loopback** *loopback_number*
6. **ipv6 address** *ipv6_format*
7. **exit**

8. **interface loopback** *loopback_number*
9. **ipv6 address** *ipv6_format*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enter global configuration mode.
Step 2	**interface nve1** **Example:** `switch(config)# interface nve1`	Configure the NVE interface.
Step 3	**source-interface loopback** *src-if* **anycast loopback** *any-if* **Example:** `switch(config-if-nve)# source interface loopback 1 anycast loopback 2`	The source interface must be a loopback interface that is configured on the switch with a valid /128 IP address. This /128 IP address must be known by the transient devices in the transport network and the remote VTEPs. This is accomplished by advertising it through a dynamic routing protocol in the transport network. **Note** The IPv6 address on loopback1, the primary IP address (PIP), and loopback2, the secondary IP address ()VIP), must be a /128 address. The VTEP IP address cannot be a link local IPv6 address.
Step 4	**exit** **Example:** `switch(config-if-nve)# exit`	Exit configuration mode.
Step 5	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 1`	Configure the loopback interface.
Step 6	**ipv6 address** *ipv6_format* **Example:** `switch(config-if)# ipv6 address 2001:db8:0:0:1:0:0:1/128`	Configure IPv6 address on the interface.
Step 7	**exit** **Example:** `switch(config-if)# exit`	Exit configuration mode.
Step 8	**interface loopback** *loopback_number* **Example:**	Configure the loopback interface.

	Command or Action	Purpose
	`switch(config)# interface loopback 2`	
Step 9	**ipv6 address** *ipv6_format* **Example:** `switch(config-inf)# ipv6 address 2001:db8:0:0:1:0:0:2/128`	Configure IPv6 address on the interface.

Configuring VXLAN EVPN and TRM with IPv6 Multicast Underlay

Configuring IPv6 multicast underlay in the VXLAN fabric involves the following:

- Configuring L2-VNI Based Multicast Group in Underlay, on page 11
- Configuring L3-VNI Based Multicast Group in Underlay, on page 12
- Enabling PIMv6 for Underlay, on page 13

Configuring L2-VNI Based Multicast Group in Underlay

Under NVE configuration on a leaf, IPv6 multicast group (IPv6) is configured for each L2-VNI (VLAN).

SUMMARY STEPS

1. **configure terminal**
2. **interface nve1**
3. **member vni** *vni*
4. **mcast-group** *ipv6-prefix*
5. **global mcast-group** *ipv6-multicast-group* **l2**
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface nve1** **Example:** `switch(config)# interface nve1`	Configures the NVE interface.
Step 3	**member vni** *vni* **Example:** `switch(config-if-nve)# member vni 10501`	Configures the Layer 2 virtual network identifier.

	Command or Action	Purpose
Step 4	**mcast-group** *ipv6-prefix* **Example:** `switch(config-if-nve-vni)# mcast-group ff04::40`	Builds the default multicast distribution tree for the Layer 2 VNI.
Step 5	**global mcast-group** *ipv6-multicast-group* **l2** **Example:** `switch(config-if-nve)# global mcast-group ff04::40 l2`	Configures the global multicast group for the Layer 2 VNI.
Step 6	**exit** **Example:** `switch(config-if-nve)# exit`	Exits configuration mode.

Configuring L3-VNI Based Multicast Group in Underlay

IPv6 multicast group (IPv6) is configured for each L3-VNI (VRF).

SUMMARY STEPS

1. **configure terminal**
2. **interface nve1**
3. **member vni** *vni* **associate-vrf**
4. **mcast-group** *ipv6-prefix*
5. **global mcast-group** *ipv6-multicast-group* **l3**
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface nve1** **Example:** `switch(config)# interface nve1`	Configures the NVE interface.
Step 3	**member vni** *vni* **associate-vrf** **Example:** `switch(config-if-nve)# member vni 50001 associate-vrf`	Associates L3VNI to VRF.
Step 4	**mcast-group** *ipv6-prefix* **Example:**	Builds the default multicast distribution tree for the Layer 3 VNI.

	Command or Action	Purpose
	`switch(config-if-nve-vni)# mcast-group ff10:0:0:1::1`	
Step 5	**global mcast-group** *ipv6-multicast-group* **l3** **Example:** `switch(config-if-nve)# global mcast-group ff04::40 l3`	Configures the global multicast group for the Layer 3 VNI.
Step 6	**exit** **Example:** `switch(config-if-nve)# exit`	Exits configuration mode.

Enabling PIMv6 for Underlay

PIMv6 in and underlay is configured as follows:

SUMMARY STEPS

1. **configure terminal**
2. **interface loopback** *number*
3. **ipv6 address** *ipv6-prefix*
4. **ipv6 pim sparse-mode**
5. **interface nve1**
6. **source-interface loopback** *number*
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface loopback** *number* **Example:** `switch(config)# interface loopback 1`	Configures an interface loopback. This example configures interface loopback 1.
Step 3	**ipv6 address** *ipv6-prefix* **Example:** `switch(config-if)# ipv6 address 11:0:0:1::1/128`	Configures an IP address for this interface. It should be a unique IP address that helps to identify this router.
Step 4	**ipv6 pim sparse-mode** **Example:** `switch(config-if)# ipv6 pim sparse-mode`	Enables PIM6 sparse mode.

