



Configure EVPN IRB

This chapter introduces you to Ethernet VPN (EVPN) Integrated Routing and Bridging (IRB) feature and describe how you can configure the EVPN IRB feature.

- [EVPN IRB , on page 1](#)
- [EVPN single-homing access EVPN gateway , on page 3](#)
- [EVPN multi-homing active-active, on page 5](#)
- [EVPN IRB Support, on page 8](#)
- [Distributed Anycast Gateway, on page 8](#)
- [VM Mobility Support, on page 10](#)
- [Configuring EVPN IRB, on page 10](#)
- [Running Configuration for EVPN IRB, on page 12](#)
- [Verify EVPN IRB, on page 13](#)
- [EVPN IPv6 Hosts with Mobility, on page 23](#)
- [EVPN IRB: DHCPv4 and DHCPv6 Relay , on page 34](#)
- [Duplicate IP Address Detection, on page 39](#)
- [EVPN E-Tree Using RT Constraints, on page 41](#)

EVPN IRB

Ethernet Virtual Private Network Integrated Routing and Bridging (EVPN IRB) is a category of network technology that provides an extensible and flexible solution for Layer 2 (L2) and Layer 3 (L3) connectivity among hosts across different IP subnets over an MPLS core and IP network.

The EVPN IRB feature is a component that

- enables L3 forwarding among hosts across different IP subnets,
- while maintaining the multi-homing capabilities of EVPN.
- Additionally, it allows EVPN hosts or subnets to communicate with IP VPNs, enhancing network flexibility and connectivity.

EVPN IRB components

To implement EVPN IRB, the network uses these key components to ensure effective traffic management and routing:

- **BGP (Border Gateway Protocol)**: It advertises subnet and host routes to the EVPN core using route-type 5 and route-type 2 messages.
- **EVPN**: It manages ethernet segment configurations and handles host route advertisements and failover scenarios.
- **L2RIB (Layer 2 Routing Information Base)**: It handles MAC or IP mobility, resolves routes, and computes the best routes.
- **BVI MA (Bridge Virtual Interface Manager)**: It manages IRB interfaces and supports routing by advertising the BVI's subnet and MAC addresses.
- **L2FIB (Layer 2 Forwarding Information Base)**: It ensures correct forwarding based on MAC and IP addresses.

EVPN IRB environments

The EVPN IRB supports these environments:

- **Single-homing interface**: Customer Edge (CE) devices connect directly to a single Physical Edge (PE) router.
- **Multi-homing interface**: CE device connect to multiple PE routers through dual links, a Link Aggregation Group (LAG), or a switch.
- **Anycast gateway or Bridge Virtual Interface (BVI)**: BVI interfaces use the same IP and MAC addresses on all PE routers, enabling devices in the network to reach the gateway using the same address, regardless of which router is the designated forwarder.

EVPN-IRB core and access options

EVPN-IRB supports these core and access options:

- **Core options**:
 - **Host routing**: Enable or disable host routing to manage subnet and host route advertisements.
 - **VRF**: Integrate with VRF for seamless route management.
- **Access options**:
 - **Switch devices**: Supports switch devices with or without LAG.
 - **Y-cable configurations**: Use Y-cable configurations for redundancy.
 - **Mixed environments connections**: Operate in mixed environments with single-homing, single-active multi-homing, and all-active multi-homing connections.
 - **Single-Flow-Active (SFA) load-balancing mode**: Implement SFA for load balancing.

Limitations and restrictions for EVPN IRB

These are the limitations and restrictions for EVPN IRB:

- Traffic from the Bridge Virtual Interface (BVI) does not use the BVI's configured MAC address but instead uses the MAC address of the physical interface, devices in the network are unable to properly learn the BVI's MAC address. This results in unnecessary traffic flooding as devices repeatedly attempt to resolve the MAC address associated with the BVI IP. To address this, implement a configurable timer under the BVI interface to periodically broadcast the BVI's MAC and IP pairing using Gratuitous ARP (GARP) for IPv4 or unsolicited Neighbor Advertisement for IPv6.
- You must enable host routing when configuring EVPN-IRB.

EVPN single-homing access EVPN gateway

EVPN single-homing access EVPN gateway is a network architecture that leverages EVPN technology to provide L2 and L3 VPN services. This setup is particularly useful for connecting CE devices to a service provider's network in a single-homed configuration by connecting each CE device to only one PE device.

How EVPN single-homing access EVPN gateway works

Summary

EVPN IRB single-homing workflow includes these key components:

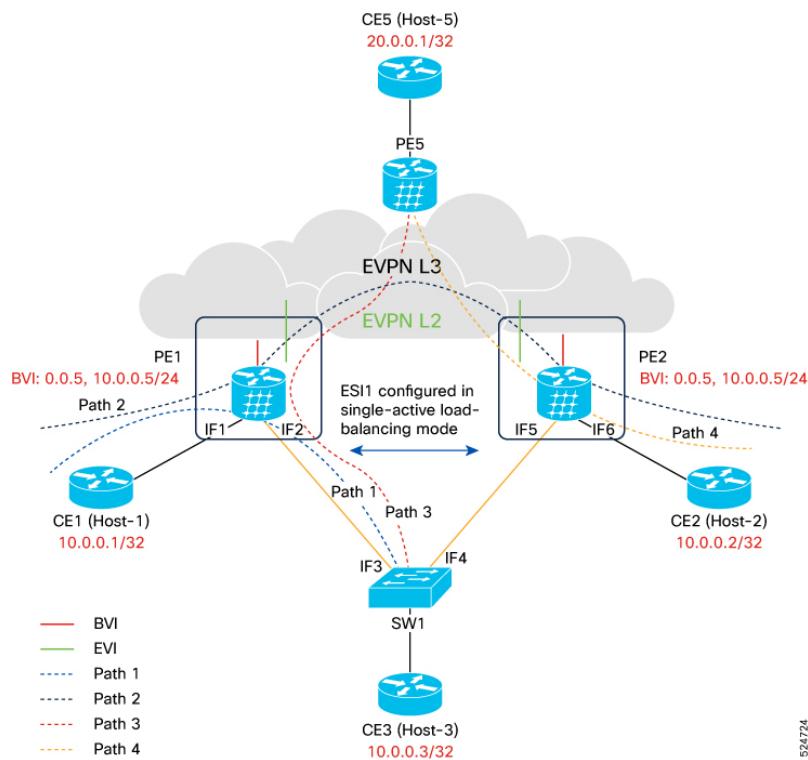
- **CE or host routers:**
 - CE1 or Host-1 with IP address 10.0.0.1/32
 - CE2 or Host-2 with IP address 10.0.0.2/32
 - CE3 or Host-3 with IP address 10.0.0.3/32
 - CE5 or Host-5 with IP address 20.0.0.1/32
- **PE routers:**
 - PE1 with BVI IP address 10.0.0.5/24 and BVI mac address 0.0.5
 - PE2 with BVI IP address 10.0.0.5/24 and BVI mac address 0.0.5
 - PE5
- **Interfaces:**
 - IF1: Connects Host-1 to PE1
 - IF2: Connects Host-3 to PE1
 - IF3: Connects PE1 to Host-3
 - IF4: Connects PE2 to Host-3
 - IF5: Connects Host-3 to PE2
- **Paths:**
 - Path 1: Host-1 communicating with Host-2 using a EVPN single-homing access EVPN gateway interface

How EVPN single-homing access EVPN gateway works

- Path 2: Host-1 communicating with Host-3 using a EVPN single-homing access EVPN gateway interface
- Path 3: Host-3 communicating with Host-5 using a EVPN single-active multihoming interface
- Path 4: Host-2 communicating with Host-5 using a EVPN multihoming active-active interface
- Switch device: SW1, connects Host-3 to PE1 and PE2
- Host routing is enabled on PE1 and PE2.
- IRB interfaces are configured as anycast.
- Host-1 and Host-2 connect to PE1 and PE2 using a EVPN single-homing interfaces.
- Host-3 connect to PE1 and PE2 using a EVPN multihoming interfaces.
- IOS XR Software performs the Designated Forwarder (DF) election for the shared Ethernet Segment Identifier (ESI) and elects IF2 as DF and IF5 as Non-Designated Forwarder (NDF).
- Interface IF5 is set to blocked state which blocks both BUM and unicast traffic.

Workflow

Figure 1: EVPN IRB single-homing network topology



Let's consider these scenarios:

- Where Host-1 wants to communicate with Host-2, which are in same subnet.
1. Host-1 sends an ARP request, which is received by the Bridge Domain (BD) on PE1.

2. PE1 learns the Host-1 MAC and IP addresses from these ARP packets and replicates this information across all output interfaces.
 3. PE1 advertises Host-1 route using EVPN route type-2 to remote PEs. Remote PEs, such as PE2, import and install this route as a remote route.
 4. Since Host-2 is directly connected to PE2, it receives the ARP request and responds with a unicast ARP response.
 5. The ARP process ensures that PE2 learns Host-2 IP address 10.0.0.2/32.
 6. PE2 performs a MAC lookup for Host-1 and forwards packet to peering PE1, enabling communication between Host-2 and Host-1.
- Where Host-1 wants to communicate with Host-3, which are in same subnet.
 1. Host-1 sends an ARP request, which is received by the BD on PE1.
 2. PE1 learns the Host-1 MAC and IP addresses from these ARP packets and replicates this information across all output interfaces.
 3. Since Host-3 is directly connected to PE1, it receives the ARP request and responds with a unicast ARP response.
 4. The ARP process ensures that PE1 learns Host-3 IP address 10.0.0.3/32, enabling communication between Host-1 and Host-3.

EVPN multi-homing active-active

EVPN multi-homing access gateway enables redundant network connectivity by allowing a CE device to connect to more than one PE devices. Disruptions to the network connectivity are prevented by allowing a CE device to be connected to a PE device or several PE devices through multi-homing. Ethernet segment is the bunch of Ethernet links through which a CE device is connected to more than one PE devices. The Multi-chassis Link Aggregation Group (MC-LAG) bundle operates as an Ethernet segment. In Release 6.2.1, only MC bundles crossing two chassis are supported.

In EVPN IRB, both EVPN and IP VPN (both VPNv4 and VPNv6) address families are enabled between ASR 9000 Data Center Interconnect (DCI) gateways. When Layer 2 (L2) stretch is not available in multiple datacenters (DC), routing is established through VPNv4 or VPNv6 routes. When Layer 2 stretch is available, host routing is applied where IP-MAC routes are learnt by ARP/IPv6 ND and are distributed to EVPN/BGP. In remote peer gateway, these IP-MAC EVPN routes are imported into IP VPN routing table from EVPN route-type 2 routes with secondary label and Layer 3 VRF route-target.

How EVPN multi-homing active-active works

Summary

EVPN IRB multi-homing active-active workflow includes these key components:

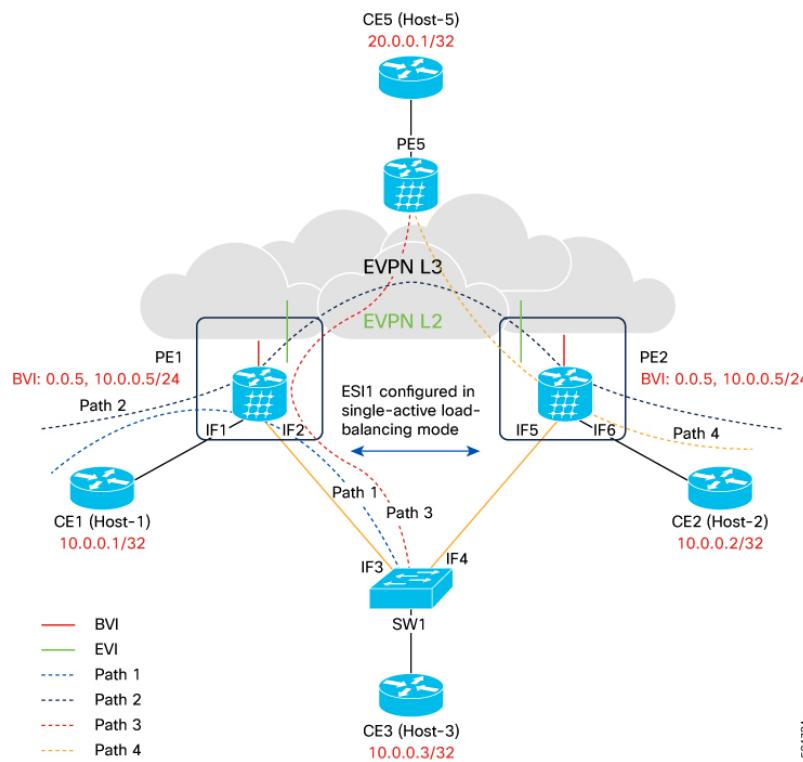
- **CE or host routers:**
 - CE1 or Host-1 with IP address 10.0.0.1/32
 - CE2 or Host-2 with IP address 10.0.0.2/32

How EVPN multi-homing active-active works

- CE3 or Host-3 with IP address 10.0.0.3/32
- CE5 or Host-5 with IP address 20.0.0.1/32
- **PE routers:**
 - PE1 with BVI IP address 10.0.0.5/24 and BVI mac address 0.0.5
 - PE2 with BVI IP address 10.0.0.5/24 and BVI mac address 0.0.5
 - PE5
- **Interfaces:**
 - IF1: Connects Host-1 to PE1
 - IF2: Connects Host-3 to PE1
 - IF3: Connects PE1 to Host-3
 - IF4: Connects PE2 to Host-3
 - IF5: Connects Host-3 to PE2
- **Paths:**
 - Path 1: Host-1 communicating with Host-2 using a EVPN single-homing access EVPN gateway interface
 - Path 2: Host-1 communicating with Host-3 using a EVPN single-homing access EVPN gateway interface
 - Path 3: Host-3 communicating with Host-5 using a EVPN single-active multihoming interface
 - Path 4: Host-2 communicating with Host-5 using a EVPN multihoming active-active interface
- Switch device: SW1, connects Host-3 to PE1 and PE2
- Host routing is enabled on PE1 and PE2.
- IRB interfaces are configured as anycast.
- Host-1 and Host-2 connect to PE1 and PE2 using a EVPN single-homing interfaces.
- Host-3 connect to PE1 and PE2 using a EVPN multihoming interfaces.
- IOS XR Software performs the Designated Forwarder (DF) election for the shared Ethernet Segment Identifier (ESI) and elects IF2 as DF and IF5 as Non-Designated Forwarder (NDF).
- Interface IF5 is set to blocked state which blocks both BUM and unicast traffic.

Workflow

Figure 2: EVPN IRB multi-homing network topology



Let's consider these scenarios:

- Where Host-2 wants to communicate with Host-5, which are in different subnet.
 1. Host-2 send an ARP request to its gateway which is IRB interface. It basically ARPs the BVI IP address 10.0.0.5.
 2. PE2 learn the Host-2 MAC and IP addresses from these ARP packets and replicates this information across all output interfaces where it is added to the BVI interface.
 3. The BVI interface on PE2 sends an ARP response to Host-2 using its BVI IP address 10.0.0.5 and MAC address 0.0.5.
 4. PE2 advertises Host-2 route using EVPN route type-2 to remote PEs. Remote PEs, such as PE5, import and install this route as a remote route.
 5. Since Host-5 is directly connected to PE5, it receives the ARP request and responds with a unicast ARP response.
 6. The ARP process ensures that PE5 learns Host-5 IP address 20.0.0.1/32, enabling communication between Host-2 and Host-5.
 7. If PE2 doesn't have Host-5 specific route, it may use an EVPN route type-5 to forward traffic to PE5, where ARP resolves Host-5, enabling Host-2 and Host-5 to communicate.
- Where Host-5 sends a packet to Host-2. If Host-2 hasn't communicated yet, PE5 might not have Host-2 specific route.

