Support for BGP VPNv4, VPNv6, and EVPN Prefixes with IPv6 Next-Hop

Information About Support for BGP VPNv4, VPNv6, and EVPN Prefixes with IPv6 Next-Hop

Previously, BGP IPv4 or L3VPN standards only had provisions for advertising prefixes with next-hop's AFI/SAFI same as that of prefix's AFI/SAFI in NLRI. The standards were:

- IPv4 prefix NLRI can only include IPv4 next hop
- VPNv4 prefix NLRI can only include RD:IPv4 next hop

From Cisco IOS XE Release 17.8, BGP supports RFC 8950 that allows advertising of VPNv4 prefix with IPv6 next-hop NLRI.

EVPN does not have same restrictions, and it supports advertising of any EVPN prefix with either IPv4 or IPv6 next hop from Cisco IOS XE Release 17.8 onwards.

VPNv4/EVPN prefixes with IPV6 next hops and VPNv6 prefixes with non-IPV4-Mapped-IPv6 next hops are not consumed by the BGP peers. It will be either reflected to an iBGP peer or advertised to an ASBR.

Note Previously, VPNv4 or VPN prefixes were expected to have an MPLS label. If it is IPv6 next hop, even if the MPLS label is not valid, the prefixes are accepted and reflected.

BGP Route Reflector/ASBR Support for IPv6 underlay

Support for VPNv4 with IPv6 next hop is as follows:

- Route-target-filtering is disabled only if it needs to be advertised to an ASBR.
- BGP sends the Extended Next Hop Encoding (ENHE) capability for VPNv4 address-family.
- BGP does not import the VPNv4 prefixes into VRF even if there are matching VRF route-targets.
- BGP advertises the remote VPNv4 prefixes with IPv6 next hop to the RR clients which have sent the EHNE capability.

Support for VPNv6 prefixes with non-IPV4-mapped-IPv6 next hop is as follows:

- Route-target-filtering is disabled only if it needs to be advertised to an ASBR.
- BGP does not import the VPNv6 prefixes into VRF even if there are matching VRF route-targets.
- BGP advertises the remote VPNv6 prefixes with IPv6 next hop to the RR clients.

Support for EVPN with IPv6 next hop is as follows:

- Route-target-filtering is disabled only if it needs to be advertised to an ASBR.
- BGP will not import the EVPN prefixes into VRF/L2RIB even if there are matching stitching VRF route-targets.
- BGP will advertise the EVPN prefixes with V6 next hop to RR clients.

Displaying Information about IPv6 Next Hop

The following table contains the commands that display information related to IPv6 next hop:

Command	Functionality
show bgp vpnv4 unicast all [**detail**] [**prefix**]	Displays the gateway address of VPNv4 prefix
show bgp l2vpn evpn [**route-type**] [**all**] [**prefix**]	BGP EVPN already supports IPv6 gateway for both IPv4 and IPv6 prefixes. This EVPN command displays IPv6 gateway for IPv6 paths.
show ip bgp neighbors	Displays the ENHE capability sent or received.
show bgp ipv4 unicast [**detail**] [**prefix**]	Displays the gateway address of IPv4 prefix.

Example: Displaying BGP Neighbor Connection Parameters

The following example shows BGP connection parameters for the neighbor with the IP address 198.51.100.225.

```
# show bgp vpnv4 unicast all neighbor 198.51.100.225
BGP neighbor is 198.51.100.225, remote AS 1, internal link
BGP version 4, remote router ID 209.165.200.225
BGP state = Established, up for 00:15:24
Last read 00:00:54, last write 00:00:18, hold time is 180, keepalive interval is 60 seconds
Last update received: 00:15:24
Neighbor sessions:
1 active, is not multisession capable (disabled)
Neighbor capabilities:
Route refresh: advertised and received(new)
Four-octets ASN Capability: advertised and received
Address family IPv4 Unicast: advertised and received
Address family VPNv4 Unicast: advertised and received
Address family VPNv6 Unicast: advertised and received
Address family L2VPN Evpn: advertised and received
Enhanced Refresh Capability: advertised and received
Extended Next Hop Encoding Capability:
VPNv4 Unicast: advertised and received
Multisession Capability:
Stateful switchover support enabled: NO for session 1
Message statistics:
InQ depth is 0
OutQ depth is 0
```

In this example output, you can see that VPNv4 Unicast is enabled with the Extended Next Hop Encoding Capability.