	Command or Action	Purpose
Step 5	**interface nve1** **Example:** `switch(config-if)# interface nve1`	Configures the NVE interface.
Step 6	**source-interface loopback** *number* **Example:** `switch(config-if-nve)# source-interface loopback 1`	Configures an source interface loopback.
Step 7	**exit** **Example:** `switch(config-if-nve)# exit`	Exits configuration mode. **Note** For the PIMv6 configuration see the *Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide*. For the TRM configuration see the *Cisco Nexus 9000 Series NX-OS VXLAN Configuration Guide*.

Example Configurations for VXLAN with IPv6 in the Underlay (VXLANv6)

The following are configuration examples for VXLAN with IPv6 in the Underlay (VXLANv6):

With IPv6 address set/match in next-hop, BGP must set/match the IPv6 next-hop address in route type-2 (MAC-IP) and route type-5 (IP Prefix).

Under route-map:

```
set ipv6 next-hop <vtep address>
match ipv6 next-hop  <vtep address>
```

BGP Underlay

Note BGP IPv6 neighbor must support L2VPN EVPN address-family session.

Note The router ID in VXLAN with IPv6 in the Underlay (VXLANv6) must be an IPv4 address.

The BGP router ID is a 32-bit value that is often represented by an IPv4 address. By default, Cisco NX-OS sets the router ID to the IPv4 address of a loopback interface on the router. For VXLAN with IPv6 in the Underlay (VXLANv6), none of the loopbacks need to have an IPv4 address in which case the default selection of router ID does not happen correctly. You can configure the router ID manually to an IPv4 address.

BGP RD (Route distinguisher) which is 64 bits in length can be configured using the autonomous system number of the 4-byte IP address. For VXLAN with IPv6 in the Underlay (VXLANv6), when using an IP address for configuring RD, you must use IPv4 as in the case of VXLAN IPv4.

```
feature bgp
nv overlay evpn

router bgp 64496
   ! IPv4 router id
   router-id  35.35.35.35
   ! Redistribute the igp/bgp routes
   address-family ipv6 unicast
     redistribute direct route-map allow

   ! For IPv6 session, directly connected peer interface
   neighbor 2001:DB8:0:1::55
     remote-as 64496
     address-family ipv6 unicast
```

OSPFv3 Underlay

```
feature ospfv3

router ospfv3 201
router-id 290.0.2.1

interface ethernet 1/2
ipv6 address 2001:0DB8::1/48
ipv6 ospfv3 201 area 0.0.0.10
```

IS-IS Underlay

```
router isis Enterprise
is-type level-1
net 49.0001.0000.0000.0003.00

interface ethernet 2/1
ipv6 address 2001:0DB8::1/48
isis circuit-type level-1
ipv6 router isis Enterprise
```

Example Configuration for VXLAN EVPN and TRM with IPv6 Multicast Underlay

In the following examples, the sample configuration for the leaf, spine, and RP are shown:

- **Leaf - Sample configuration of IPv6 multicast underlay:**
  - NVE Configuration

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 10501
    mcast-group ff04::40
  member vni 50001 associate-vrf
    mcast-group ff10:0:0:1::1
```

  - PIMv6 Configuration

```
feature pim6

ipv6 pim rp-address 101:101:101:101::101 group-list ff00::/8

interface loopback1
  ipv6 address 172:172:16:1::1/128
  ipv6 pim sparse-mode

interface Ethernet1/27
  ipv6 address 27:50:1:1::1/64
  ospfv3 hello-interval 1
  ipv6 router ospfv3 v6u area 0.0.0.0
  ipv6 pim sparse-mode
  no shutdown
```

  - BGP Configuration

```
router bgp 100
    router-id 172.16.1.1
    address-family ipv4 unicast
      maximum-paths 64
      maximum-paths ibgp 64
    address-family ipv6 unicast
      maximum-paths 64
      maximum-paths ibgp 64
    address-family ipv4 mvpn
    address-family l2vpn evpn
    neighbor 172:17:1:1::1
      remote-as 100
      update-source loopback0
      address-family ipv4 mvpn
        send-community
        send-community extended
      address-family ipv6 mvpn
        send-community
        send-community extended
      address-family l2vpn evpn
        send-community
     neighbor 172:17:2:2::1
       remote-as 100
       update-source loopback0
```

```
      address-family ipv4 mvpn
        send-community
        send-community extended
      address-family ipv6 mvpn
        send-community
        send-community extended
      address-family l2vpn evpn
        send-community
        send-community extended
    vrf VRF1
      reconnect-interval 1
      address-family ipv4 unicast
        network 150.1.1.1/32
        advertise l2vpn evpn
        redistribute hmm route-map hmmAdv

evpn
  vni 10501 l2
    rd auto
    route-target import auto
    route-target export auto
vrf context VRF1
  vni 50001
rd auto
address-family ipv4 unicast
   route-target both auto
   route-target both auto mvpn
   route-target both auto evpn
address-family ipv6 unicast
   route-target both auto
   route-target both auto mvpn
   route-target both auto evpn

Note: Incase of vPC leafs, you need to configure identical "mvpn vri id" on both
the vPC nodes. For example:

router bgp 100
  mvpn vri id 2001
```

Note MVPN VRI ID must be unique within the network or setup. That is, if the network has three different sets of vPC pairs, each pair must have a different VRI ID.

- **Spine - sample configuration of IPv6 multicast underlay:**
  - NVE Configuration

    ```
    nv overlay evpn
    ```

  - PIMv6 Configuration

    ```
    feature pim6

    ipv6 pim rp-address 101:101:101:101::101 group-list ff00::/8
    ipv6 pim anycast-rp 101:101:101:101::101 102:102:102:102::102
    ipv6 pim anycast-rp 101:101:101:101::101 103:103:103:103::103

    interface loopback101
      ipv6 address 101:101:101:101::101/128
      ipv6 router ospfv3 v6u area 0.0.0.0
      ipv6 pim sparse-mode
    ```