- If PE5 directs traffic to PE1 first, a Generalized Learning (G-LEAN) adjacency process occurs, and traffic is dropped until it is resolved.
- Once PE5 identifies PE2 as the best destination for Host-2, it forwards the packet to PE2, and PE2 performs these steps:
 1. PE2 performs an IP lookup, finding the BVI interface as the destination.
 2. Destination MAC is set to Host-2 MAC as learned by ARP and source MAC remains as the BVI MAC address 0.0.5.
 3. PE2 performs a MAC lookup within the bridge domain and forwards the packet to Host-2.

EVPN IRB Support

EVPN IRB supports the following scenarios:

- In single-homing scenario, only physical, VLAN, .1q, .1ad, or QinQ access methods are supported.
- In dual-homing scenario, only two PE gateways in a redundancy group are supported.
- Both IPv4 and IPv6 are supported.

Distributed Anycast Gateway

EVPN IRB for the given subnet is configured on all the EVPN PEs that are hosted on this subnet. To facilitate optimal routing while supporting transparent virtual machine mobility, hosts are configured with a single default gateway address for their local subnet. That single (anycast) gateway address is configured with a single (anycast) MAC address on all EVPN PE nodes locally supporting that subnet. This process is repeated for each locally defined subnet requires Anycast Gateway support.

The host-to-host Layer 3 traffic, similar to Layer 3 VPN PE-PE forwarding, is routed on the source EVPN PE to the destination EVPN PE next-hop over an IP or MPLS tunnel, where it is routed again to the directly connected host. Such forwarding is also known as Symmetric IRB because the Layer 3 flows are routed at both the source and destination EVPN PEs.

The following solutions are part of the Distributed Anycast Gateway feature:

EVPN IRB with Active-Active Multi-Homing with Subnet Stretch or Host-Routing across the Fabric

For a bridge domain or subnet that is stretched across remote EVPN PEs, both /32 host routes and MAC routes are distributed in a EVPN overlay control plane to enable Layer 2 and Layer 3 traffic to the end points in a stretched subnet.

This type of multi-homing has the following characteristics:

- All-active MC-LAG on access

- Layer 2 or Layer 3 ECMP for the fabric for dual-homed hosts based on Route Type 1 and Route Type 2
- Layer 3 unipath over the Fabric for single-homed hosts based on Route Type 2
- Layer 2 subnet stretch over the fabric
- Layer 2 stretch within redundancy group of leafs with orphan ports

MAC and IP Unicast Control Plane

This use case has following types:

Prefix Routing or No Subnet Stretch

IP reachability across the fabric is established using subnet prefix routes that are advertised using EVPN Route Type 5 with the VPN label and VRF RTs. Host ARP and MAC sync are established across multi-homing EVPN PEs using MAC+IP Route Type 2 based on a shared ESI to enable local switching through both the multi-homing EVPN PEs.

Host Routing or Stretched Subnet

When a host is discovered through ARP, the MAC and IP Route Type 2 is advertised with both MAC VRF and IP VRF router targets, and with VPN labels for both MAC-VRF and IP-VRF. Particularly, the VRF route targets and Layer 3 VPN label are associated with Route Type 2 to achieve PE-PE IP routing identical to traditional L3VPNs. A remote EVPN PE installs IP/32 entries directly in Layer 3 VRF table through the advertising EVPN PE next-hop with the Layer 3 VPN label encapsulation, much like a Layer 3 VPN imposition PE. This approach avoids the need to install separate adjacency rewrites for each remote host in a stretched subnet. Instead, it inherits a key Layer 3 VPN scale benefit of being able to share a common forwarding rewrite or load-balance resource across all IP host entries reachable through a set of EVPN PEs.

ARP and MAC sync

For hosts that are connected through LAG to more than one EVPN PE, the local host ARP and MAC entries are learnt in data plane on either or both of the multihoming EVPN PEs. Local ARP and MAC entries are synced across the two multihoming EVPN PEs using MAC and IP Route Type 2 based on a shared ESI to enable local switching through both the multihoming EVPN PEs. Essentially, a MAC and IP Route Type 2 that is received with a local ESI causes the installation of a synced MAC entry that points to the local AC port, and a synced ARP entry that is installed on the local BVI interface.



Note Only one Ethernet Flow Point (EFP) is supported per non-Zero ESI per bridge domain or EVI. This is a limitation of EVPN.

MAC and IP Route Re-origination

MAC and IP Route Type 2 received with a local ESI, which is used to sync MAC and ARP entries, is also re-originated from the router that installs a SYNC entry, if the host is not locally learnt and advertised based on local learning. This route re-origination is required to establish overlay IP ECMP paths on remote EVPN PEs, and to minimize traffic hit on local AC link failures, that can result in MAC and IP route withdraw in the overlay.

Intra-subnet Unicast Data Plane

The Layer 2 traffic is bridged on the source EVPN PE using ECMP paths to remote EVPN PEs, established through MAC+IP RT2, for every ES and for every EVI, ES and EAD Route Type 2 routes that are advertised from the local EVPN PEs.

Inter-subnet Unicast Data Plane

Inter-subnet traffic is routed on the source EVPN PEs through overlay ECMP to the destination EVPN PEs next-hops. Data packets are encapsulated with the VPN label advertised from the EVPN PE and tunnel label for the BGP next-hop towards the spine. It is then routed again on the destination EVPN PE using a local ARP adjacency towards the host. IP ECMP on the remote EVPN PEs is established through local and re-originated routes advertised from the local EVPN PEs.

VM Mobility Support

VM mobility is the ability of virtual machines to migrate between one server and another while retaining their existing MAC and IP addresses.

The following are the two key components in EVPN Route Type 2 that enable VM Mobility:

- Host MAC advertisement component that is imported into local bridge MAC table, and Layer 2 bridged traffic across the network overlay.
- Host IP advertisement component that is imported into the IP routing table in a symmetric IRB design, enables routed traffic across the network overlay.

The above-mentioned components are advertised together in a single MAC + IP host route advertisement. An additional MAC-only route could also be advertised.

The following behaviors of VM are supported. The VM can:

- retain existing MAC and acquire a new IP address
- retain existing IP address and acquire a new MAC
- retain both existing MAC and IP address



Note IRB solution supports VM mobility with IP+MAC pair. VM mobility move, with new IP to MAC, or new MAC to IP, is not supported.

Configuring EVPN IRB

```
/* Configure CEF to prefer RIB prefixes over adjacency prefixes.*/

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether 3
```

```

RP/0/RSP0/CPU0:router(config-if)# lacp system mac 1.1.1
RP/0/RSP0/CPU0:router(config-if)# exit

RP/0/RSP0/CPU0:router(config)# cef adjacency route override rib

/* Configure EVPN L3VRF per DC tenant. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# vrf irb1
RP/0/RSP0/CPU0:router(config-vrf)# address-family ipv4 unicast
RP/0/RSP0/CPU0:router(config-vrf-af)# import route-target 1000:1
RP/0/RSP0/CPU0:router(config-vrf-af)# export route-target 1000:1
RP/0/RSP0/CPU0:router(config-vrf-af)# exit

/* Configure Layer 2 attachment circuit (AC) from multichassis (MC) bundle interface, and
bridge-group virtual interface (BVI) per bridge domain. */
/* Note: When a VM migrates from one subnet to another (subnet stretching), apply the
following IRB configuration to both the EVPN PEs. *\

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bvi 1001
RP/0/RSP0/CPU0:router(config-if)# host-routing
RP/0/RSP0/CPU0:router(config-if)# vrf irb1
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.10.0.4 255.255.255.0
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.16.0.1 secondary
RP/0/RSP0/CPU0:router(config-if)# mac-address 00aa.1001.00aa

/* Configure EVPN Layer 2 bridging service. Note: This configuration is performed in Layer
2 gateway or bridging scenario. */

Router# configure
Router(config)# l2vpn
Router(config-l2vpn)# bridge group 1
Router(config-l2vpn-bg)# bridge-domain 1-1
Router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/0/0/1.1
Router(config-l2vpn-bg-bd-ac)# evi 1
Router(config-l2vpn-bg-bd-ac-evi)# commit
Router(config-l2vpnbg-bd-ac-evi)# exit

/* Configure BGP. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# router bgp 3107
RP/0/RSP0/CPU0:router(config-bgp)# vrf irb1
RP/0/RSP0/CPU0:router(config-bgp-vrf)# rd auto
RP/0/RSP0/CPU0:router(config-bgp-vrf)# address-family ipv4 unicast
RP/0/RSP0/CPU0:router(config-bgp-vrf-af)# redistribute connected
RP/0/RSP0/CPU0:router(config-bgp-vrf-af)# redistribute static
RP/0/RSP0/CPU0:router(config-bgp-vrf-af)# exit

/* Configure EVPN, and configure main bundle ethernet segment parameters in EVPN. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# evi 2001
RP/0/RSP0/CPU0:router(config-evpn-evi)# bgp
RP/0/RSP0/CPU0:router(config-evpn-evi-bgp)# route-target import 1000:1
RP/0/RSP0/CPU0:router(config-evpn-evi-bgp)# route-target export 1000:1
RP/0/RSP0/CPU0:router(config-evpn-evi-bgp)# exit
RP/0/RSP0/CPU0:router(config-evpn-evi)# advertise-mac
RP/0/RSP0/CPU0:router(config-evpn-evi)# unknown-unicast-suppression

```

```
/* Configure Layer 2 VPN. */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group irb
RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain irb1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface bundle-Ether3.1001
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# routed interface BVI100
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-bvi)# split-horizon group core
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-bvi)# evi 10001
```

Running Configuration for EVPN IRB

```
/* Configure LACP */
interface Bundle-Ether3
  lacp system mac 1.1.1
!

/* Configure CEF adjacency overwrite. */
cef adjacency route override rib

/* Configure EVPN Layer 3 VRF per DC tenant. */
vrf irb1
address-family ipv4 unicast
  import route-target
    1000:1
  !
  export route-target
    1000:1
  !
!
!

/* Configure Layer 2 attachment circuit (AC) from multichassis (MC) bundle interface, and
bridge-group virtual interface (BVI) per bridge domain.*/

interface Bundle-Ether3.1001 l2transport
  encapsulation dot1q 1001
  rewrite ingress tag pop 1 symmetric
!
interface BVI1001
  host-routing
  vrf irb1
  ipv4 address 10.0.1.1 255.255.255.0
  mac-address 0000.3030.1
!

/* Configure BGP. */

router bgp 3107
  vrf irb1
  rd auto
  address-family ipv4 unicast
  redistribute connected
  redistribute static
!
```

```

!
/* Configure EVPN. */

evpn
evi 10001
  bgp
    route-target import 1000:1
    route-target export 1000:1
  !
  advertise-mac
  unknown-unicast-suppression
!

/* Configure Layer2 VPN. */

l2vpn
bridge group irb
  bridge-domain irb1
    interface Bundle-Ether3.1001
    !
    routed interface BVI1001
      split-horizon group core
    !
  evi 10001
!
!
```

Verify EVPN IRB

Verify the Address Resolution Protocol (ARP) protocol entries, and synced entries in multi-homing scenarios.

RP/0/RSP0/CPU0:router# **show arp vrf evpn1**

0/1/CPU0						
Address	Age	Hardware Addr	State	Type	Interface	
10.1.1.1	-	0010.0001.0001	Interface	ARPA	BVI1	
10.1.1.11	02:23:46	1000.0001.0001	Dynamic	ARPA	BVI1	
10.1.1.93	-	0000.f65a.357c	EVPN_SYNC	ARPA	BVI1	
10.1.2.1	-	0011.0112.0001	Interface	ARPA	BVI2	
10.1.2.91	02:24:14	0000.f65a.3570	Dynamic	ARPA	BVI2	
10.1.2.93	02:21:52	0000.f65a.357d	Dynamic	ARPA	BVI2	

0/0/CPU0						
Address	Age	Hardware Addr	State	Type	Interface	
10.1.1.1	-	0010.0001.0001	Interface	ARPA	BVI1	
10.1.1.11	02:23:46	1000.0001.0001	Dynamic	ARPA	BVI1	
10.1.1.93	-	0000.f65a.357c	EVPN_SYNC	ARPA	BVI1	
10.1.2.1	-	0011.0112.0001	Interface	ARPA	BVI2	
10.1.2.91	02:24:14	0000.f65a.3570	Dynamic	ARPA	BVI2	
10.1.2.93	02:21:52	0000.f65a.357d	Dynamic	ARPA	BVI2	

Verify the adjacency entries, particularly verify newly added information for synced IPv4 and IP ARP entries.

Verify EVPN IRB

```
RP/0/RSP0/CPU0:router# show adjacency ipv4 BVI 1 internal detail location 0/0/CPU0

BVI1, 10.1.1.93 (ipv4)
Version: 1169, references: 2, transient lock: 0
Encapsulation information (14 bytes) 0000f65a357c0000f65a357c0800 MTU: 1500
Adjacency pointer is: 0x770a9278
Platform adjacency pointer is: 0x7d7bc380
Last updated: Feb 28 15:58:21.998
Adjacency producer: arp (prod_id: 10)
Flags: incomplete adj,
Additional Adjacency Information (4 bytes long),
Upto first 4 bytes (in hex): 01000000
Netio idb pointer not cached Cached interface type: 78

Adjacency references:
bfd_agent (JID 150, PID 3637), 0 reference
l2fib_mgr (JID 185, PID 4003), 0 reference
fib_mgr (JID 294, PID 3605), 1 reference
aib (JID 314, PID 3590), 1 reference

BVI1, 10.1.1.11 (ipv4) Version: 1493,
references: 3, transient lock: 0
Encapsulation information (14 bytes) 1000000100010010000100010800
MTU: 1500
Adjacency pointer is: 0x770ab778
Platform adjacency pointer is: 0x7d7bc10
Last updated: Mar 2 17:22:00.544
Adjacency producer: arp (prod_id: 10)
Flags: incomplete adj,
Netio idb pointer not cached Cached interface type: 78
Adjacency references:
bfd_agent (JID 150, PID 3637), 0 reference
l2fib_mgr (JID 185, PID 4003), 1 reference
fib_mgr (JID 294, PID 3605), 1 reference
aib (JID 314, PID 3590), 1 reference
```

Verify the entries to obtain details learnt in L2FIB line cards. In multi-homing active-active scenario, the link-local addresses are also updated and distributed to EVPN peer gateways.

```
RP/0/RSP0/CPU0:router# show l2vpn mac-learning mac-ipv4 all location 0/0/cPU0
```

Topo ID	Producer	Next Hop(s)	Mac Address	IP Address
6	0/0/CPU0	BV1	1000.0001.0001	10.1.1.11
7	0/0/CPU0	BV2	0000.f65a.3570	10.1.2.91
7	0/0/CPU0	BV2	0000.f65a.357d	10.1.2.93

```
RP/0/RSP0/CPU0:router# show l2vpn mac-learning mac-ipv4 all location 0/0/cPU0
```

Topo ID	Producer	Next Hop(s)	Mac Address	IP Address
6	0/0/CPU0	BV1	0000.f65a.357c	fe80::200:f6ff:fe5a:357c
7	0/0/CPU0	BV2	0000.f65a.3570	10:1:2::91
7	0/0/CPU0	BV2	0000.f65a.357d	10:1:2::93
7	0/0/CPU0	BV2	0000.f65a.3570	fe80::200:f6ff:fe5a:3570