Example: Behavior of Route-Map Inbound with Next Hop Set for VPNv4/v6 and EVPN

The following example shows the behavior for route-map inbound set next hop for VPNv4/v6 and EVPN.

```
route-map test
set ip next-hop ..
set ipv6 next-hop ...

router bgp 1
addr vpnv4
neighbor ... route-map test in
```

The **set ipv6 nexthop** rule is ignored. The **set ip next-hop** rule is only applied to VPNv4 prefixes with v4 next hop and *not* applied to VPNv4 prefixes with IPv6 next hop.

```
addr vpnv6
neighbor .... route-map test in
```

The **set ip next-hop** rule is ignored. The **set ipv6 next-hop** rule is only applied to VPNv6 prefixes. The VPNv6 prefixes will only have IPv6 next hops.

```
addr l2vpn evpn
neighbor ... route-map test in
```

The **set ipv6 next-hop** rule is ignored. The **set ip next-hop** rule is only applied to EVPN prefixes with IPv4 next hop and *not* applied to EVPN prefixes with V6 next hop.

Configure Gateway IP

To configure the Gateway IP Address to determine how to reach the specified network prefix, peform these steps:

- Configure Route Map with IPv4 Prefix Lists, on page 4
- Configure Route Map with IPv6 Prefix Lists, on page 4
- Set Up a VRF for IPv4 and IPv6 Address Families with an Export Route Map , on page 5
- Configure BGP and EVPN L2VPN with Gateway IP, on page 6

Configure Route Map with IPv4 Prefix Lists

To configure route map with IPv4 prefix lists, perform these steps:

Step 1 Create the route map entry and enter the route-map configuration mode.

route-map name {permit | deny}*[sequence-number]*

Example:

```
Device(config)# route-map gateway-v4map permit 10
```

Step 2 Match against one or more IP address prefix lists.

match ipv4 address prefix-list name*[name]*

Example:

```
Device(config-route-map)# match ip address prefix-list gateway-v4list
```

Step 3 Populate the gateway IP with the value from the nexthop.

set evpn gateway-ip use-nexthop

Example:

```
Device(config-route-map)# set evpn gateway-ip use-nexthop
```

Step 4 Create a prefix list to match either IP packets or routes.

ip prefix-list *name* [**seq** *number*] { **permit | deny**} prefix [**eq** *length]* | **ge** *length* | [**le** *length*]]

Example:

```
Device(config)# ip prefix-list gateway-v4list seq 5 permit 100.0.0.0/24
```

Configure Route Map with IPv6 Prefix Lists

To configure route map with IPv6 prefix lists, perform these steps:

Step 1 Create the route map entry and enter the route-map configuration mode.

route-map name {permit | deny} *[sequence-number]*

Example:

```
Device(config)# route-map gateway-v4map permit 10
```

Step 2 Match against one or more IP address prefix lists.

match ipv6 address prefix-list name *[name]*

Example:

```
Device(config-route-map)# match ipv6 address prefix-list gateway-v4list
```

Step 3 Populate the gateway IP with the value from the nexthop.

set evpn gateway-ip use-nexthop

Example:

```
Device(config-route-map)# set evpn gateway-ip use-nexthop
```

Step 4 Create a prefix list to match either IP packets or routes.

ip prefix-list *name* [**seq** *number*] { **permit | deny**} prefix [**eq** *length*] | [**ge** length] | [**le** *length]]*

Example:

```
Device(config)# Device(config)# ipv6 prefix-list gateway-v6list seq 5 permit 100::/64
```