```
interface loopback102
  ipv6 address 102:102:102:102::102/128
  ipv6 router ospfv3 v6u area 0.0.0.0
  ipv6 pim sparse-mode

interface Ethernet1/50/1
  ipv6 address 27:50:1:1::2/64
  ipv6 pim sparse-mode
  no shutdown
```

- BGP Configuration

```
feature bgp

router bgp 100
        router-id 172.16.40.1
         address-family ipv4 mvpn
        address-family ipv6 mvpn
        address-family l2vpn evpn
        neighbor 172:16:1:1::1
          remote-as 100
          update-source loopback0
          address-family ipv4 mvpn
            send-community
            send-community extended
             route-reflector-client
          address-family ipv6 mvpn
           send-community
            send-community extended
             route-reflector-client
          address-family l2vpn evpn
          send-community
            send-community extended
             route-reflector-client
```

Verifying VXLAN with IPv6 in the Underlay (VXLANv6)

To display the status for the VXLAN with IPv6 in the Underlay (VXLANv6) configuration, enter one of the following commands:

Table 1: VXLAN with IPv6 in the Underlay (VXLANv6) Verification Commands

Command	Purpose
show running-config interface nve 1	Displays interface NVE 1 running configuration information.
show nve interface 1 detail	Displays NVE interface detail.
show nve peers	Displays the peering time and VNI information for VTEP peers.
show nve vni ingress-replication	Displays NVE VNI ingress replication information.
show nve peers 2018:1015::abcd:1234:3 int nv1 counters	Displays NVE peers counter information.

Command	Purpose
show bgp l2vpn evpn 1012.0383.9600	Displays BGP L2VPN information for route type 2.
show bgp l2vpn evpn 303:304::1	Displays BGP L2VPN EVPN for route type 3.
show bgp l2vpn evpn 5.116.204.0	Displays BGP L2VPN EVPN for route type 5.
show l2route peerid	Displays L2route peerid.
show l2route topology detail	Displays L2route topology detail.
show l2route evpn imet all detail	Displays L2route EVPN imet detail.
show l2route fl all	Display L2route flood list detail.
show l2route mac all detail	Displays L2route MAC detail.
show l2route mac-ip all detail	Displays MAC address and host IP address.
show ip route 1.191.1.0 vrf vxlan-10101	Displays route table for VRF.
show forwarding ipv4 route 1.191.1.0 detail vrf vxlan-10101	Displays forwarding information.
show ipv6 route vrf vxlan-10101	Displays IPv6 routing table.
show bgp l2vpn evpn	Displays BGP's updated routes.
show bgp evi *evi-id*	Displays BGP EVI information.
show forwarding distribution peer-id	Displays forwarding information.
show forwarding nve l2 ingress-replication-peers	Displays forwarding information for ingress replication.
show forwarding nve l3 peers	Displays nv3 Layer 3 peers information.
show forwarding ecmp platform	Displays forwarding ECMP platform information.
show forwarding ecmp platform	Displays forwarding ECMP platform information.
show forwarding nve l3 ecmp	Displays forwarding NVE Layer 3 ECMP information.

Example of the **show running-config interface nve 1**

Command

```
switch# show running-config interface nve 1
interface nve1
  no shutdown
  source-interface loopback1 anycast loopback2
  host-reachability protocol bgp
  member vni 10011
    ingress-replication protocol bgp
  member vni 20011 associate-vrf
```

Example of the **show nve interface 1 detail**

Command

```
switch# show nve interface nve 1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [notified]
 Local Router MAC: a093.51cf.78f7
 Host Learning Mode: Control-Plane
 Source-Interface: loopback1 (primary: 30:3:1::2)
 Anycast-Interface: loopback2 (secondary: 303:304::1)
 Source Interface State: Up
 Anycast Interface State: Up
 Virtual RMAC Advertisement: Yes
 NVE Flags:
 Interface Handle: 0x49000001
 Source Interface hold-down-time: 745
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: 0600.0000.0001
 Interface state: nve-intf-add-complete
```

Example of the **show nve peers** Command

```
switch# show nve peers
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- ---------------  ----- --------- -------- -----------------
nve1      1:1::1:1          Up    CP        00:44:09   5087.89d4.6bb7
```

Up

Example of the **show nve vni ingress-replication**

Command

```
switch# show nve vni ingress-replication
Interface VNI      Replication List  Source  Up Time
--------- -------- ----------------- ------- -------

nve1      10011    1:1::1:1           BGP-IMET   00:46:55
```

Example of the **show nve peers** *ipv6-address* **int nv1 counters** Command .

```
switch# show nve peers 2018:2015::abcd:1234:3 int nve 1 counters
Peer IP: 2018:1015::abcd:1234:3
TX
        0 unicast packets 0 unicast bytes
        0 multicast packets 0 multicast bytes
RX
        0 unicast packets 0 unicast bytes
        0 multicast packets 0 multicast bytes
```

Example of the **show bgp l2vpn evpn** Command for Route-Type 2.