Verify sequence ID for VM mobility.

```
RP/0/RSP0/CPU0:router# show l2route evpn mac-ip all detail

Sun Apr 30 18:09:19.368 PDT
Flags: (Sst)=Static; (L)=Local; (R)=Remote; (F)=Flood;
(N)=No Redistribution; (Rtr)=Router MAC; (B)=Best Route;
(P)=Probe; (S)=Peer Sync; (F)=Flush;
(D)=Duplicate MAC; (Z)=Frozen MAC;

Topo ID      Mac Address      IP Address    Prod   Next Hop(s)      Seq No  Flags
Opaque Data Type  Opaque Data Len  Opaque Data Value
-----  -----  -----  -----  -----  -----
-----  -----  -----  -----  -----  -----
33          0022.6730.0001  10.130.0.2  L2VPN  Bundle-Ether6.1300  0        SB 0 12
0x060000000          0x220000080          0x000000000

Last Update: Sun Apr 30 15:00:01.911 PDT

33          0022.6730.0002  10.130.0.3  LOCAL  Bundle-Ether6.1300  0        B           N/A
N/A          N/A          N/A          N/A          N/A          N/A          N/A


```

Verify the entries to obtain details learnt in L2FIB RP when it is an aggregator. Route processor (RP) entries are aggregated entries obtained from the line cards. In some cases of MAC move, there could be different states for the same MAC. This is displayed in RP aggregated entries. RP determines the update to be sent to L2RIB according to MAC-Learning algorithms.

```
RP/0/RSP0/CPU0:router# show l2vpn mac-learning mac-ipv4 all location 0/RSP0/CPU0

Topo ID  Producer      Next Hop(s)      Mac Address      IP Address
-----  -----  -----  -----  -----
6        0/0/CPU0      BV1            1000.0001.0001  10.1.1.11
7        0/0/CPU0      BV2            0000.f65a.3570  10.1.2.91
7        0/0/CPU0      BV2            0000.f65a.357d  10.1.2.93
```

Verify the entries in L2RIB that are updated by RP L2FIB. Note the following when you verify the entries:

- The entries with producer as L2VPN and NH as remote IP are learnt from the remote peer gateways, which are learnt from BGP, updated to EVPN, and then updated to L2RIB. So these entries are not from local IP-MAC learning.
- The entries with producer as L2VPN and NH as local bundle interfaces are synced entries from MH-AA peer gateway.
- The entries with producer as LOCAL and NH as local bundle interfaces are dynamically learnt local entries.

```
RP/0/RSP0/CPU0:router# show l2route evpn mac-ip evi 6

Topo ID      Mac Address      IP Address      Prod   Next Hop(s)
-----  -----  -----  -----  -----
6          0000.f65a.3569  10.1.1.101     L2VPN  172.16.0.2/24014/ME
6          0000.f65a.3575  10.1.1.97      L2VPN  172.16.0.7/24025/ME
6          0000.f65a.3575  10:1:1::97     L2VPN  172.16.0.7/24025/ME
6          0000.f65a.3575  fe80::200:f6ff:fe5a:3575 L2VPN  172.16.0.7/24025/ME
```

Verify EVPN IRB

6	0000.f65a.357c	10.1.1.93	L2VPN	Bundle-Ether1.11
6	0000.f65a.357c	10:1:1::93	L2VPN	Bundle-Ether1.11
6	0000.f65a.357c	fe80::200:f6ff:fe5a:357c	LOCAL	Bundle-Ether1.11
6	0010.0001.0012	10.1.1.12	L2VPN	172.16.0.7/24025/ME
6	1000.0001.0001	10.1.1.11	LOCAL	Bundle-Ether1.11
6	90e2.ba8e.c0c9	10.1.1.102	L2VPN	172.16.0.2/24014/ME

Verify entries to obtain details of EVPN.

```
RP/0/RSP0/CPU0:router# show evpn evi vpn-id 1 mac ipv4 10.1.1.93 detail
```

EVI	MAC address	IP address	Nexthop	Label
---	-----	-----	-----	-----
1	0000.f65a.357c	10.1.1.93	172.16.0.2	24014
Ethernet Tag : 0				
Multi-paths Resolved : True				
Static : No				
Local Ethernet Segment : N/A				
Remote Ethernet Segment : 0100.6cbc.a77c.c180.0000				
Local Sequence Number : N/A				
Remote Sequence Number : 0				
Local Encapsulation : N/A				
Remote Encapsulation : MPLS				

Verify local BGP entries with appropriate second label and second IP VRF route-target.

```
RP/0/RSP0/CPU0:router# show bgp l2vpn evpn rd 172.16.0.1:1
[2][0][48][0000.f65a.357c][32][10.1.1.93]/136
```

```
BGP routing table entry for [2][0][48][0000.f65a.357c][32][10.1.1.93]/136, Route
Distinguisher: 172.16.0.1:1
Versions:
Process bRIB/RIB SendTblVer
Speaker 3772 3772
Local Label: 24013
Last Modified: Feb 28 16:06:37.073 for 2d19h
Paths: (2 available, best #1)
Advertised to peers (in unique update groups):
172.16.0.9
Path #1: Received by speaker 0
Advertised to peers (in unique update groups):
172.16.0.9
Local
0.0.0.0 from 0.0.0.0 (172.16.0.1)
Second Label 24027 >>> Second label when IRB host-routing
is enabled.
Origin IGP, localpref 100, valid, redistributed, best, group-best, import-candidate,
rib-install
Received Path ID 0, Local Path ID 0, version 3772
Extended community: So0:172.16.0.2:1 RT:100:100
EVPN ESI: 0100.6cbc.a77c.c180.0000
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 (metric 101) from 172.16.0.9 (172.16.0.2)
Received Label 24014, Second Label 24031
```

Origin IGP, localpref 100, valid, internal, add-path, import-candidate, imported, rib-install
Received Path ID 0, Local Path ID 2, version 3769
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100 >>> Second RT is IP VRF RT for
remote to import into IP VRF routing table.
Originator: 172.16.0.2, Cluster list: 172.16.0.9
EVPN ESI: 0100.6cbc.a77c.c180.0000
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1

```
RP/0/RSP0/CPU0:router# show bgp 12vpn evpn rd 172.16.0.1:1
[2][0][48][0000.f65a.357c][128][10:1:1::93]/232

[2][0][48][0000.f65a.357c][128][10:1:1::93]/232
BGP routing table entry for [2][0][48][0000.f65a.357c][128][10:1:1::93]/232, Route
Distinguisher: 172.16.0.1:1
Versions:
Process bRIB/RIB SendTblVer
Speaker 3172 3172
Local Label: 24013
Last Modified: Feb 28 11:34:33.073 for 3d00h
Paths: (2 available, best #1)
Advertised to peers (in unique update groups):
172.16.0.9
Path #1: Received by speaker 0
Advertised to peers (in unique update groups):
172.16.0.9
Local
0.0.0.0 from 0.0.0.0 (172.16.0.1)
Second Label 24029
Origin IGP, localpref 100, valid, redistributed, best, group-best, import-candidate,
rib-install
Received Path ID 0, Local Path ID 0, version 3172
Extended community: SoO:172.16.0.2:1 RT:100:100
EVPN ESI: 0100.6cbc.a77c.c180.0000
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 (metric 101) from 172.16.0.9 (172.16.0.2)
Received Label 24014, Second Label 24033
Origin IGP, localpref 100, valid, internal, add-path, import-candidate, imported, rib-install
Received Path ID 0, Local Path ID 2, version 3167
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100
Originator: 172.16.0.2, Cluster list: 172.16.0.9
EVPN ESI: 0100.6cbc.a77c.c180.0000
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1
```

Verify the remote peer gateway BGP entries with correct label and route-target. Particularly verify the local auto-generated RD on a remote EVPN gateway. EVPN type-2 routes are imported into EVPN. The host routes of IPv4 /32 addresses are imported only into IP VRF route-table in the remote EVPN gateway, but not in the local EVPN gateway where local BVI adjacency is used to overwrite RIB entries.

```
RP/0/RSP0/CPU0:router# show bgp 12vpn evpn rd 172.16.0.7:1
[2][0][48][0000.f65a.357c][32][10.1.1.93]/136
BGP routing table entry for [2][0][48][0000.f65a.357c][32][10.1.1.93]/136, Route
Distinguisher: 172.16.0.7:1
Versions:
Process bRIB/RIB SendTblVer
```

Verify EVPN IRB

```

Speaker 16712 16712
Last Modified: Feb 28 16:06:36.448 for 2d19h
Paths: (2 available, best #1)
Not advertised to any peer
Path #1: Received by speaker 0
Not advertised to any peer
Local
172.16.0.1 from 172.16.0.9 (172.16.0.1)
Received Label 24013, Second Label 24027 >>> First label for L2 MAC unicast bridging;
second label for EVPN IRB host-routing
Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported,
rib-install
Received Path ID 0, Local Path ID 0, version 16712
Extended community: SoO:172.16.0.2:1 RT:100:1 RT:100:100
Originator: 172.16.0.1, Cluster list: 172.16.0.9
EVPN ESI: 0100.6cbc.a77c.c180.0000
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.1:1
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 from 172.16.0.9 (172.16.0.2)
Received Label 24014, Second Label 24031
Origin IGP, localpref 100, valid, internal, backup, add-path, import-candidate, imported,
rib-install
Received Path ID 0, Local Path ID 1, version 16706
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100
Originator: 172.16.0.2, Cluster list: 172.16.0.9
EVPN ESI: 0100.6cbc.a77c.c180.0000
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1

```

```

RP/0/RSP0/CPU0:router# show bgp l2vpn evpn rd 172.16.0.7:1
[2] [0] [48] [0000.f65a.357c] [128] [10:1:1::93]/232

BGP routing table entry for [2] [0] [48] [0000.f65a.357c] [128] [10:1:1::93]/232, Route
Distinguisher: 172.16.0.7:1
Versions:
Process bRIB/RIB SendTblVer
Speaker 6059 6059
Last Modified: Feb 28 12:03:22.448 for 2d23h
Paths: (2 available, best #1)
Not advertised to any peer
Path #1: Received by speaker 0
Not advertised to any peer
Local
172.16.0.1 from 172.16.0.9 (172.16.0.1)
Received Label 24013, Second Label 24029
Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported,
rib-install
Received Path ID 0, Local Path ID 0, version 6043
Extended community: SoO:172.16.0.2:1 RT:100:1 RT:100:100
Originator: 172.16.0.1, Cluster list: 172.16.0.9
EVPN ESI: 0100.6cbc.a77c.c180.0000
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.1:1
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 from 172.16.0.9 (172.16.0.2)
Received Label 24014, Second Label 24033
Origin IGP, localpref 100, valid, internal, backup, add-path, import-candidate, imported,
rib-install
Received Path ID 0, Local Path ID 1, version 6059

```

```
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100
Originator: 172.16.0.2, Cluster list: 172.16.0.9
EVPN ESI: 0100.6cbc.a77c.c180.0000
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1
```

Verify the remote peer gateway with host routes of IPv4 /32 addresses imported into the IP VRF routing table.

```
RP/0/RSP0/CPU0:router# show bgp vpng4 unicast vrf evpn1 10.1.1.93/32

BGP routing table entry for 10.1.1.93/32, Route Distinguisher: 172.16.0.7:11
Versions:
Process bRIB/RIB SendTblVer
Speaker 22202 22202
Last Modified: Feb 28 16:06:36.447 for 2d19h
Paths: (2 available, best #1)
Not advertised to any peer
Path #1: Received by speaker 0
Not advertised to any peer
Local
172.16.0.1 from 172.16.0.9 (172.16.0.1)
Received Label 24027
Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported
Received Path ID 0, Local Path ID 0, version 22202
Extended community: SoO:172.16.0.2:1 RT:100:1 RT:100:100
Originator: 172.16.0.1, Cluster list: 172.16.0.9
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.1:1 >>>
The source from

>>> L2VPN and from

>>> synced ARP entry.
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 from 172.16.0.9 (172.16.0.2)
Received Label 24031
Origin IGP, localpref 100, valid, internal, backup, add-path, import-candidate, imported
Received Path ID 0, Local Path ID 1, version 22201
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100
Originator: 172.16.0.2, Cluster list: 17.0.0.9
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1 >>>
The source from

>>> L2VPN and

>>> from dynamic

>>> ARP entry.
```

```
RP/0/RSP0/CPU0:router# show bgp vpng6 unicast vrf evpn1 10:1:1::93/128
```

```
BGP routing table entry for 10:1:1::93/128, Route Distinguisher: 172.16.0.7:11
Versions:
Process bRIB/RIB SendTblVer
Speaker 22163 22163
```

Verify EVPN IRB

```
Last Modified: Feb 28 12:09:30.447 for 2d23h
Paths: (2 available, best #1)
Not advertised to any peer
Path #1: Received by speaker 0
Not advertised to any peer
Local
172.16.0.1 from 172.16.0.9 (172.16.0.1)
Received Label 24029
Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported
Received Path ID 0, Local Path ID 0, version 22163
Extended community: SoO:172.16.0.2:1 RT:100:1 RT:100:100
Originator: 172.16.0.1, Cluster list: 172.16.0.9
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.1:1 >>>
Source from

>>> L2VPN and from

>>> synced ARP entry.
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 from 172.16.0.9 (172.16.0.2)
Received Label 24033
Origin IGP, localpref 100, valid, internal, backup, add-path, import-candidate, imported
Received Path ID 0, Local Path ID 1, version 22163
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100
Originator: 172.16.0.2, Cluster list: 172.16.0.9
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1 >>>
Source from

>>> L2VPN and from

>>> dynamic ARP entry.
```

```
RP/0/RSP0/CPU0:router# show bgp vpng6 unicast vrf evpn1 10:1:1::93/128

BGP routing table entry for 10:1:1::93/128, Route Distinguisher: 172.16.0.7:11
Versions:
Process bRIB/RIB SendTblVer
Speaker 22163 22163
Last Modified: Feb 28 12:09:30.447 for 2d23h
Paths: (2 available, best #1)
Not advertised to any peer
Path #1: Received by speaker 0
Not advertised to any peer
Local
172.16.0.1 from 172.16.0.9 (172.16.0.1)
Received Label 24029
Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported
Received Path ID 0, Local Path ID 0, version 22163
Extended community: SoO:172.16.0.2:1 RT:100:1 RT:100:100
Originator: 172.16.0.1, Cluster list: 172.16.0.9
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.1:1
Path #2: Received by speaker 0
Not advertised to any peer
Local
172.16.0.2 from 172.16.0.9 (172.16.0.2)
Received Label 24033
Origin IGP, localpref 100, valid, internal, backup, add-path, import-candidate, imported
Received Path ID 0, Local Path ID 1, version 22163
Extended community: SoO:172.16.0.2:1 RT:200:1 RT:700:100
```

```
Originator: 172.16.0.2, Cluster list: 172.16.0.9
Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 172.16.0.2:1
```