Set Up a VRF for IPv4 and IPv6 Address Families with an Export Route Map

To set up a VRF for IPv4 and IPv6 address families with an export route map, perform these steps:

Before you begin

Step 1 Create a VRF routing table.

vrf definition *vrf-name*

Example:

```
Device(config)# vrf definition red
```

Step 2 Create routing and forwarding tables for the VRF instance.

rd *route-distinguisher*

Example:

```
Device(config-vrf)# rd 1:1
```

Step 3 Configure the IPv4 address family.

address-family { **ipv4 | ipv6**}

Example:

```
Device(config-vrf)# address-family ipv4
```

Step 4 Associate an export map with a VPN Routing and Forwarding (VRF) instance.

export map *map-tag*

Example:

```
Device(config-vrf-af)# export map gateway-v4map
```

Step 5 Exit address family configuration mode.

exit

Example:

```
Device(config-vrf-af)# exit
```

Step 6 Configure the IPv6 address family.

address-family { **ipv4** | **ipv6**}

Example:

```
Device(config-vrf)# address-family ipv6
```

Step 7 Associate an export map with a VPN Routing and Forwarding (VRF) instance.

export map *map-tag*

Example:

```
Device(config-vrf-af)# export map gateway-v6map
```

Step 8 Exit address family configuration mode.

exit

Example:

```
Device(config-vrf-af)# exit
```

Configure BGP and EVPN L2VPN with Gateway IP

to configure BGP and EVPN L2VPN with Gateway IP, perform these steps:

Step 1 Configure BGP.

router bgp *number*

Example:

```
Device(config)# router bgp 1000
```

Step 2 Configure the neighbor address to allows BGP sessions to use any operational interface for TCP connections.

neighbor *ip-address* **update-source** *interface-type interface-number*

Example:

```
Device(config-router)# neighbor 2.2.2.2 update-source loopback0
```

Step 3 Configure the neighbor with the remote AS number.

neighbor ip-address remote-as asn

Example:

```
Device(config-router)# neighbor 2.2.2.2 remote-as 2000
```

Step 4 Specify L2VPN address family to enter the address family configuration mode.

address-family l2vpn evpn

Example:

```
Device(config-router)# address-family l2vpn evpn
```

Step 5 Enable the exchange information from a BGP neighbor.

neighbor ip-address activate

Example:

```
Device(config-router-af)# neighbor 2.2.2.2 activate
```

Step 6 Specify the communities attribute sent to a BGP neighbor.

neighbor *ip-address* **send-community extended**

Example:

```
Device(config-router-af)# neighbor 2.2.2.2 send-community extended
```

Step 7 Disable advertisement of gateway IP towards the specified peer.

neighbor ip-address advertise-gw-ip-disable

Example:

```
Device(config-router-af)# neighbor 2.2.2.2 advertise-gw-ip-disable
```

Note Advertise gateway-ip disable option is also accessible through the peer-policy template and peer-group settings.

Step 8 Exit the address family configuration mode.

exit-address-family

Example:

```
Device(config-router-af)# exit-address-family
```

Feature Information for Support for BGP VPNv4, VPNv6, and EVPN Prefixes with IPv6 Next-Hop

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Support for BGP VPNv4, VPNv6, and EVPN Prefixes with IPv6 Next-Hop

Feature Name	Releases	Feature Information
Support for BGP VPNv4, VPNv6, and EVPN Prefixes with IPv6 Next-Hop	17.8.1	This feature allows you to use the Multiprotocol BG (BGP-MP) capability to carry VPNv4 Network Layer Reachability Information (NLRI) in an IPv6 next hop. This helps to reduce the operating cost by carrying both VPNv4 and IPv6 over the same BGP session. VPNv4 or EVPN prefixes with IPv6 next hops and VPNv6 prefixes with non-IPv4-mapped-IPv6 next hops are not supported by the BGP peers. It is either reflected to an iBGP peer or advertised to an ASBR.