```
switch# show bgp l2vpn evpn 1012.0383.9600
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 30.3.1.1:34067    (L2VNI 2001300)
BGP routing table entry for [2]:[0]:[0]:[48]:[1012.0383.9600]:[0]:[0.0.0.0]/216, version
1051240
Paths: (1 available, best #1)
Flags: (0x000102) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP
```

```
  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    303:304::1 (metric 0) from 0:: (30.3.1.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 2001300
      Extcommunity: RT:2:2001300 ENCAP:8

  Path-id 1 advertised to peers:
    2::21              2::66
BGP routing table entry for [2]:[0]:[0]:[48]:[1012.0383.9600]:[32]:[4.231.115.2]/272, version
 1053100
Paths: (1 available, best #1)
Flags: (0x000102) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    303:304::1 (metric 0) from 0:: (30.3.1.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 2001300 3003901
      Extcommunity: RT:2:2001300 RT:2:3003901 ENCAP:8 Router MAC:0600.0000.0001

  Path-id 1 advertised to peers:
    2::21              2::66
```

Example of the **show bgp l2vpn evpn** Command for Route-Type 3

```
switch# show bgp l2vpn evpn 303:304::1
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 30.3.1.1:32769    (L2VNI 2000002)
BGP routing table entry for [3]:[0]:[128]:[303:304::1]/184, version 1045060
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    303:304::1 (metric 0) from 0:: (30.3.1.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: RT:2:2000002 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 2000002, Tunnel Id: 303:304::1

  Path-id 1 advertised to peers:
    2::21              2::66
```

Example of the **show bgp l2vpn evpn** Command for Route-Type 5

```
switch# show bgp l2vpn evpn 5.116.204.0
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 2.0.0.52:302
BGP routing table entry for [5]:[0]:[0]:[24]:[5.116.204.0]/224, version 119983
Paths: (2 available, best #2)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: iBGP

  Path type: internal, path is valid, not best reason: Neighbor Address, no labeled nexthop

  Gateway IP: 0.0.0.0
```

```
  AS-Path: 65001 5300 , path sourced external to AS
    3::52 (metric 200) from 2::66 (2.0.0.66)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 3003301
      Extcommunity: RT:2:3003301 ENCAP:8 Router MAC:f80b.cb53.4897
      Originator: 2.0.0.52 Cluster list: 2.0.0.66

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 2 destination(s)
             Imported paths list: evpn-tenant-0301 default
  Gateway IP: 0.0.0.0
  AS-Path: 65001 5300 , path sourced external to AS
    3::52 (metric 200) from 2::21 (2.0.0.21)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 3003301
      Extcommunity: RT:2:3003301 ENCAP:8 Router MAC:f80b.cb53.4897
      Originator: 2.0.0.52 Cluster list: 2.0.0.21

  Path-id 1 not advertised to any peer
```

Example of the **show l2route peerid** Command

```
switch# show l2route peerid
NVE Ifhdl    IP Address                                PeerID       Ifindex      Num of MAC's
 Num of NH's
------------ ---------------------------------------- ------------ ------------ ------------
 ------------
1224736769   4999:1::1:1:1                             4            1191182340   23377
  0
```

Example of the **show l2route topology detail** Command

```
switch# show l2route topology detail
Flags:(L2cp)=L2 Ctrl Plane; (Dp)=Data Plane; (Imet)=Data Plane BGP IMET; (L3cp)=L3 Ctrl
Plane; (Bfd)=BFD over Vxlan; (Bgp)=BGP EVPN; (Of)=Open Flow mode; (Mix)=Open Flow IR mixed
 mode; (Acst)=Anycast GW on spine;
Topology ID   Topology Name   Attributes
-----------   -------------   ----------
101           Vxlan-10101     VNI: 10101
                              Encap:1 IOD:0 IfHdl:1224736769
                              VTEP IP: 5001:1::1:1:7
                              Emulated IP: ::
                              Emulated RO IP: 0.0.0.0
                              TX-ID: 2004 (Rcvd Ack: 0)
                              RMAC: 00fe.c83e.84a7, VRFID: 3
                              VMAC: 00fe.c83e.84a7
                              VMAC RO: 0000.0000.0000
                              Flags: L3cp, Sub_Flags: --, Prev_Flags: -
```

Example of the **show l2route evpn imet all detail** Command

```
switch# show l2route evpn imet all detail
Flags- (F): Originated From Fabric, (W): Originated from WAN

Topology ID  VNI   Prod  IP Addr        Eth Tag PMSI-Flags Flags   Type Label(VNI)  Tunnel
 ID     NFN Bitmap
-----------  -----  -----  --------------  -------  ----------  -------  ----  -----------
-------------  ----------
901          10901 BGP   4999:1::1:1:1  0       0           -       6    10901
4999:1::1:1:1
```

Example of the **show l2route fl all** Command

```
switch# show l2route fl all
Topology ID Peer-id     Flood List                                Service Node
----------- ----------- ----------------------------------------- ------------
901         4           4999:1::1:1:1                             no
```

Example of the **show l2route mac all detail** Command

```
switch# show l2route mac all detail

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv (AD):Auto-Delete (D):Del Pending
(S):Stale (C):Clear, (Ps):Peer Sync (O):Re-Originated (Nho):NH-Override
(Pf):Permanently-Frozen, (Orp): Orphan

Topology    Mac Address    Prod   Flags         Seq No     Next-Hops

----------- -------------- ------ ------------- ----------
-------------------------------------
901         0016.0901.0001 BGP    SplRcv        0          6002:1::1:1:1

            Route Resolution Type: Regular
            Forwarding State: Resolved (PeerID: 2)
            Sent To: L2FM
          Encap: 1
```

Example of the **show l2route mac-ip all detail** Command