Verify local forwarding with local adjacency which overwrite the RIB entries, and remote peer that use the IP VRF host route entries for IP VPN forwarding.

```
RP/0/RSP0/CPU0:router# show bgp vpng4 unicast vrf evpn1 10.1.1.93/32
-- For local routing and forwarding
RP/0/RSP0/CPU0:PE11-R1#show route vrf evpn1 10.1.1.93
Routing entry for 10.1.1.93/32
Known via "bgp 3107", distance 200, metric 0, type internal
Installed Feb 28 15:57:28.154 for 2d20h
Routing Descriptor Blocks
172.16.0.2, from 172.16.0.9      >>> From MH-AA peer.
Nexthop in Vrf: "default", Table: "default", IPv4 Unicast, Table Id: 0xe0000000
Route metric is 0
No advertising protos.

RP/0/RSP0/CPU0:PE11-R1# show cef vrf evpn1 10.1.1.93 location 0/0/CPU0
10.1.1.93/32, version 0, internal 0x1120001 0x0 (ptr 0x7b40052c) [1], 0x0 (0x7b286010), 0x0
(0x0)
Updated Feb 28 15:58:22.688
local adjacency 10.1.1.93
Prefix Len 32, traffic index 0, Adjacency-prefix, precedence n/a, priority 15
via 10.1.1.93/32, BV11, 2 dependencies, weight 0, class 0 [flags 0x0]
path-idx 0 NHID 0x0 [0x7f531f88 0x0]
next hop
local adjacency          >>> Forwarding with local synced ARP adjacency entries.
```

For remote routing and forwarding:

```
RP/0/RSP0/CPU0:router# show route vrf evpn1 10.1.1.93
Routing entry for 10.1.1.93/32
Known via "bgp 3107", distance 200, metric 0
Number of pic paths 1 , type internal
Installed Feb 28 16:06:36.431 for 2d20h
Routing Descriptor Blocks
172.16.0.1, from 172.16.0.9
Nexthop in Vrf: "default", Table: "default", IPv4 Unicast, Table Id: 0xe0000000
Route metric is 0
172.16.0.2, from 172.16.0.9, BGP backup path
Nexthop in Vrf: "default", Table: "default", IPv4 Unicast, Table Id: 0xe0000000
Route metric is 0
No advertising protos.

RP/0/RSP0/CPU0:router# show cef vrf evpn1 10.1.1.93 location 0/0/CPU0
10.1.1.93/32, version 86, internal 0x5000001 0x0 (ptr 0x99fac884) [1], 0x0 (0x0), 0x208
(0x96c58494)
Updated Feb 28 16:06:39.285
Prefix Len 32, traffic index 0, precedence n/a, priority 3
via 172.16.0.1/32, 15 dependencies, recursive [flags 0x6000]
path-idx 0 NHID 0x0 [0x97955380 0x0]
recursion-via-/32
next hop VRF - 'default', table - 0xe0000000
next hop 172.16.0.1/32 via 34034/0/21
next hop 100.0.57.5/32 Te0/0/0/3 labels imposed {ImplNull 24011 24027}
next hop 100.0.67.6/32 Te0/0/0/1 labels imposed {ImplNull 24009 24027}
```

Verify EVPN IRB

```

via 172.16.0.2/32, 11 dependencies, recursive, backup [flags 0x6100]
path-idx 1 NHID 0x0 [0x979554a0 0x0]
recursion-via-/32
next hop VRF - 'default', table - 0xe0000000
next hop 172.16.0.2/32 via 34035/0/21
next hop 100.0.57.5/32 Te0/0/0/3 labels imposed {ImplNull 24012 24031}
next hop 100.0.67.6/32 Te0/0/0/1 labels imposed {ImplNull 24010 24031}

```

The following sections describe how to verify the subnet stretching.

Verify the VRF.

```

RP/0/RP0/CPU0:leafW# show run vrf cust130

vrf cust130
address-family ipv4 unicast
import route-target
130:130
!
export route-target
130:130
!
!
```

Verify the BGP configuration.

```

RP/0/RP0/CPU0:leafW# show run router bgp | begin vrf cust130

vrf cust130
rd auto
address-family ipv4 unicast
label mode per-vrf
maximum-paths ibgp 10
redistribute connected
!
```

Verify the L2VPN.

```

RP/0/RP0/CPU0:leafW# show run l2vpn bridge group bg130

l2vpn
bridge group bg130
bridge-domain bd130
interface Bundle-Ether1.1300
!
interface Bundle-Ether5.1300
!
routed interface BVI130
evi 130
!
!
```

EVPN IPv6 Hosts with Mobility

EVPN IPv6 Hosts with Mobility feature enables you to provide EVPN IPv6 service over IPv4-MPLS core network. This feature supports all-active multihoming and virtual machine (VM) or host move.

Service Providers (SPs) use a stable and established core with IPv4-MPLS backbone for providing IPv4 VPN services. The IPv6 VPN Provider Edge Transport over MPLS (IPv6 on Provider Edge Routers [6PE] and IPv6 on VPN Provider Edge Routers [6VPE]) facilitates SPs to offer IPv6 VPN services over IPv4 backbone without an IPv6 core. The provider edge (PE) routers run MP-iBGP to advertise IPv6 reachability and IPv6 label distribution. For 6PE, the labels are allocated per IPv6 prefix learnt from connected customer edge (CE) routers and for 6VPE, the PE router can be configured to allocate labels on a per-prefix or per-CE and per-VRF level.

Mobility Support

In global VRF, mobility is not supported. However, you can move a host from one ES to another ES within the same bridge domain. The host gets a new MAC address and IP address. The host can have multiple IP addresses for the same MAC address.

In non-default VRF, mobility is supported with the following conditions:

- Basic MAC move: The IP address and MAC address remains the same. You can move a host from one ES to another ES with the same IP address and MAC address.
- Same MAC address but with a different IP address: The host gets a new IP address
- Same IP address but with a different MAC address: The host gets a new MAC address but retains the same IP address
- Multiple IP addresses with the same MAC address: Many VMs are involved in the same the MAC move

Restrictions

- In customer VRFs, when host routing is not configured, MAC-IP advertisement is different between zero ESI and none-zero ESI. When host routing is not configured, MAC-IP with non-zero ESI is advertised without L3 RT (VRF RT). MAC-IP with zero ESI is not advertised. The following table lists the behavior of MAC-IP advertisement with respect to ESI and host routing.

ESI Type	With host routing	Without host routing
MAC-IP with non-zero ESI	Advertised with L3 VRF RT	Advertised without L3 VRF RT
MAC-IP with zero ESI	Advertised with L3 VRF RT	Not advertised

- In global VRF, Layer 2 stretch is not supported.
- MAC move in global VRF is only supported if the host is within the same bridge domain. You can move a host from one ES to another ES within the same bridge domain.
- Duplication of IP address detection is not supported.
- Maximum number of leafs allowed per ESI is two.

Configure EVPN IPv6 Hosts with Mobility

Perform the following tasks to configure EVPN IPv6 Hosts with Mobility feature:

- Configure VRF
- Configure ISIS
- Configure BGP
- Configure AC interface
- Configure BVI interface
- Configure EVPN
- Configure L2VPN



Note

- You cannot configure the EVPN remote peer using the VPNV4 unicast if you have configured the **advertise vpng4 unicast re-originated** command under the L2VPN EVPN address-family. You can either configure the VPNV4 unicast or the advertise vpng4 unicast re-originated under L2VPN EVPN address-family.
- You cannot configure the EVPN remote peer using the VPNV6 unicast if you have configured the **advertise vpng6 unicast re-originated** command under the L2VPN EVPN address-family. You can either configure the VPNV6 unicast or the advertise vpng6 unicast re-originated under L2VPN EVPN address-family.

```
/* Configure VRF */

Router# configuration
Router(config)# vrf cust102
Router(config-vrf)# address-family ipv4 unicast
Router(config-vrf-af)# import route-target 160102:16102
Router(config-vrf-af)# export route-target 160102:16102
Router(config-vrf-af)# exit
!
Router(config-vrf)# address-family ipv6 unicast
Router(config-vrf-af)# import route-target 6160102:16102
Router(config-vrf-af)# export route-target 6160102:16102
Router(config-vrf-af)# commit
!

/* Configure ISIS */

Router# configuration
Route(config)# router isis v6
Route(config-isis)# 49.0001.0000.0160.0005.00
Route(config-isis)# nsr
Route(config-isis)# log adjacency changes
Route(config-isis)# lsp-gen-interval maximum-wait 5000 initial-wait 1 secondary-wait
20
Route(config-isis)# lsp-mtu 1468
Route(config-isis)# lsp-refresh-interval 65000
```

```

Route(config-isis)# max-lsp-lifetime 65535
Route(config-isis)# address-family ipv4 unicast
Route(config-isis-af)# metric-style wide
Route(config-isis-af)# microloop avoidance protected
Route(config-isis-af)# spf-interval maximum-wait 5000 initial-wait 1 secondary-wait 20
Route(config-isis-af)# segment-routing mpls sr-prefer
Route(config-isis-af)# segment-routing prefix-sid-map advertise-local
Route(config-isis-af)# exit
!
Route(config-isis)# interface Bundle-Ether10
Route(config-isis-if)# point-to-point
Route(config-isis-if)# address-family ipv4 unicast
Route(config-isis-af)# fast-reroute per-prefix
Route(config-isis-af)# fast-reroute per-prefix ti-lfa
Route(config-isis-af)# metric 10
Route(config-isis-af)# exit
!
Route(config-isis)# interface Bundle-Ether20
Route(config-isis-if)# point-to-point
Route(config-isis-if)# address-family ipv4 unicast
Route(config-isis-af)# fast-reroute per-prefix
Route(config-isis-af)# fast-reroute per-prefix ti-lfa
Route(config-isis-af)# metric 10
Route(config-isis-af)# exit
!
Route(config-isis)# interface loopback0
Route(config-isis-if)# passive
Route(config-isis-if)# address-family ipv4 unicast
Route(config-isis-af)# exit
!
Route(config-isis)# interface loopback10
Route(config-isis-if)# passive
Route(config-isis-if)# address-family ipv4 unicast
Route(config-isis-af)# prefix-sid index 1605
Route(config-isis-af)# commit
Route(config-isis-af)# exit
!

/* Configure Segment Routing */

Router# configure
Router(config)# segment-routing
Router(config-sr)# global-block 16000 23999
Router(config-sr)# commit

/* Configure BGP */

Router(config)# router bgp 100
Router(config-bgp)# bfd minimum-interval 50
Router(config-bgp)# bfd multiplier 3
Router(config-bgp)# bgp router-id 160.0.0.5
Router(config-bgp)# address-family ipv4 unicast      --> To support V4 Global VRF
Router(config-bgp-af)# maximum-paths ibgp 10 unequal-cost  --> ECMP
Router(config-bgp-af)# redistribute connected      --> V4 Global VRF
Router(config-bgp-af)# exit
!
Router(config-bgp)# address-family ipv4 unicast      --> VRF
Router(config-bgp-af)# vrf all
Router(config-bgp-af)# label mode per-vrf
Router(config-bgp-af)# exit
!
Router(config-bgp)# address-family ipv6 unicast      --> For 6PE
Router(config-bgp-af)# label mode per-vrf

```

Configure EVPN IPv6 Hosts with Mobility

```

Router(config-bgp-af)# maximum-paths ibgp 8
Router(config-bgp-af)# redistribute static
Router(config-bgp-af)# allocate-label all
Router(config-bgp-af)# exit
!
Router(config-bgp)# address-family vpnv6 unicast    ---> 6 VPE
Router(config-bgp-af)# vrf all
Router(config-bgp-af)# label mode per-vrf
Router(config-bgp-af)# exit
!
Router(config-bgp)# address-family l2vpn evpn      ----> EVPN
Router(config-bgp-af)# bgp implicit-import          ----> Global VRF
Router(config-bgp-af)# exit
!
Router(config-bgp)# neighbor-group evpn-rr
Router(config-bgp-nbr)# remote-as 100
Router(config-bgp-nbr)# bfd fast-detect
Router(config-bgp-nbr)# update-source loopback0
Router(config-bgp-nbr)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy nh-lo10 out
Router(config-bgp-nbr-af)# exit
!
Router(config-bgp-nbr)# address-family ipv6 labeled-unicast  ----> For 6PE
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# exit
!
Router(config-bgp-nbr)# address-family l2vpn evpn
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy nh-lo10 out
Router(config-bgp-nbr-af)# advertise vpnv4 unicast re-originated -> For Route Type 5
Router(config-bgp-nbr-af)# advertise vpnv6 unicast re-originated -> For Route Type 5
Router(config-bgp-nbr-af)# exit
!
Router(config-bgp)# neighbor 160.0.0.1
Router(config-bgp-nbr)# use neighbor-group evpn-rr
Router(config-bgp-nbr)# exit
!
Router(config-bgp)# neighbor 160.0.0.2
Router(config-bgp-nbr)# use neighbor-group evpn-rr
Router(config-bgp-nbr)# exit
!
Router(config-bgp)# vrf all
Router(config-bgp-vrf)# rd 1605:102
Router(config-bgp-vrf)# address-family ipv4 unicast
Router(config-bgp-vrf-af)# label mode per-vrf
Router(config-bgp-vrf-af)# maximum-paths ibgp 10 unequal-cost
Router(config-bgp-vrf-af)# redistribute connected   ---> Triggers Route Type 5
Router(config-bgp-vrf-af)# exit
!
Router(config-bgp-vrf)# address-family ipv6 unicast
Router(config-bgp-vrf-af)# label mode per-vrf
Router(config-bgp-vrf-af)# maximum-paths ibgp 10 unequal-cost
Router(config-bgp-vrf-af)# redistribute connected
Router(config-bgp-vrf-af)# exit
!

/* Configure AC interface */

Router(config)# interface Bundle-Ether1.102 l2transport
Router(config-l2vpn-subif)# encapsulation dot1q 102
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

```

```

Router(config-l2vpn-subif)# exit

/* Configure BVI interface */

Router(config)# interface BVI100
Router(config-if)# ipv4 address 56.78.100.1 255.255.255.0
Router(config-if)# ipv6 address 56:78:100::1/64
Router(config-if)# mac-address 22.22.22
Router(config-if)# exit
!
Router(config)# interface BVI102
Router(config-if)# host-routing
Router(config-if)# vrf cust102
Router(config-if-vrf)# ipv4 address 56.78.102.1 255.255.255.0
Router(config-if-vrf)# ipv6 nd dad attempts 0
Router(config-if-vrf)# ipv6 address 56:78:100::1/64
Router(config-if-vrf)# ipv6 address 56:78:102::1/64
Router(config-if-vrf)# mac-address 22.22.22
Router(config-if)# commit

/* Configure CEF */ [Required for dual homing]

Router# configure
Router(config)# cef adjacency route override rib

/* Configure EVPN, and configure main bundle ethernet segment parameters in EVPN */

Router# configure
Router(config)# evpn
Router(config-evpn)# evi 102
Router(config-evpn-evi)# bgp
Router(config-evpn-evi)# rd 1605:102
Router(config-evpn-evi-bgp)# route-target import 160102:102
Router(config-evpn-evi-bgp)# route-target export 160102:102
Router(config-evpn-evi-bgp)# exit
Router(config-evpn-evi)# advertise-mac
Router(config-evpn-evi)# exit
!
Router(config-evpn)# interface Bundle-Ether1
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 56.56.56.56.56.56.56.01
Router(config-evpn-ac-es)# exit
!
Router(config-evpn)# interface Bundle-Ether2
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 56.56.56.56.56.56.56.02
Router(config-evpn-ac-es)# commit

/* Configure L2VPN */

Router# configure
Router(config)# l2vpn
Router(config-l2vpn)# bridge group bg102
Router(config-l2vpn-bg)# bridge-domain bd102
Router(config-l2vpn-bg-bd)# interface Bundle-Ether1.102
Router(config-l2vpn-bg-bd-ac)# exit
!
Router(config-l2vpn-bg-bd)# interface Bundle-Ether2.102
Router(config-l2vpn-bg-bd-ac)# exit
!
Router(config-l2vpn-bg-bd)# interface Bundle-Ether3.102
Router(config-l2vpn-bg-bd-ac)# exit
!
```

Running Configuration

```

Router(config-l2vpn-bg-bd)# interface Bundle-Ether4.102
Router(config-l2vpn-bg-bd-ac)# exit
!
Router(config-l2vpn-bg-bd)# interface Bundle-Ether5.102
Router(config-l2vpn-bg-bd-ac)# routed interface BVI102
Router(config-l2vpn-bg-bd-bvi)# evi 102
Router(config-l2vpn-bg-bd-bvi-evi)# commit

```

Running Configuration

```

/* Configure VRF */

vrf cust102
address-family ipv4 unicast
import route-target
160102:16102
!
export route-target
160102:16102
!
!
address-family ipv6 unicast
import route-target
6160102:16102
!
export route-target
6160102:16102
!
!
!

/* Configure ISIS */

router isis v6
net 49.0001.0000.0160.0005.00
nsr
log adjacency changes
lsp-gen-interval maximum-wait 5000 initial-wait 1 secondary-wait 20
lsp-mtu 1468
lsp-refresh-interval 65000
max-lsp-lifetime 65535
address-family ipv4 unicast
metric-style wide
microloop avoidance protected
spf-interval maximum-wait 5000 initial-wait 1 secondary-wait 20
segment-routing mpls sr-prefer
segment-routing prefix-sid-map advertise-local
!
interface Bundle-Ether10
point-to-point
address-family ipv4 unicast
fast-reroute per-prefix
fast-reroute per-prefix ti-lfa
metric 10
!
!
interface Bundle-Ether20
point-to-point
address-family ipv4 unicast
fast-reroute per-prefix
fast-reroute per-prefix ti-lfa
metric 10

```

```
!
!
interface Loopback0
passive
address-family ipv4 unicast
!
!
interface Loopback10
passive
address-family ipv4 unicast
prefix-sid index 1605
!
!
!

/* Configure Segment Routing */

segment-routing
global-block 16000 23999
!

/* Configure BGP */

router bgp 100
bfd minimum-interval 50
bfd multiplier 3
bgp router-id 160.0.0.5
address-family ipv4 unicast      ---> To support V4 Global VRF
maximum-paths ibgp 10 unequal-cost ---> ECMP
redistribute connected    --> V4 Global VRF
!
address-family vpnv4 unicast ---> VRF
vrf all
label mode per-vrf
!
address-family ipv6 unicast      ---> For 6PE
label mode per-vrf
maximum-paths ibgp 8
redistribute connected
redistribute static
allocate-label all
!
address-family vpnv6 unicast      ---> 6VPE
vrf all
label mode per-vrf
!
address-family l2vpn evpn      ----> EVPN
bgp implicit-import      ----> Global VRF
!

neighbor-group evpn-rr
remote-as 100
bfd fast-detect
update-source Loopback0
address-family ipv4 unicast
route-policy pass-all in
route-policy nh-lo10 out
!
address-family ipv6 labeled-unicast  ----> For 6PE
route-policy pass-all out
!
address-family l2vpn evpn
route-policy pass-all in
route-policy nh-lo10 out
```

Running Configuration

```

advertise vpng4 unicast re-originated    ---> For Route Type 5
advertise vpng6 unicast re-originated    ----> For Route Type 5
!
!
neighbor 160.0.0.1
use neighbor-group evpn-rr
!
neighbor 160.0.0.2
use neighbor-group evpn-rr
!
vrf cust102
rd 1605:102
address-family ipv4 unicast
label mode per-vrf
maximum-paths ibgp 10 unequal-cost
redistribute connected <----- Triggers Route Type 5
!
address-family ipv6 unicast
label mode per-vrf
maximum-paths ibgp 10 unequal-cost
redistribute connected
!
!

/* Configure AC interface */

interface Bundle-Ether1.102 l2transport
encapsulation dot1q 102
rewrite ingress tag pop 1 symmetric
!
/* Configure BVI interface */
interface BVI100
ipv4 address 56.78.100.1 255.255.255.0
ipv6 address 56:78:100::1/64
mac-address 22.22.22
!
interface BVI102
host-routing
vrf cust102
ipv4 address 56.78.102.1 255.255.255.0
ipv6 nd dad attempts 0
ipv6 address 56:78:100::1/64
ipv6 address 56:78:102::1/64
mac-address 22.22.22
!

/*
Configure CEF */ [ Required for Dual homing]

cef adjacency route override rib

/* Configure EVPN */

evpn
evi 102
bgp
rd 1605:102
route-target import 160102:102
route-target export 160102:102
!
advertise-mac
!
!
!
```

```

interface Bundle-Ether1
  ethernet-segment
  identifier type 0 56.56.56.56.56.56.56.01
!
!
interface Bundle-Ether2
  ethernet-segment
  identifier type 0 56.56.56.56.56.56.56.02
!
!

/* Configure L2VPN */

l2vpn
  bridge group bg102
  bridge-domain bd102
  interface Bundle-Ether1.102
!
  interface Bundle-Ether2.102
!
  interface Bundle-Ether3.102
!
  interface Bundle-Ether4.102
!
  interface Bundle-Ether5.102
!
  routed interface BVI102
!
evi 102
!
!
!
```

Verification

Verify that you have configured EVPN IPv6 Hosts with Mobility feature is configured.

```

/* 6PE and Static Route Advertisement */
Host route is advertised as EVPN Route Type 2

Router# show bgp ipv6 unicast 56:78:100::2
BGP routing table entry for 56:78:100::2/128
Versions:
  Process bRIB/RIB SendTblVer
  Speaker 212 212
  Local Label: 2
Last Modified: Oct 31 19:13:10.998 for 00:00:19
Paths: (1 available, best #1)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Not advertised to any peer
  Local
  160.5.5.5 (metric 20) from 160.0.0.1 (160.0.0.5)
  Received Label 2
  Origin IGP, localpref 100, valid, internal, best, group-best, imported
  Received Path ID 0, Local Path ID 0, version 212
  Extended community: Flags 0x20: SoO:160.5.5.5:100 RT:160100:100
  mac: 00:06:01:00:01:02
  Originator: 160.0.0.5, Cluster list: 100.0.0.4
  Source AFI: L2VPN EVPN, Source VRF: default, Source Route Distinguisher: 1605:100

/*
  Manually configured static route in global VRF */

```

Verification

```

Router# show bgp ipv6 unicast 56:78:100::2

BGP routing table entry for 30::1/128
Versions:
  Process bRIB/RIB SendTblVer
  Speaker 9 9
  Local Label: 2
Last Modified: Oct 30 20:25:17.159 for 23:15:55
Paths: (2 available, best #2)
  Advertised to update-groups (with more than one peer):
    0.2
      Path #1: Received by speaker 0
      Not advertised to any peer
      Local
      160.0.0.6 (metric 20) from 160.0.0.1 (160.0.0.6)
      Received Label 2
      Origin incomplete, metric 0, localpref 100, valid, internal, labeled-unicast
      Received Path ID 0, Local Path ID 0, version 0
      mac: 10:11:04:64:f2:7f
      Originator: 160.0.0.6, Cluster list: 100.0.0.4
      Path #2: Received by speaker 0
      Advertised to update-groups (with more than one peer):
        0.2
        Local
        56:78:100::2 from :: (160.0.0.5)
        Origin incomplete, metric 0, localpref 100, weight 32768, valid, redistributed, best,
        group-best
        Received Path ID 0, Local Path ID 0, version 9
        mac: 10:11:04:64:f2:7f

/* Verify Ethernet Segments are peering for Dual homing */

Router# show evpn ethernet-segment int bundle-Ether 1

Ethernet Segment Id Interface Nexthops
-----
0056.5656.5656.5656.5601 BE1 160.5.5.5
                                         160.6.6.6
-----

/* Verify DF election */

Router# show evpn ethernet-segment int bundle-Ether 1 carving detail
Legend:
A - Load-balancing mode and Access Protection incompatible,
B - No Forwarders EVPN-enabled,
C - Backbone Source MAC missing (PBB-EVPN),
RT - ES-Import Route Target missing,
E - ESI missing,
H - Interface handle missing,
I - Name (Interface or Virtual Access) missing,
M - Interface in Down state,
O - BGP End of Download missing,
P - Interface already Access Protected,
Pf - Interface forced single-homed,
R - BGP RID not received,
S - Interface in redundancy standby state,
X - ESI-extracted MAC Conflict
SHG - No local split-horizon-group label allocated

Ethernet Segment Id Interface Nexthops
-----
0056.5656.5656.5656.5601 BE1 160.5.5.5

```

```

160.6.6.6
ES to BGP Gates : Ready
ES to L2FIB Gates : Ready
Main port :
Interface name : Bundle-Ether1
Interface MAC : 008a.9644.acdd
IfHandle : 0x080004dc
State : Up
Redundancy : Not Defined
ESI type : 0
Value : 56.5656.5656.5656.5601
ES Import RT : 5656.5656.5656 (from ESI)
Source MAC : 0000.0000.0000 (N/A)
Topology :
Operational : MH
Configured : All-active (AApF) (default)
Primary Services : Auto-selection
Secondary Services: Auto-selection
Service Carving Results:
Forwarders : 161
Permanent : 10
EVI:ETag P : 700:1, 701:1, 702:1, 703:1, 704:1, 705:1
EVI:ETag P : 706:1, 707:1, 708:1, 709:1
Elected : 76
EVI E : 100, 102, 104, 106, 108, 110
EVI E : 112, 114, 116, 118, 120, 122,
EVI E : 124, 126, 128, 130, 132, 134,
EVI E : 136, 138, 140, 142, 144, 146,
EVI E : 148, 150, 152, 154, 156, 158,
EVI E : 160, 162, 164, 166, 168, 170,
EVI E : 172, 174, 176, 178, 180, 182,
EVI E : 184, 186, 188, 190, 192, 194,
EVI E : 196, 198, 200, 202, 204, 206,
EVI E : 208, 210, 212, 214, 216, 218,
EVI E : 220, 222, 224, 226, 228, 230,
EVI E : 232, 234, 236, 238, 240, 242,
EVI E : 244, 246, 248, 250
Not Elected : 75
EVI NE : 101, 103, 105, 107, 109, 111
EVI NE : 113, 115, 117, 119, 121, 123,
EVI NE : 125, 127, 129, 131, 133, 135,
EVI NE : 137, 139, 141, 143, 145, 147,
EVI NE : 149, 151, 153, 155, 157, 159,
EVI NE : 161, 163, 165, 167, 169, 171,
EVI NE : 173, 175, 177, 179, 181, 183,
EVI NE : 185, 187, 189, 191, 193, 195,
EVI NE : 197, 199, 201, 203, 205, 207,
EVI NE : 209, 211, 213, 215, 217, 219,
EVI NE : 221, 223, 225, 227, 229, 231,
EVI NE : 233, 235, 237, 239, 241, 243,
EVI NE : 245, 247, 249
MAC Flushing mode : STP-TCN
Peering timer : 3 sec [not running]
Recovery timer : 30 sec [not running]
Carving timer : 0 sec [not running]
Local SHG label : 68663
Remote SHG labels : 1
68670 : nexthop 160.6.6.6

```

EVPN IRB: DHCPv4 and DHCPv6 Relay

EVPN IRB: DHCPv4 and DHCPv6 Relay feature provides DHCP support for the end users in EVPN multi-homing Active-Active (MH-AA) deployment scenario. This feature enables reduction of traffic flooding, increase in load sharing at VTEP, faster convergence during link and device failures, and simplification of data center automation.

DHCPv4 and DHCPv6 Relay agents relay request packets, coming over the access interface, to external DHCPv4 and DHCPv4 server to request allocation of addresses (/32) and IANA (::128) for the end user.

DHCPv4 and DHCPv6 Relay profiles are configured on BVI interfaces which relay DHCPv4 or DHCPv4 requests from Layer 2 (L2) attachment circuit (AC) to external DHCP servers for host IPv4 addresses (/32) and IANA (::128) IPv6 addresses.

This feature is compliant with RFC-6607.

Multi-homing Active-Active EVPN Gateways

Multi-homing Active-Active EVPN Gateways are configured with anycast IP address and MAC addresses. ASR 9000 devices have centralized L2/Layer 3 (L3) gateway. Based on native EVPN and MAC learning, IRB uses distributed anycast IP and anycast MAC address. Static clients are configured with anycast gateway address as the default gateway. DHCP client sends DHCP requests for IP addresses with BVI as the gateway. L2 access can be either single homing or multi-homing. Not all access protocols are supported with IRB. There may or may not be L2 stretch between DC centers. Internet gateway is also included for clients to access external network. No EVPN is configured on the Internet gateway.

EVPN IRB Route Distribution

In EVPN IRB DHCPv4 and DHCPv6, DHCP application processes and DHCP packet forwarding are independent of EVPN IRB L2 and L3 routing. There is no subscriber routing information with the stateless DHCP relay. But DHCP clients work similar to static clients in the EVPN core for L2 and L3 bridging and routing. When the **relay information option**, **relay information option vpn**, **relay information option von-mode cisco** and **relay information option von-mode rfc** commands are configured on the DHCP relay agent, the DHCP relay agent inserts the sub options of DHCP Option 82, such as subnet selection and VPN ID options. These options are considered by DHCP server while allocating IP address.

DHCP clients use the L2 AC interface to access EVPN bridge domain and use BVI interface as default gateway. So the clients must get the IP addresses from the DHCP server as in the same subnet of BVI interface.

DHCP Request Forwarding Path

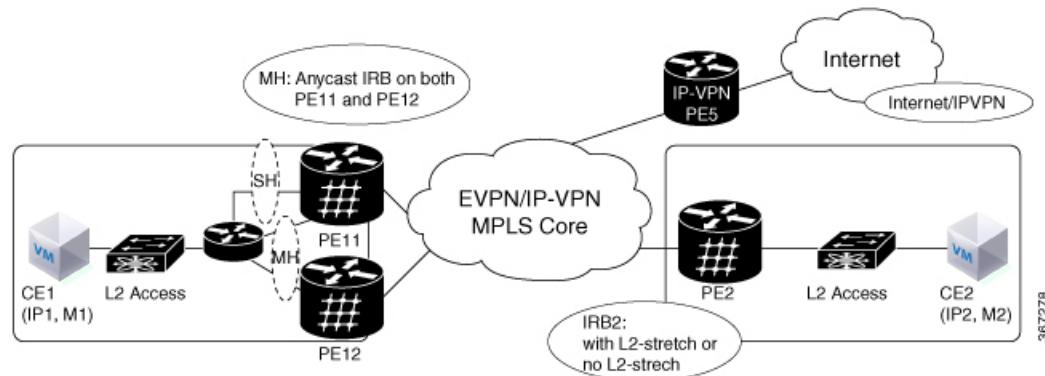
Clients broadcast requests to the access switch with DHAA to EVPN PE routers. The access switch does load balancing. The load balancing configurations in access switch will impact PE in DH-AA and DHCP to send the DHCP requests. The DHCP request reaches the Bridge Domain (BD) BVI interface which is configured with DHCP relay. Because AA PE routers are configured the same IP addresses, BVI IP addresses cannot be used as DHCP relay source IP address.

For DHCPv4, configuring GIADDR field for each DHCP relay profile is allowed. Loopback interface with unique IP addresses can be configured in VRF which is reachable to DHCP servers. Configuring DHCP relay source address is not supported.

In case of DHCPv6 servers, DHCPv6 relay picks up an available Loopback interface IPv6 address as DHCPv6 relay source IP address. After the DHCP clients get the IP address. DHCP clients are not normally routable to DHCP servers. DHCP clients send unicast DHCP renew messages to the DHCP server. If the DHCP servers are not routable, the DHCP unicast messages fail, then the DHCP client sends broadcast rebinding messages with the corresponding DHCP relay.