```
switch# show l2route mac-ip all detail
 Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv(D):Del Pending (S):Stale (C):Clear
(Ps):Peer Sync (Ro):Re-Originated (Orp):Orphan
Topology    Mac Address    Host IP                                   Prod   Flags         Seq
 No     Next-Hops
----------- -------------- ----------------------------------------- ------ -----------
---------- ---------------
901         0016.0901.0001 46.1.1.101                                 BGP    --            0
        6002:1::1:1:1
            Sent To: ARP
            encap-type:1
```

Example of the **show ip route 1.191.1.0 vrf vxlan-10101** Command

```
switch# show ip route 1.191.1.0 vrf vxlan-10101
IP Route Table for VRF "vxlan-10101"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

1.191.1.0/29, ubest/mbest: 6/0
    *via fe80::2fe:c8ff:fe09:8fff%default, Po1001, [200/0], 00:56:21, bgp-4002, internal,
tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:1, underlay_vrf: 1) encap: VXLAN

*via fe80::2fe:c8ff:fe09:8fff%default, Po1002, [200/0], 00:56:21, bgp-4002, internal, tag
4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:1, underlay_vrf: 1) encap: VXLAN

    *via fe80::2fe:c8ff:fe09:8fff%default, Po1001, [200/0], 00:56:32, bgp-4002, internal,
tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:2, underlay_vrf: 1) encap: VXLAN
```

```
    *via fe80::2fe:c8ff:fe09:8fff%default, Po1002, [200/0], 00:56:32, bgp-4002, internal,
tag 4007 (evpn)
segid: 10101 VTEP:(5001:1::1:1:2, underlay_vrf: 1) encap: VXLAN
```

Example of the **show forwarding ipv4 route 1.191.1.0 detail vrf vxlan-10101** Command

```
switch# show forwarding ipv4 route 1.191.1.0 detail vrf vxlan-10101

slot  1
=======
Prefix 1.191.1.0/29, No of paths: 2, Update time: Mon Apr 15 15:38:17 2019

   5001:1::1:1:1      nve1
   5001:1::1:1:2      nve1
```

Example of the **show ipv6 route vrf vxlan-10101** Command

```
switch# show ipv6 route vrf vxlan-10101
IPv6 Routing Table for VRF "vxlan-10101"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]

2:2:2::101/128, ubest/mbest: 1/0
    *via 5001:1::1:1:1/128%default, [200/0], 00:55:31, bgp-4002, internal, tag 4002 (evpn)
 segid 10101
VTEP:(5001:1::1:1:1, underlay_vrf: 1) encap: VXLAN
```

Example of the **show forwarding distribution peer-id**

Command

```
switch# show forwarding distribution peer-id
UFDM Peer-id allocations: App id 0
App: VXLAN   Vlan: 1     Id: 4999:1::1:1:1 0x49030001 Peer-id: 0x6
App: VXLAN   Vlan: 1     Id: 5001:1::1:1:1 0x49030001 Peer-id: 0x2
App: VXLAN   Vlan: 1     Id: 5001:1::1:1:2 0x49030001 Peer-id: 0x1
App: VXLAN   Vlan: 1     Id: 5001:1::1:1:7 0x49030001 Peer-id: 0x7
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:101 0x49030001 Peer-id: 0x8
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:102 0x49030001 Peer-id: 0x5
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:103 0x49030001 Peer-id: 0x9
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:104 0x49030001 Peer-id: 0xa
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:105 0x49030001 Peer-id: 0xb
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:106 0x49030001 Peer-id: 0xc
App: VXLAN   Vlan: 1     Id: 5001:1::1:2:107 0x49030001 Peer-id: 0xd
```

Example of the **show forwarding nve l2 ingress-replication-peers**

Command

```
switch# show forwarding nve l2 ingress-replication-peers
slot  1
=======

Total count of VLANS with ingr-repl peers: 1950
VLAN 1024 VNI 0 Vtep Ifindex 0x0 plt_space : 0x1ca75e14
         peer :  6002:1::1:1:1
         peer :  5001:1::1:1:7
         peer :  4999:1::1:1:1

PSS VLAN:1024, VNI:0, vtep:0x0x0, peer_cnt:3
```

```
         peer :  6002:1::1:1:1  marked : 0
         peer :  5001:1::1:1:7  marked : 0
         peer :  4999:1::1:1:1  marked : 0
 VLAN 1280 VNI 0 Vtep Ifindex 0x0 plt_space : 0x1ca75e14
         peer :  6002:1::1:1:1
         peer :  5001:1::1:1:7
         peer :  4999:1::1:1:1

PSS VLAN:1280, VNI:0, vtep:0x0x0, peer_cnt:3
         peer :  6002:1::1:1:1  marked : 0
         peer :  5001:1::1:1:7  marked : 0
         peer :  4999:1::1:1:1  marked : 0
```

Example of the **show forwarding nve l3 peers**

Command

```
switch# show forwarding nve l3 peers
slot  1
=======

EVPN configuration state: disabled, PeerVni Adj enabled
NVE cleanup transaction-id 0
tunnel_id    Peer_id  Peer_address   Interface       rmac         origin state del count
-------------------------------------------------------------------------------------
0x0        1225261062 4999:1::1:1:1        nve1      0600.0001.0001 URIB       merge-done
no    100
0x0        1225261058 5001:1::1:1:1        nve1      2cd0.2d51.9f1b NVE        merge-done
no    100
0x0        1225261057 5001:1::1:1:2        nve1      00a6.cab6.bbbb NVE        merge-done
no    100
0x0        1225261063 5001:1::1:1:7        nve1      00fe.c83e.84a7 URIB       merge-done
no    100
0x0        1225261064 5001:1::1:2:101      nve1      0000.5500.0001 URIB       merge-done
no    100
0x0        1225261061 5001:1::1:2:102      nve1      0000.5500.0002 URIB       merge-done
no    100
0x0        1225261065 5001:1::1:2:103      nve1      0000.5500.0003 URIB       merge-done
no    100
0x0        1225261066 5001:1::1:2:104      nve1      0000.5500.0004 URIB       merge-done
no    100
0x0        1225261067 5001:1::1:2:105      nve1      0000.5500.0005 URIB       merge-done
no    100
```

Example of the **show forwarding ecmp platform**

Command