In the below figure, DHCP clients are configured on PE11 and PE12.

Figure 3: EVPN IRB with ASR 9000 as Centralized DCI Gateway



Configure EVPN IRB: DHCPv4 and DHCPv6 Relay

Perform the following tasks to configure the EVPN IRB: DHCPv4 and DHCPv6 Relay feature:

```
/* PE11 configuration */

Router(config)# dhcp ipv4
Router(config-dhcpv4)# profile DHCPv4_RELAY relay
Router(config-dhcpv4-relay-profile)# helper-address vrf default 10.20.20.20 giaddr 192.0.2.1
Router(config-dhcpv4-relay-profile)# relay information option vpn
Router(config-dhcpv4-relay-profile)# relay information option vpn-mode rfc
Router(config-dhcpv4-relay-profile)# exit
Router(config-dhcpv4)# exit
Router(config)# interface BVII relay profile DHCPv4_RELAY

Router(config)# dhcp ipv6
Router(config-dhcpv6)# profile DHCPv6_RELAY relay
Router(config-dhcpv6-relay-profile)# helper-address vrf default 20::20
Router(config-dhcpv6-relay-profile)# exit
Router(config-dhcpv6)# exit
Router(config)# interface BVII relay profile DHCPv6_RELAY
Router(config-if)# exit
Router(config)# interface Loopback 5
Router(config-if)# ipv4 address 192.0.2.1 255.255.255.255
Router(config-if)# exit
Router(config)# interface BVII
Router(config-if)# host-routing
Router(config-if)# ipv4 address 10.10.10.2 255.255.255.0
Router(config-if)# ipv6 address 2001:DB8:0:ABCD::1/64
Router(config-if)# ipv6 enable
Router(config-if)# mac-address 0.12.3456

/* PE12 configuration */
Router# dhcp ipv4
Router(config-dhcpv4)# profile DHCPv4_RELAY relay
Router(config-dhcpv4-relay-profile)# helper-address vrf default 10.20.20.20 giaddr 127.0.0.1
Router(config-dhcpv4-relay-profile)# relay information option vpn
Router(config-dhcpv4-relay-profile)# relay information option vpn-mode cisco
Router(config-dhcpv4-relay-profile)# exit
```

Configure EVPN IRB: DHCPv4 and DHCPv6 Relay

```

Router(config-dhcpv4)# exit
Router(config)# interface BVI1 relay profile DHCPv4_RELAY
Router(config)# dhcp ipv6
Router(config-dhcpv6)# profile DHCPv6_RELAY relay
Router(config-dhcpv6-relay-profile)# helper-address vrf default 20::20
Router(config-dhcpv6-relay-profile)# exit
Router(config-dhcpv6)# exit
Router(config) interface BVI1 relay profile DHCPv6_RELAY
Router(config)# interface Loopback 6
Router(config)# exit
Router(config-if)# ipv4 address 127.0.0.1 255.255.255.255

Router(config)# interface BVI1
Router(config-if)# host-routing
Router(config-if)# vrf evpn 1
Router(config-if)# exit
Router(config-if)# ipv4 address 10.10.10.2 255.255.255.0
Router(config-if)# proxy-arp
Router(config-if)# ipv6 address 3000:0:0:8003::2/64
Router(config-if)# ipv6 enable
Router(config-if)# mac-address 1122.3344.5566

```

Running Configuration

```

/* PE11 Configuration */

dhcp ipv4
profile DHCPv4_RELAY relay
    helper-address vrf default 10.20.20.20 giaddr 192.0.2.1
    relay information option vpn
    relay information option vpn-mode cisco
!
interface BVI1 relay profile DHCPv4_RELAY
!
dhcp ipv6
profile DHCPv6_RELAY relay
    helper-address vrf default 20::20
!
interface BVI1 relay profile DHCPv6_RELAY
!
interface Loopback5
ipv4 address 192.0.2.1 255.255.255.0
!
interface BVI1
host-routing
ipv4 address 10.10.10.2 255.255.255.0
ipv6 address 2001:DB8:0:ABCD::1/64
ipv6 enable
mac-address 0.12.3456
!

/* PE12 Configuration */

dhcp ipv4
profile DHCPv4_RELAY relay
    helper-address vrf default 10.20.20.20 giaddr 127.0.0.1
    relay information option vpn
    relay information option vpn-mode cisco
!
interface BVI1 relay profile DHCPv4_RELAY

```

```

!
dhcp ipv6
profile DHCPv6_RELAY relay
  helper-address vrf default 20::20
!
interface BVI1 relay profile DHCPv6_RELAY
!
interface Loopback6
  ipv4 address 127.0.0.1 255.255.255.255
!
interface BVI1
host-routing
vrf evpn1
  ipv4 address 10.10.10.2 255.255.255.0
  proxy-arp
  ipv6 address 3000:0:0:8003::2/64
  ipv6 enable
  mac-address 1122.3344.5566
!
```

Verification

Verify the DHCPv4 configuration.

```

Router# show running-configuration dhcp ipv4
Thu Feb 15 21:44:31.550 IST
dhcp ipv4
  profile DHCPv4_RELAY relay
    helper-address vrf default 10.20.20.20 giaddr 192.0.2.1
    relay information option vpn
    relay information option vpn-mode rfc
!
interface BVI1 relay profile DHCPv4_RELAY
!
```

Verify the DHCPv4 relay profile details.

```

Router# show dhcp ipv4 relay profile name DHCPv4_RELAY
Thu Feb 15 21:47:32.247 IST
Profile: DHCPv4_RELAY
Helper Addresses:
  10.20.20.20, vrf default, giaddr 192.0.2.1
Information Option: Disabled
Information Option Allow Untrusted: Disabled
Information Option VPN: Enabled
Information Option VPN Mode: RFC
Information Option Policy: Replace
Information Option Check: Disabled
GIADDR Policy: Keep
Broadcast-flag Policy: Ignore
Mac Mismatch Action: Forward
VRF References:
Interface References: BVI1
```

Verify the DHCPv4 relay packet statistics.

Configure EVPN IRB: DHCPv4 and DHCPv6 Relay

```
Router# show dhcp ipv4 relay statistics
```

Fri Feb 16 12:34:51.202 IST

VRF	RX	TX	DR
default	4	4	0
**nVSatellite	0	0	0

Verify DHCPv4 relay packet statistics in detail.

```
Router# show dhcp vrf default ipv4 relay statistics
```

Fri Feb 16 12:36:05.544 IST

DHCP IPv4 Relay Statistics for VRF default:

TYPE	RECEIVE	TRANSMIT	DROP
DISCOVER	1	1	0
OFFER	1	1	0
REQUEST	1	1	0
DECLINE	0	0	0
ACK	1	1	0
NAK	0	0	0
RELEASE	0	0	0
INFORM	0	0	0
LEASEQUERY	0	0	0
LEASEUNASSIGNED	0	0	0
LEASEUNKNOWN	0	0	0
LEASEACTIVE	0	0	0
BOOTP-REQUEST	0	0	0
BOOTP-REPLY	0	0	0
BOOTP-INVALID	0	0	0

Verify the DHCPv6 configuration.

```
Router# show running-configuration dhcp ipv6
```

Fri Feb 16 15:40:52.721 IST

```
dhcp ipv6
profile DHCPv6_RELAY relay
    helper-address vrf default 20::20
!
interface BVI1 relay profile DHCPv6_RELAY
!
```

Verify DHCPv6 relay profile.

```
Router# show dhcp ipv6 relay statistics
```

Fri Feb 16 15:41:00.456 IST

VRF	RX	TX	DR
default	4	4	0
**nVSatellite	0	0	0

Verify DHCPv6 relay packet statistics in detail.

```
Routerr# show dhcp ipv6 relay statistics vrf default
Fri Feb 16 15:41:09.991 IST
```

DHCP IPv6 Relay Statistics for VRF default:

TYPE	RECEIVE	TRANSMIT	DROP
SOLICIT	1	0	0
ADVERTISE	0	1	0
REQUEST	1	0	0
REPLY	0	1	0
CONFIRM	0	0	0
DECLINE	0	0	0
RENEW	0	0	0
REBIND	0	0	0
RELEASE	0	0	0
RECONFIG	0	0	0
INFORM	0	0	0
RELAY_FWD	0	0	0
RELAY REP	0	0	0
LEASEQUERY	0	0	0
LEASEQUERY REP	0	0	0
LEASEQUERY DONE	0	0	0
LEASEQUERY DATA	0	0	0

Duplicate IP Address Detection

The Duplicate IP Address Detection feature automatically detects any host with a duplicate IP address and blocks all MAC-IP routes that have a duplicate IP address.

This protects the network from hosts that are assigned duplicate IP addresses unintentionally or by malicious intent in an EVPN fabric. Hosts with duplicate IP address cause unnecessary churn in a network and causes traffic loss to either or both the hosts with the same IP address.

The system handles mobility of EVPN hosts by keeping track of MAC and IP addresses as they move from one host to another. If two hosts are assigned the same IP address, the IOS XR system keeps learning and re-learning MAC-IP routes from both the hosts. Each time it learns the MAC-IP route from one host, it is counted as one move since the newly learnt route supersedes the route previously learnt from the other host. This continues back and forth until the IP address is marked as duplicate based on the configured parameters.

It uses the following parameters to determine when an IP address should be marked as duplicate, and frozen or unfrozen as it moves between different hosts. The configurable parameters are:

- **move-interval**: The period within which a MAC or IP address has to move certain number of times between different hosts to be considered as duplicate and frozen temporarily. This number is specified in the **move-count** parameter.
- **move-count**: The number of times a MAC or IP address has to move within the interval specified for the **move-interval** parameter between different hosts to be considered a duplicate.
- **freeze-time**: The length of time a MAC or IP address is locked after it has been detected as a duplicate. After this period, the IP address is unlocked and it is allowed to learn again.

Configure Duplicate IP Address Detection

- **retry-count:** The number of times a MAC or IP address is unlocked after it has been detected as a duplicate before it is frozen permanently.

The system maintains a count of the number of times an IP address has been moved from one host to another host, either to another local host or to a host behind a remote Top of Rack (TOR). If an IP address moves certain number of times specified in the **move-count** parameter within the interval specified in the **move-interval** parameter is considered a duplicate IP address. All MAC-IP routes with that IP address is frozen for the time specified in the **freeze-time** parameter. A syslog notifies the user that the particular IP address is frozen. While an IP address is frozen, any new MAC-IP routes or updates to existing MAC-IP routes with the frozen IP address are ignored.

After **freeze-time** has elapsed, the corresponding MAC-IP routes are unfrozen and the value of the **move-count** is reset to zero. For any unfrozen local MAC-IP routes, an ARP probe and flush are initiated while the remote MAC-IP routes are put in the probe mode. This restarts the duplicate detection process.

The system also maintains the information about the number of times a particular IP address has been frozen and unfrozen. If an IP address is marked as duplicate after it is unfrozen **retry-count** times, it is frozen permanently until user manually unfreezes it. Use the following commands to manually unfreeze frozen MAC, IPv4 and IPv6 addresses respectively:

- **clear l2route evpn mac {mac-address} | all [evi evi] frozen-flag**
- **clear l2route evpn ipv4 {ipv4-address} | all [evi evi] frozen-flag**
- **clear l2route evpn ipv6 {ipv6-address} | all [evi evi] frozen-flag**

Configure Duplicate IP Address Detection

Perfrom these tasks to configure Duplicate IP Address Detection feature.

Configuration Example

```
/* Ipv4 Address Duplicate Detection Configuration */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# host ipv4-address duplicate-detection
RP/0/RSP0/CPU0:router(config-evpn-host-ipv4-addr)# move-count 2
RP/0/RSP0/CPU0:router(config-evpn-host-ipv4-addr)# freeze-time 10
RP/0/RSP0/CPU0:router(config-evpn-host-ipv4-addr)# retry-count 2
RP/0/RSP0/CPU0:router(config-evpn-host-ipv4-addr)# commit

/* Ipv6 Address Duplicate Detection Configuration */

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# evpn
RP/0/RSP0/CPU0:router(config-evpn)# host ipv6-address duplicate-detection
RP/0/RSP0/CPU0:router(config-evpn-host-ipv6-addr)# move-count 2
RP/0/RSP0/CPU0:router(config-evpn-host-ipv6-addr)# freeze-time 10
RP/0/RSP0/CPU0:router(config-evpn-host-ipv6-addr)# retry-count 2
RP/0/RSP0/CPU0:router(config-evpn-host-ipv6-addr)# commit
```

Running Configuration

This section shows the running configuration to detect duplicate IP address.

```

evpn
host ipv4-address duplicate-detection
move-count 2
freeze-time 10
retry-count 2
!
evpn
host ipv6-address duplicate-detection
move-count 2
freeze-time 10
retry-count 2
!
```

Verification

The show output given in the following section display the details of the duplicate IP address detection and recovery parameters.

```

RP/0/RSP0/CPU0:router#show l2route evpn mac-ip all detail

Flags: (Stt)=Static; (L)=Local; (R)=Remote; (F)=Flood;
      (N)=No Redistribution; (Rtr)=RP/0/RSP0/CPU0:router MAC; (B)=Best Route;
      (S)=Peer Sync; (Spl)=Split; (Rcv)=Recd;
      (D)=Duplicate MAC; (Z)=Frozen MAC;

          Topo ID      Mac Address      IP Address     Prod      Next Hop(s)      Seq No   Flags
          Opaque Data Type    Opaque Data Len   Opaque Data Value
-----  -----  -----  -----  -----
-----  -----  -----  -----  -----
33      0022.6730.0001  10.130.0.2  L2VPN  Bundle-Ether6.1300  0        SB 0 12
0x06000000
```

Related Topics

- [Duplicate IP Address Detection, on page 39](#)

Associated Commands

- `evpn host ipv4-address duplicate-detection`
- `evpn host ipv6-address duplicate-detection`
- `show l2route evpn mac-ip all detail`

EVPN E-Tree Using RT Constraints

The EVPN E-Tree using RT constraints feature enables you to configure BGP RT import and export policies for an attachment circuit. This feature allows you to define communication between the leaf and root nodes. The provider edge (PE) nodes can receive L2 traffic either from the attachment circuit (AC) of a bridge domain (BD) or from the remote PE node. For a given BD, L2 communication can only happen from root to leaf and leaf to root. This feature does not allow any L2 communication between the ACs of two or more leafs. This feature uses two BGP RTs for every EVI. Associate one RT with root ACs and the other with leaf ACs. For example, there are two distinct sets of RTs, one for root-rt and another for leaf-rt.

This feature provides you with the following benefits by performing filtering of unicast and multicast traffic at the ingress PE nodes:

- Achieve efficiency of the BGP MAC routes scale
- Reduce the consumption of hardware resources
- Utilize the link bandwidth efficiently

Rules for Import and Export Policies under the BGP of EVPN EVI Instances

- Root PE exports its ROOT-RT using BGP export policy. It also imports other ROOT-RT from the corresponding root PE for the same EVI. This is necessary where there is more than one root for a particular BD and EVPN EVI. For example, in a multihome active-active scenario or multihome port-active and single-active scenarios.
- Root PE imports LEAF-RT using BGP import policy for a EVPN EVI. This enables the root to be aware of all remote L2 MAC addresses through EVPN RT2 advertisement of leaf PE node for a given E-Tree EVI.
- Leaf PE exports its LEAF-RT using BGP export policy to let the root to be aware of the reachability of its directly connected L2 endpoints through EVPN RT2 advertisement.
- Leaf PE imports ROOT-RT using BGP import policy. It helps the leaf to know about the L2 endpoints which are reachable through the AC of BD under EVPN EVI instance of root PE. You must not import LEAF-RT using BGP Import policy to avoid L2 Communication between two leaf PEs.
- Use split-horizon filtering to block traffic among leaf ACs on a BD for a given E-Tree EVI.

The BGP import and export policies applies to all EVPN RTs along with the RT2 advertisement.

MAC Address Learning

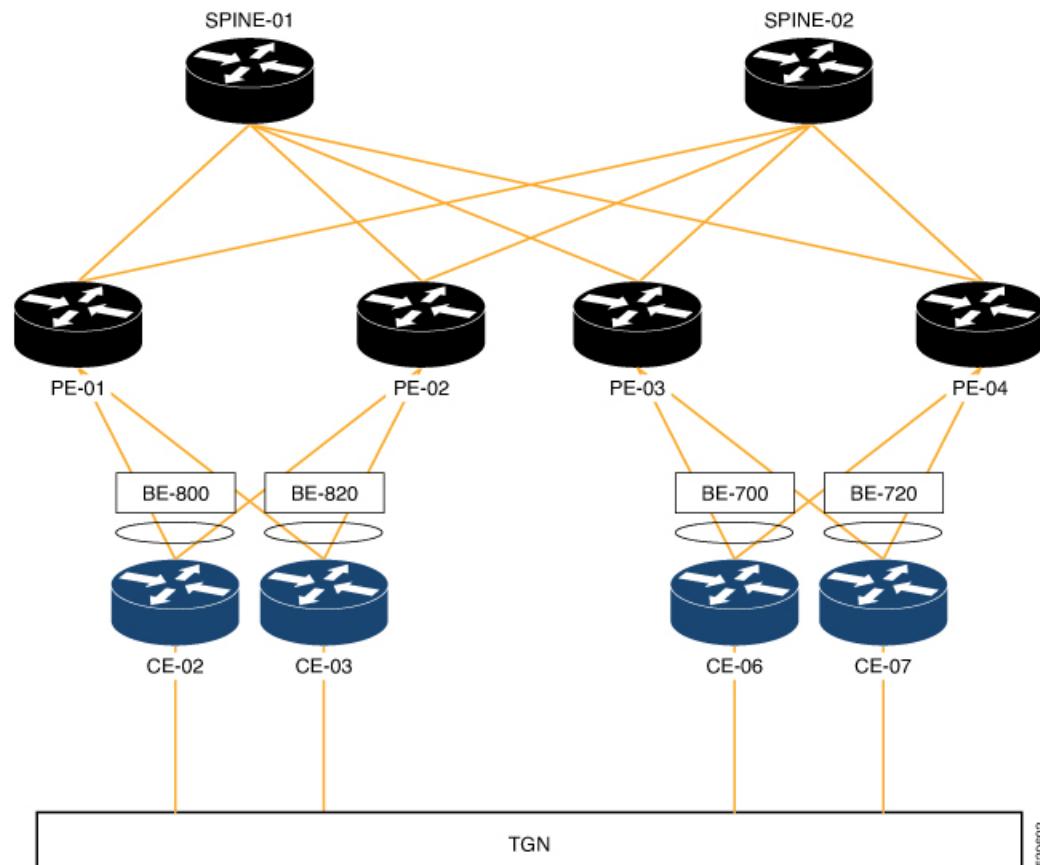
- L2 MAC addresses are learnt on AC of a particular BD on leaf PE as type LOCAL. The same MAC address is advertised to root PE as EVPN RT2. On the remote root PE, the MAC table replicates the entry of MAC address with the learn type as L2VPN. Also, it associates the MPLS label of its BGP peer, which advertises RT2 to root PE node.
- L2 MAC addresses are learnt on AC of a particular BD on the root as type LOCAL. The same MAC address is advertised to peer root (except for MH A/A) or leaf PE as EVPN RT2. On the remote root PE or leaf PE, the MAC table replicates the entry of MAC address with the learn type as L2VPN. Also, it associates the MPLS label of its BGP peer, which advertises RT2 to PE node.
- L2 MAC addresses are learnt on AC of a particular BD on the root as type LOCAL. The same MAC address is advertised to peer root for MH A/A as EVPN RT2. The MAC table of the peer root node synchronizes the replicated entry of MAC address with the learn type as L2VPN for same the ESI and with the same AC as the next hop. This avoids flooding and duplication of known unicast traffic.

The following scenario describes the feature topology:

CE with Multihoming Active-Active and CE with Multihoming Active-Active

Consider a topology where you connect CE-02 and CE-03 to PE-01 and PE-02. Both the CEs are in multihoming active-active mode. Connect CE-02 to PE-01 and PE-02 using AC BE-800.305. Connect CE-03 to PE-01 and

PE-02 using AC BE-820.305. Connect CE-06 and CE-07 to PE-03 and PE-04. Connect CE-06 to PE-03 and PE-04 using AC BE-700.305. Connect CE-07 to PE-03 and PE-04 using AC BE-720.305. Associate the bridge domain BD-305 with other AC on the respective PEs along with EVI-305 instance. Configure the respective RT on root and leaf with its import and export RTs for EVI-305. Configure PE-01 and PE-02 as root. Configure PE-03 and PE-04 as leaf.



Configuration

Perform the following tasks on PE-01, PE-02, PE-03, and PE-04.

- Configure bridge domain
- Configure attachment circuit
- Configure EVPN EVI
- Configure bundle Ethernet
- Configure EVPN interface



Note Use the **etree rt-leaf** command only if the leaf sites are in the EVPN all-active multihoming mode and not required for EVPN single homing mode.

Configuration Example

```

/* Configure PE-01 (as root) */

/* Configure bridge domain */
Router # configure
Router(config)# l2vpn
Router(config-l2vpn)# bridge group EVPN_BD
Router(config-l2vpn-bg)# bridge-domain evpn_bvi_305
Router(config-l2vpn-bg-bd)# interface Bundle-Ether800.305
Router(config-l2vpn-bg-bd-ac)# exit
Router (config-l2vpn-bg-bd)# interface Bundle-Ether820.305
Router(config-l2vpn-bg-bd-ac)# exit
Router (config-l2vpn-bg-bd)# evi 305
Router (config-l2vpn-bg-bd-evi)# commit

/* Configure attachment circuit */
Router# configure
Router(config)# interface Bundle-Ether800.305 12transport
Router(config-l2vpn-subif)# encapsulation dot1q 305
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

Router# configure
Router(config)# interface Bundle-Ether820.305 12transport
Router(config-l2vpn-subif)# encapsulation dot1q 305
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

/* Configure EVPN EVI */
Router# configure
Router(config)# evpn
Router(config-evpn)# evi 305
Router(config-evpn-instance)# bgp
Router(config-evpn-instance-bgp)# route-target import 1001:305
Router(config-evpn-instance-bgp)# route-target export 1001:305
Router(config-evpn-instance-bgp)# route-target import 1001:5305
Router(config-evpn-instance-bgp)# exit
Router(config-evpn-instance)# control-word-disable
Router(config-evpn-instance)# advertise-mac
Router(config-evpn-instance-mac)# commit

/* Configure bundle Ethernet */
Router# configure
Router(config)# interface Bundle-Ether800
Router(config-if)# lacp system mac 00aa.aabb.2020
Router(config-if)# lacp switchover suppress-flaps 300
Router(config-if)# lacp cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

Router# configure
Router(config)# interface Bundle-Ether820
Router(config-if)# lacp system mac 00aa.aabb.2222
Router(config-if)# lacp switchover suppress-flaps 300
Router(config-if)# lacp cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

/* Configure EVPN interface */
Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether800

```

```

Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.88.88.88.88.88.00
Router(config-evpn-ac-es)# bgp route-target 0001.0000.0001
Router(config-evpn-ac-es)# commit

Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether820
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.88.88.88.88.88.20
Router(config-evpn-ac-es)# bgp route-target 0001.0000.0020
Router(config-evpn-ac-es)# commit

/* Configure PE-02 (as root) */

/* Configure bridge domain */
Router# configure
Router(config)# l2vpn
Router(config-l2vpn)# bridge group EVPN_BD
Router(config-l2vpn-bg)# bridge-domain evpn_bvi_305
Router(config-l2vpn-bg-bd)# interface Bundle-Ether800.305
Router(config-l2vpn-bg-bd-ac)# exit
Router(config-l2vpn-bg-bd)# interface Bundle-Ether820.305
Router(config-l2vpn-bg-bd-ac)# exit
Router(config-l2vpn-bg-bd)# evi 305
Router(config-l2vpn-bg-bd-evi)# commit

/* Configure attachment circuit */
Router# configure
Router(config)# interface Bundle-Ether800.305 l2transport
Router(config-l2vpn-subif)# encapsulation dot1q 305
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

Router# configure
Router(config)# interface Bundle-Ether820.305 l2transport
Router(config-l2vpn-subif)# encapsulation dot1q 305
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

/* Configure EVPN EVI */
Router# configure
Router(config)# evpn
Router(config-evpn)# evi 305
Router(config-evpn-instance)# bgp
Router(config-evpn-instance-bgp)# route-target import 1001:305
Router(config-evpn-instance-bgp)# route-target export 1001:305
Router(config-evpn-instance-bgp)# route-target import 1001:5305
Router(config-evpn-instance-bgp)# exit
Router(config-evpn-instance)# control-word-disable
Router(config-evpn-instance)# advertise-mac
Router(config-evpn-instance-mac)# commit

/* Configure bundle Ethernet */
Router# configure
Router(config)# interface Bundle-Ether800
Router(config-if)# lacp system mac 00aa.aabb.2020
Router(config-if)# lacp switchover suppress-flaps 300
Router(config-if)# lacp cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

Router# configure

```

CE with Multihoming Active-Active and CE with Multihoming Active-Active

```

Router(config)# interface Bundle-Ether820
Router(config-if)# lACP system mac 00aa.aabb.2222
Router(config-if)# lACP switchover suppress-flaps 300
Router(config-if)# lACP cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

/* Configure EVPN interface */
Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether800
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.88.88.88.88.88.88.00
Router(config-evpn-ac-es)# bgp route-target 0001.0000.0001
Router(config-evpn-ac-es)# commit

Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether820
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.88.88.88.88.88.88.20
Router(config-evpn-ac-es)# bgp route-target 0001.0000.0020
Router(config-evpn-ac-es)# commit

/* Configure PE-03 (as leaf) */

/* Configure bridge domain */
Router# configure
Router(config)# l2vpn
Router(config-l2vpn)# bridge group EVPN_BD
Router(config-l2vpn-bg)# bridge-domain evpn_bvi_305
Router(config-l2vpn-bg-bd)# interface Bundle-Ether700.305
Router(config-l2vpn-bg-bd-ac)# split-horizon group
Router(config-l2vpn-bg-bd-ac)# exit
Router(config-l2vpn-bg-bd)# interface Bundle-Ether720.305
Router(config-l2vpn-bg-bd-ac)# split-horizon group
Router(config-l2vpn-bg-bd-ac)# exit
Router(config-l2vpn-bg-bd)# evi 305
Router(config-l2vpn-bg-bd-evi)# commit

/* Configure attachment circuit */
Router# configure
Router(config)# interface Bundle-Ether700.305 l2transport
Router(config-l2vpn-subif)# encapsulation dot1q 305
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

Router# configure
Router(config)# interface Bundle-Ether720.305 l2transport
Router(config-l2vpn-subif)# encapsulation dot1q 305
Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

/* Configure EVPN EVI */
Router# configure
Router(config)# evpn
Router(config-evpn)# evi 305
Router(config-evpn-instance)# bgp
Router(config-evpn-instance-bgp)# route-target import 1001:305
Router(config-evpn-instance-bgp)# route-target export 1001:5305
Router(config-evpn-instance-bgp)# exit
Router(config-evpn-instance)# etree
Router(config-evpn-instance-etree)# rt-leaf
Router(config-evpn-instance)# exit

```

```

Router(config-evpn-instance)# control-word-disable
Router(config-evpn-instance)# advertise-mac
Router(config-evpn-instance-mac)# commit

/* Configure bundle Ethernet */
Router# configure
Router(config)# interface Bundle-Ether700
Router(config-if)# lacp system mac 00aa.aabb.1010
Router(config-if)# lacp switchover suppress-flaps 300
Router(config-if)# lacp cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

Router# configure
Router(config)# interface Bundle-Ether720
Router(config-if)# lacp system mac 00aa.aabb.1212
Router(config-if)# lacp switchover suppress-flaps 300
Router(config-if)# lacp cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

/* Configure EVPN interface */
Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether700
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.77.77.77.77.77.77.00
Router(config-evpn-ac-es)# bgp route-target 0000.0000.0001
Router(config-evpn-ac-es)# commit

Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether720
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.77.77.77.77.77.77.20
Router(config-evpn-ac-es)# bgp route-target 0000.0000.0020
Router(config-evpn-ac-es)# commit

/* Configure PE-04 (as leaf) */

/* Configure bridge domain */
Router# configure
Router# 12vpn
Router(config-12vpn)# bridge group EVPN_BD
Router(config-12vpn-bg)# bridge-domain evpn_bvi_305
Router(config-12vpn-bg-bd)# interface Bundle-Ether700.305
Router(config-12vpn-bg-bd-ac)# split-horizon group
Router(config-12vpn-bg-bd-ac)# exit
Router(config-12vpn-bg-bd)# interface Bundle-Ether720.305
Router(config-12vpn-bg-bd-ac)# split-horizon group
Router(config-12vpn-bg-bd-ac)# exit
Router(config-12vpn-bg-bd-bd)# evi 305
Router(config-12vpn-bg-bd-evi)# commit

/* Configure attachment circuit */
Router# configure
Router(config)# interface Bundle-Ether700.305 12transport
Router(config-12vpn-subif)# encapsulation dot1q 305
Router(config-12vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-12vpn-subif)# commit

Router# configure
Router(config)# interface Bundle-Ether720.305 12transport
Router(config-12vpn-subif)# encapsulation dot1q 305

```

CE with Multihoming Active-Active and CE with Multihoming Active-Active

```

Router(config-l2vpn-subif)# rewrite ingress tag pop 1 symmetric
Router(config-l2vpn-subif)# commit

/* Configure EVPN EVI */
Router# configure
Router(config)# evpn
Router(config-evpn)# evi 305
Router(config-evpn-instance)# bgp
Router(config-evpn-instance-bgp)# route-target import 1001:305
Router(config-evpn-instance-bgp)# route-target export 1001:5305
Router(config-evpn-instance-bgp)# exit
Router(config-evpn-instance)# etree
Router(config-evpn-instance-etree)# rt-leaf
Router(config-evpn-instance)# exit
Router(config-evpn-instance)# control-word-disable
Router(config-evpn-instance)# advertise-mac
Router(config-evpn-instance-mac)# commit

/* Configure bundle Ethernet */
Router# configure
Router(config)# interface Bundle-Ether700
Router(config-if)# lACP system mac 00aa.aabb.1010
Router(config-if)# lACP switchover suppress-flaps 300
Router(config-if)# lACP cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

Router# configure
Router(config)# interface Bundle-Ether720
Router(config-if)# lACP system mac 00aa.aabb.1212
Router(config-if)# lACP switchover suppress-flaps 300
Router(config-if)# lACP cisco enable link-order signaled
Router(config-if)# bundle wait-while 100
Router(config-if)# commit

/* Configure EVPN interface */
Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether700
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.77.77.77.77.77.77.00
Router(config-evpn-ac-es)# bgp route-target 0000.0000.0001
Router(config-evpn-ac-es)# commit

Router(config)# evpn
Router(config-evpn)# interface Bundle-Ether720
Router(config-evpn-ac)# ethernet-segment
Router(config-evpn-ac-es)# identifier type 0 00.77.77.77.77.77.77.20
Router(config-evpn-ac-es)# bgp route-target 0000.0000.0020
Router(config-evpn-ac-es)# commit