```
switch# show forwarding ecmp platform
slot  1
=======



ECMP Hash: 0x198b8aae, Num Paths: 2, Hw index: 0x17532
Partial Install: No
Hw ecmp-index: unit-0:1073741827 unit-1:0 unit-2:0, cmn-index: 95538
Hw NVE ecmp-index: unit-0:0 unit-1:0 unit-2:0, cmn-index: 95538
Refcount: 134, Holder: 0x0,   Intf: Ethernet1/101, Nex-Hop: fe80:7::1:2
    Hw adj: unit-0:851977 unit-1:0 unit-2:0, cmn-index: 500010 LIF:4211
  Intf: Ethernet1/108, Nex-Hop: fe80:8::1:2
    Hw adj: unit-0:851978 unit-1:0 unit-2:0, cmn-index: 500012 LIF:4218
   VOBJ count: 0, VxLAN VOBJ count: 0, VxLAN: 0
```

```
ECMP Hash: 0x2bb2905e, Num Paths: 3, Hw index: 0x17533
Partial Install: No
Hw ecmp-index: unit-0:1073741828 unit-1:0 unit-2:0, cmn-index: 95539
Hw NVE ecmp-index: unit-0:0 unit-1:0 unit-2:0, cmn-index: 95539
Refcount: 16, Holder: 0x0,   Intf: Ethernet1/101, Nex-Hop: fe80:7::1:2
    Hw adj: unit-0:851977 unit-1:0 unit-2:0, cmn-index: 500010 LIF:4211
  Intf: Ethernet1/108, Nex-Hop: fe80:8::1:2
    Hw adj: unit-0:851978 unit-1:0 unit-2:0, cmn-index: 500012 LIF:4218
  Intf: port-channel1003, Nex-Hop: fe80:9::1:2
    Hw adj: unit-0:851976 unit-1:0 unit-2:0, cmn-index: 500011 LIF:4106
   VOBJ count: 0, VxLAN VOBJ count: 0, VxLAN: 0
```

Example of the **show forwarding ecmp recursive**

Command

```
switch# show forwarding ecmp recursive
slot  1
=======

Virtual Object 17 (vxlan):
    Hw vobj-index (0): unit-0:851976 unit-1:0 unit-2:0, cmn-index: 99016
    Hw NVE vobj-index (0): unit-0:0 unit-1:0 unit-2:0, cmn-index: 99016
    Hw vobj-index (1): unit-0:0 unit-1:0 unit-2:0, cmn-index: 0
    Hw NVE vobj-index (1): unit-0:0 unit-1:0 unit-2:0 cmn-index: 0
    Num prefixes : 1
Partial Install: No
    Active paths:
        Recursive NH 5001:1::1:2:10a/128 , table 0x80000001
    CNHs:
        fe80:9::1:2, port-channel1003
        Hw adj: unit-0:851976 unit-1:0 unit-2:0, cmn-index: 500011, LIF:4106
        Hw NVE adj: unit-0:0 unit-1:0 unit-2:0, cmn-index: 500011, LIF:4106
    Hw instance new : (0x182c8, 99016) ls count new 1
    FEC: fec_type 0
        VOBJ Refcount : 1
Virtual Object 167 (vxlan): ECMP-idx1:0x17536(95542), ECMP-idx2:0x0(0),
    Hw vobj-index (0): unit-0:1073741832 unit-1:0 unit-2:0, cmn-index: 99166
    Hw NVE vobj-index (0): unit-0:3 unit-1:0 unit-2:0, cmn-index: 99166
    Hw vobj-index (1): unit-0:0 unit-1:0 unit-2:0, cmn-index: 0
    Hw NVE vobj-index (1): unit-0:0 unit-1:0 unit-2:0 cmn-index: 0
    Num prefixes : 1
Partial Install: No
    Active paths:
        Recursive NH 5001:1::1:3:125/128 , table 0x80000001
    CNHs:
        fe80:7::1:2, Ethernet1/101
        Hw adj: unit-0:851977 unit-1:0 unit-2:0, cmn-index: 500010, LIF:4211
        Hw NVE adj: unit-0:0 unit-1:0 unit-2:0, cmn-index: 500010, LIF:4211
        fe80:8::1:2, Ethernet1/108
        Hw adj: unit-0:851978 unit-1:0 unit-2:0, cmn-index: 500012, LIF:4218
        Hw NVE adj: unit-0:0 unit-1:0 unit-2:0, cmn-index: 500012, LIF:4218
    Hw instance new : (0x1835e, 99166) ls count new 2
    FEC: fec_type 0
        VOBJ Refcount : 1
```

Example of the **show forwarding nve l3 ecmp**

Command

```
switch# show forwarding nve l3 ecmp
slot  1
=======


ECMP Hash: 0x70a50e4, Num Paths: 2, Hw Index: 0x17534
table_id: 403, flags: 0x0, adj_flags: 0x0, Ref-ct: 101
  tunnel_id: 5001:1::1:1:1, segment_id: 10101
  tunnel_id: 5001:1::1:1:2, segment_id: 10101
Hw ecmp-index: unit0: 1073741830 unit1: 0 unit2: 0

ECMP Hash: 0x1189f35e, Num Paths: 2, Hw Index: 0x17535
table_id: -2147483245, flags: 0x0, adj_flags: 0x0, Ref-ct: 50
  tunnel_id: 5001:1::1:1:1, segment_id: 10101
  tunnel_id: 5001:1::1:1:2, segment_id: 10101
Hw ecmp-index: unit0: 1073741831 unit1: 0 unit2: 0
```

Verifying VXLAN EVPN and TRM with IPv6 Multicast Underlay

Use the following show command to verify the status of the IPv6 Multicast Underlay configuration:

```
switch(config)# show run interface nve 1

!Command: show running-config interface nve1
!Running configuration last done at: Wed Jul  5 10:03:58 2023
!Time: Wed Jul  5 10:04:01 2023
version 10.3(99x) Bios:version 01.08

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback1
  member vni 10501
    mcast-group ff04::40
  member vni 50001 associate-vrf
    mcast-group ff10:0:0:1::1
```

Use the following show commands to verify the PIMv6 ASM configuration:

```
switch(config)# show ipv6 mroute
IPv6 Multicast Routing Table for VRF "default"

(*, ff04::40/128), uptime: 05:20:19, nve pim6 ipv6
  Incoming interface: Ethernet1/36, RPF nbr: fe80::23a:9cff:fe23:8367
  Outgoing interface list: (count: 1)
    nve1, uptime: 05:20:19, nve


(172:172:16:1::1/128, ff04::40/128), uptime: 05:20:19, nve m6rib pim6 ipv6
  Incoming interface: loopback1, RPF nbr: 172:172:16:1::1
  Outgoing interface list: (count: 2)
    Ethernet1/36, uptime: 01:47:03, pim6
    Ethernet1/27, uptime: 04:14:20, pim6


(*, ff10:0:0:1::10/128), uptime: 05:20:18, nve ipv6 pim6
  Incoming interface: Ethernet1/36, RPF nbr: fe80::23a:9cff:fe23:8367
  Outgoing interface list: (count: 1)
    nve1, uptime: 05:20:18, nve
```

```
(172:172:16:1::1/128, ff10:0:0:1::10/128), uptime: 05:20:18, nve m6rib ipv6 pim6
  Incoming interface: loopback1, RPF nbr: 172:172:16:1::1
  Outgoing interface list: (count: 2)
    Ethernet1/36, uptime: 04:04:35, pim6
    Ethernet1/27, uptime: 04:13:35, pim6
```

```
switch(config)# show ipv6 pim neighbor
PIM Neighbor Status for VRF "default"
Neighbor                  Interface            Uptime    Expires   DR       Bidir-  BFD
     ECMP Redirect
                                                                    Priority Capable State
   Capable
fe80::23a:9cff:fe28:5e07     Ethernet1/27         20:23:38  00:01:44  1        yes     n/a
     no
   Secondary addresses:
    27:50:1:1::2
```

```
switch(config)# show ipv6 pim rp
PIM RP Status Information for VRF "default"
BSR disabled
BSR RP Candidate policy: None
BSR RP policy: None

RP: 101:101:101:101::101, (0),
 uptime: 21:30:43   priority: 255,
 RP-source: (local),
 group ranges:
 ff00::/8
```

The following example provides the output for leaf switch BGP neighbor-1:

```
switch(config-if)# show ipv6 bgp neighbors

BGP neighbor is 33:52:1:1::2, remote AS 200, ebgp link, Peer index 3
  BGP version 4, remote router ID 172.17.1.1
  Neighbor previous state = OpenConfirm
  BGP state = Established, up for 00:00:16
  Neighbor vrf: default
  Peer is directly attached, interface Ethernet1/33
  Enable logging neighbor events
  Last read 0.926823, hold time = 3, keepalive interval is 1 seconds
  Last written 0.926319, keepalive timer expiry due 0.073338
  Received 23 messages, 0 notifications, 0 bytes in queue
  Sent 67 messages, 0 notifications, 0(0) bytes in queue
  Enhanced error processing: On
    0 discarded attributes
  Connections established 1, dropped 0
  Last update recd 00:00:15, Last update sent  = 00:00:15
   Last reset by us 00:08:45, due to session closed
  Last error length sent: 0
  Reset error value sent: 0
  Reset error sent major: 104 minor: 0
  Notification data sent:
  Last reset by peer never, due to No error
  Last error length received: 0
  Reset error value received 0
  Reset error received major: 0 minor: 0
  Notification data received:

  Neighbor capabilities:
  Dynamic capability: advertised (mp, refresh, gr) received (mp, refresh, gr)
  Dynamic capability (old): advertised received
  Route refresh capability (new): advertised received
  Route refresh capability (old): advertised received
  4-Byte AS capability: advertised received
```

```
  Address family IPv6 Unicast: advertised received
  Graceful Restart capability: advertised received

  Graceful Restart Parameters:
  Address families advertised to peer:
    IPv6 Unicast
  Address families received from peer:
    IPv6 Unicast
  Forwarding state preserved by peer for:
  Restart time advertised to peer: 400 seconds
  Stale time for routes advertised by peer: 300 seconds
  Restart time advertised by peer: 120 seconds
  Extended Next Hop Encoding Capability: advertised received
  Receive IPv6 next hop encoding Capability for AF:
    IPv4 Unicast  VPNv4 Unicast

  Message statistics:
                              Sent               Rcvd
  Opens:                        46                  1
  Notifications:                 0                  0
  Updates:                       2                  2
  Keepalives:                   18                 18
  Route Refresh:                 0                  0
  Capability:                    2                  2
  Total:                        67                 23
  Total bytes:                 521                538
  Bytes in queue:                0                  0

  For address family: IPv6 Unicast
  BGP table version 10, neighbor version 10
  3 accepted prefixes (3 paths), consuming 864 bytes of memory
  0 received prefixes treated as withdrawn
  2 sent prefixes (2 paths)
  Inbound soft reconfiguration allowed(always)
  Allow my ASN 3 times
  Last End-of-RIB received 00:00:01 after session start
  Last End-of-RIB sent 00:00:01 after session start
  First convergence 00:00:01 after session start with 2 routes sent

  Local host: 33:52:1:1::1, Local port: 179
  Foreign host: 33:52:1:1::2, Foreign port: 17226
  fd = 112
```

The following example provides the output for leaf switch BGP neighbor-2:

```
switch(config-if)# show bgp l2vpn evpn neighbors 172:17:1:1::1

BGP neighbor is 172:17:1:1::1, remote AS 200, ebgp link, Peer index 5
  BGP version 4, remote router ID 172.17.1.1
  Neighbor previous state = OpenConfirm
  BGP state = Established, up for 00:01:33
  Neighbor vrf: default
  Using loopback0 as update source for this peer
  Using iod 65 (loopback0) as update source
  Enable logging neighbor events
  External BGP peer might be up to 5 hops away
  Last read 0.933565, hold time = 3, keepalive interval is 1 seconds
  Last written 0.915927, keepalive timer expiry due 0.083742
  Received 105 messages, 0 notifications, 0 bytes in queue
  Sent 105 messages, 0 notifications, 0(0) bytes in queue
  Enhanced error processing: On
    0 discarded attributes
  Connections established 1, dropped 0
  Last update recd 00:01:32, Last update sent  = 00:01:32
   Last reset by us never, due to No error
```

```
Last error length sent: 0
Reset error value sent: 0
Reset error sent major: 0 minor: 0
Notification data sent:
Last reset by peer never, due to No error
Last error length received: 0
Reset error value received 0
Reset error received major: 0 minor: 0
Notification data received:

Neighbor capabilities:
Dynamic capability: advertised (mp, refresh, gr) received (mp, refresh, gr)
Dynamic capability (old): advertised received
Route refresh capability (new): advertised received
Route refresh capability (old): advertised received
4-Byte AS capability: advertised received
Address family IPv4 MVPN: advertised received
Address family IPv6 MVPN: advertised received
Address family L2VPN EVPN: advertised received
Graceful Restart capability: advertised received

Graceful Restart Parameters:
Address families advertised to peer:
  IPv4 MVPN  IPv6 MVPN  L2VPN EVPN
Address families received from peer:
  IPv4 MVPN  IPv6 MVPN  L2VPN EVPN
Forwarding state preserved by peer for:
Restart time advertised to peer: 400 seconds
Stale time for routes advertised by peer: 300 seconds
Restart time advertised by peer: 120 seconds
Extended Next Hop Encoding Capability: advertised received
Receive IPv6 next hop encoding Capability for AF:
  IPv4 Unicast  VPNv4 Unicast

Message statistics:
                          Sent               Rcvd
Opens:                       1                  1
Notifications:               0                  0
Updates:                     6                  3
Keepalives:                 95                 95
Route Refresh:               0                  0
Capability:                  6                  6
Total:                     105                105
Total bytes:              2551               2047
Bytes in queue:              0                  0

For address family: IPv4 MVPN
BGP table version 3, neighbor version 3
0 accepted prefixes (0 paths), consuming 0 bytes of memory
0 received prefixes treated as withdrawn
0 sent prefixes (0 paths)
Community attribute sent to this neighbor
Extended community attribute sent to this neighbor
Allow my ASN 3 times
Outbound route-map configured is RN_NextHop_Unchanged, handle obtained
Last End-of-RIB received 00:00:01 after session start
Last End-of-RIB sent 00:00:01 after session start
First convergence 00:00:01 after session start with 0 routes sent

For address family: IPv6 MVPN
BGP table version 3, neighbor version 3
0 accepted prefixes (0 paths), consuming 0 bytes of memory
0 received prefixes treated as withdrawn
0 sent prefixes (0 paths)
```

```
Community attribute sent to this neighbor
Extended community attribute sent to this neighbor
Allow my ASN 3 times
Outbound route-map configured is RN_NextHop_Unchanged, handle obtained
Last End-of-RIB received 00:00:01 after session start
Last End-of-RIB sent 00:00:01 after session start
First convergence 00:00:01 after session start with 0 routes sent

For address family: L2VPN EVPN
BGP table version 7, neighbor version 7
0 accepted prefixes (0 paths), consuming 0 bytes of memory
0 received prefixes treated as withdrawn
4 sent prefixes (4 paths)
Community attribute sent to this neighbor
Extended community attribute sent to this neighbor
Allow my ASN 3 times
Advertise GW IP is enabled
Outbound route-map configured is RN_NextHop_Unchanged, handle obtained
Last End-of-RIB received 00:00:01 after session start
Last End-of-RIB sent 00:00:01 after session start
First convergence 00:00:01 after session start with 4 routes sent

Local host: 172:16:1:2::1, Local port: 21132
Foreign host: 172:17:1:1::1, Foreign port: 179
fd = 113
```