```

Running Configuration

This section shows the PE-01, PE-02, PE-3, and PE-04 running configuration.

```

/* PE-01 Configuration */
l2vpn
bridge group EVPN_BD
bridge-domain evpn_bvi_305
interface Bundle-Ether800.305
!
interface Bundle-Ether820.305
!
```

```

evi 305
!
!
!
interface Bundle-Ether800.305 12transport
encapsulation dot1q 305
rewrite ingress tag pop 1 symmetric
!
interface Bundle-Ether820.305 12transport
encapsulation dot1q 305
rewrite ingress tag pop 1 symmetric
!
evpn
evi 305
bgp
  route-target import 1001:305
  route-target export 1001:305
  route-target import 1001:5305
!
control-word-disable
advertise-mac
!
!
!
interface Bundle-Ether800
lacp system mac 00aa.aabb.2020
lacp switchover suppress-flaps 300
lacp cisco enable link-order signaled
bundle wait-while 100
!
interface Bundle-Ether820
lacp system mac 00aa.aabb.2222
lacp switchover suppress-flaps 300
lacp cisco enable link-order signaled
bundle wait-while 100
!
evpn
  interface Bundle-Ether800
    ethernet-segment
      identifier type 0 00.88.88.88.88.88.88.00
      bgp route-target 0001.0000.0001
    !
  !
  !
  evpn
    interface Bundle-Ether820
      ethernet-segment
        identifier type 0 00.88.88.88.88.88.88.20
        bgp route-target 0001.0000.0020
    !
  !
  !
/* PE-02 Configuration */
12vpn
bridge group EVPN_BD
bridge-domain evpn_bvi_305
  interface Bundle-Ether800.305
  !
  interface Bundle-Ether820.305
  !
  evi 305
  !
!
```

CE with Multihoming Active-Active and CE with Multihoming Active-Active

```

interface Bundle-Ether800.305 12transport
encapsulation dot1q 305
rewrite ingress tag pop 1 symmetric
!
interface Bundle-Ether820.305 12transport
encapsulation dot1q 305
rewrite ingress tag pop 1 symmetric
!
evpn
evi 305
bgp
  route-target import 1001:305
  route-target export 1001:305
  route-target import 1001:5305
!
control-word-disable
advertise-mac
!
!
interface Bundle-Ether800
lacp system mac 00aa.aabb.2020
lacp switchover suppress-flaps 300
lacp cisco enable link-order signaled
bundle wait-while 100
!
interface Bundle-Ether820
lacp system mac 00aa.aabb.2222
lacp switchover suppress-flaps 300
lacp cisco enable link-order signaled
bundle wait-while 100
!
evpn
interface Bundle-Ether800
  ethernet-segment
    identifier type 0 00.88.88.88.88.88.88.00
    bgp route-target 0001.0000.0001
  !
  !
evpn
interface Bundle-Ether820
  ethernet-segment
    identifier type 0 00.88.88.88.88.88.88.20
    bgp route-target 0001.0000.0020
  !
  !

/* PE-03 Configuration */
12vpn
bridge group EVPN_BD
bridge-domain evpn_bvi_305
  interface Bundle-Ether700.305
    split-horizon group
  !
  interface Bundle-Ether720.305
    split-horizon group
  !
  evi 305
  !
  !
  !
  interface Bundle-Ether700.305 12transport
  encapsulation dot1q 305
  rewrite ingress tag pop 1 symmetric
!

```

```

interface Bundle-Ether720.305 12transport
encapsulation dot1q 305
rewrite ingress tag pop 1 symmetric
!
evpn
evi 305
bgp
route-target import 1001:305
route-target export 1001:5305
!
etree
rt-leaf
!
control-word-disable
advertise-mac
!
!
!
interface Bundle-Ether700
lacp system mac 00aa.aabb.1010
lacp switchover suppress-flaps 300
lacp cisco enable link-order signaled
bundle wait-while 100
!
interface Bundle-Ether720
lacp system mac 00aa.aabb.1212
lacp switchover suppress-flaps 300
lacp cisco enable link-order signaled
bundle wait-while 100
!
evpn
interface Bundle-Ether700
ethernet-segment
identifier type 0 00.77.77.77.77.77.77.00
bgp route-target 0000.0000.0001
!
!
!
evpn
interface Bundle-Ether720
ethernet-segment
identifier type 0 00.77.77.77.77.77.77.20
bgp route-target 0000.0000.0020
!
!
!
```

/* PE-04 Configuration */

```

12vpn
bridge group EVPN_BD
bridge-domain evpn_bvi_305
interface Bundle-Ether700.305
split-horizon group
!
interface Bundle-Ether720.305
split-horizon group
!
evi 305
!
!
!
interface Bundle-Ether700.305 12transport
encapsulation dot1q 305
rewrite ingress tag pop 1 symmetric

```

CE with Multihoming Active-Active and CE with Multihoming Active-Active

```

!
interface Bundle-Ether720.305 12transport
  encapsulation dot1q 305
  rewrite ingress tag pop 1 symmetric
!
evpn
  evi 305
    bgp
      route-target import 1001:305
      route-target export 1001:5305
    !
    etree
      rt-leaf
    !
    control-word-disable
    advertise-mac
  !
!
!
interface Bundle-Ether700
  lacp system mac 00aa.aabb.1010
  lacp switchover suppress-flaps 300
  lacp cisco enable link-order signaled
  bundle wait-while 100
!
interface Bundle-Ether720
  lacp system mac 00aa.aabb.1212
  lacp switchover suppress-flaps 300
  lacp cisco enable link-order signaled
  bundle wait-while 100
!
evpn
  interface Bundle-Ether700
    ethernet-segment
      identifier type 0 00.77.77.77.77.77.77.00
      bgp route-target 0000.0000.0001
    !
  !
  !
  evpn
    interface Bundle-Ether720
      ethernet-segment
        identifier type 0 00.77.77.77.77.77.77.20
        bgp route-target 0000.0000.0020
      !
    !
!
```

Verification

This section shows how the L2 MAC addresses are synchronized as LOCAL and L2VPN with multihoming active-active peers PE. Also, the root PE is aware of MAC addresses learnt on leaf PE remotely through RT2 advertisements.

Router:PE-01# show l2route evpn mac all				
Topo ID	Mac Address	Producer	Next Hop(s)	
204	001f.0100.0001	LOCAL	Bundle-Ether820.305, N/A	
204	001f.0100.0001	L2VPN	Bundle-Ether820.305, N/A	
204	001f.0100.0002	LOCAL	Bundle-Ether820.305, N/A	
204	001f.0100.0002	L2VPN	Bundle-Ether820.305, N/A	
204	001f.0100.0003	LOCAL	Bundle-Ether820.305, N/A	

```

204      001f.0100.0003 L2VPN      Bundle-Ether820.305, N/A
204      001f.0100.0004 LOCAL      Bundle-Ether820.305, N/A
204      001f.0100.0004 L2VPN      Bundle-Ether820.305, N/A
204      001f.0100.0005 LOCAL      Bundle-Ether820.305, N/A
204      001f.0100.0005 L2VPN      Bundle-Ether820.305, N/A
204      0020.0100.0001 L2VPN      26791/I/ME, N/A
204      0020.0100.0002 L2VPN      26791/I/ME, N/A
204      0020.0100.0003 L2VPN      26791/I/ME, N/A
204      0020.0100.0004 L2VPN      26791/I/ME, N/A
204      0020.0100.0005 L2VPN      26791/I/ME, N/A
204      0021.0100.0001 L2VPN      Bundle-Ether800.305, N/A
204      0021.0100.0002 L2VPN      Bundle-Ether800.305, N/A
204      0021.0100.0003 LOCAL      Bundle-Ether800.305, N/A
204      0021.0100.0004 L2VPN      Bundle-Ether800.305, N/A
204      0021.0100.0005 LOCAL      Bundle-Ether800.305, N/A
204      0022.0100.0001 L2VPN      26790/I/ME, N/A
204      0022.0100.0002 L2VPN      26790/I/ME, N/A
204      0022.0100.0003 L2VPN      26790/I/ME, N/A
204      0022.0100.0004 L2VPN      26790/I/ME, N/A
204      0022.0100.0005 L2VPN      26790/I/ME, N/A

```

Router:PE-02# show l2route evpn mac all

Topo ID	Mac Address	Producer	Next Hop(s)
204	001f.0100.0001	LOCAL	Bundle-Ether820.305, N/A
204	001f.0100.0001	L2VPN	Bundle-Ether820.305, N/A
204	001f.0100.0002	LOCAL	Bundle-Ether820.305, N/A
204	001f.0100.0002	L2VPN	Bundle-Ether820.305, N/A
204	001f.0100.0003	LOCAL	Bundle-Ether820.305, N/A
204	001f.0100.0003	L2VPN	Bundle-Ether820.305, N/A
204	001f.0100.0004	LOCAL	Bundle-Ether820.305, N/A
204	001f.0100.0004	L2VPN	Bundle-Ether820.305, N/A
204	001f.0100.0005	LOCAL	Bundle-Ether820.305, N/A
204	001f.0100.0005	L2VPN	Bundle-Ether820.305, N/A
204	0020.0100.0001	L2VPN	27367/I/ME, N/A
204	0020.0100.0002	L2VPN	27367/I/ME, N/A
204	0020.0100.0003	L2VPN	27367/I/ME, N/A
204	0020.0100.0004	L2VPN	27367/I/ME, N/A
204	0020.0100.0005	L2VPN	27367/I/ME, N/A
204	0021.0100.0001	LOCAL	Bundle-Ether800.305, N/A
204	0021.0100.0002	LOCAL	Bundle-Ether800.305, N/A
204	0021.0100.0003	L2VPN	Bundle-Ether800.305, N/A
204	0021.0100.0004	LOCAL	Bundle-Ether800.305, N/A
204	0021.0100.0005	L2VPN	Bundle-Ether800.305, N/A
204	0022.0100.0001	L2VPN	27366/I/ME, N/A
204	0022.0100.0002	L2VPN	27366/I/ME, N/A
204	0022.0100.0003	L2VPN	27366/I/ME, N/A
204	0022.0100.0004	L2VPN	27366/I/ME, N/A
204	0022.0100.0005	L2VPN	27366/I/ME, N/A

The following output shows how the multihoming PE is aware of its local L2 MAC addresses as well as the MAC addresses learnt on the root node only. Leaf multihoming PE is not aware of any other MAC addresses learnt on other leaf PE nodes except if they are learnt on a multihoming active-active ethernet-segment on the peer leaf PE.

Router:PE-03# show l2route evpn mac all

Topo ID	Mac Address	Producer	Next Hop(s)
200	0011.0100.0003	L2VPN	30579/I/ME, N/A
200	0011.0100.0005	L2VPN	30579/I/ME, N/A
204	001f.0100.0001	L2VPN	30588/I/ME, N/A
204	001f.0100.0002	L2VPN	30588/I/ME, N/A
204	001f.0100.0003	L2VPN	30588/I/ME, N/A

CE with Multihoming Active-Active and CE with Multihoming Active-Active

```

204      001f.0100.0004 L2VPN    30588/I/ME, N/A
204      001f.0100.0005 L2VPN    30588/I/ME, N/A
204      0020.0100.0001 LOCAL   Bundle-Ether720.305, N/A
204      0020.0100.0001 L2VPN   Bundle-Ether720.305, N/A
204      0020.0100.0002 LOCAL   Bundle-Ether720.305, N/A
204      0020.0100.0002 L2VPN   Bundle-Ether720.305, N/A
204      0020.0100.0003 LOCAL   Bundle-Ether720.305, N/A
204      0020.0100.0003 L2VPN   Bundle-Ether720.305, N/A
204      0020.0100.0004 LOCAL   Bundle-Ether720.305, N/A
204      0020.0100.0004 L2VPN   Bundle-Ether720.305, N/A
204      0020.0100.0005 LOCAL   Bundle-Ether720.305, N/A
204      0021.0100.0001 L2VPN   30587/I/ME, N/A
204      0021.0100.0002 L2VPN   30587/I/ME, N/A
204      0021.0100.0003 L2VPN   30587/I/ME, N/A
204      0021.0100.0004 L2VPN   30587/I/ME, N/A
204      0021.0100.0005 L2VPN   30587/I/ME, N/A
204      0022.0100.0001 LOCAL   Bundle-Ether700.305, N/A
204      0022.0100.0001 L2VPN   Bundle-Ether700.305, N/A
204      0022.0100.0002 LOCAL   Bundle-Ether700.305, N/A
204      0022.0100.0002 L2VPN   Bundle-Ether700.305, N/A
204      0022.0100.0003 LOCAL   Bundle-Ether700.305, N/A
204      0022.0100.0003 L2VPN   Bundle-Ether700.305, N/A
204      0022.0100.0004 LOCAL   Bundle-Ether700.305, N/A
204      0022.0100.0004 L2VPN   Bundle-Ether700.305, N/A
204      0022.0100.0005 LOCAL   Bundle-Ether700.305, N/A
204      0022.0100.0005 L2VPN   Bundle-Ether700.305, N/A

```

Router:PE-04# show l2route evpn mac all

Topo ID	Mac Address	Producer	Next Hop(s)
200	0011.0100.0003	L2VPN	30545/I/ME, N/A
200	0011.0100.0005	L2VPN	30545/I/ME, N/A
204	001f.0100.0001	L2VPN	30550/I/ME, N/A
204	001f.0100.0002	L2VPN	30550/I/ME, N/A
204	001f.0100.0003	L2VPN	30550/I/ME, N/A
204	001f.0100.0004	L2VPN	30550/I/ME, N/A
204	001f.0100.0005	L2VPN	30550/I/ME, N/A
204	0020.0100.0001	LOCAL	Bundle-Ether720.305, N/A
204	0020.0100.0001	L2VPN	Bundle-Ether720.305, N/A
204	0020.0100.0002	LOCAL	Bundle-Ether720.305, N/A
204	0020.0100.0002	L2VPN	Bundle-Ether720.305, N/A
204	0020.0100.0003	LOCAL	Bundle-Ether720.305, N/A
204	0020.0100.0003	L2VPN	Bundle-Ether720.305, N/A
204	0020.0100.0004	LOCAL	Bundle-Ether720.305, N/A
204	0020.0100.0004	L2VPN	Bundle-Ether720.305, N/A
204	0020.0100.0005	LOCAL	Bundle-Ether720.305, N/A
204	0020.0100.0005	L2VPN	Bundle-Ether720.305, N/A
204	0021.0100.0001	L2VPN	30549/I/ME, N/A
204	0021.0100.0002	L2VPN	30549/I/ME, N/A
204	0021.0100.0003	L2VPN	30549/I/ME, N/A
204	0021.0100.0004	L2VPN	30549/I/ME, N/A
204	0021.0100.0005	L2VPN	30549/I/ME, N/A
204	0022.0100.0001	LOCAL	Bundle-Ether700.305, N/A
204	0022.0100.0001	L2VPN	Bundle-Ether700.305, N/A
204	0022.0100.0002	LOCAL	Bundle-Ether700.305, N/A
204	0022.0100.0002	L2VPN	Bundle-Ether700.305, N/A
204	0022.0100.0003	LOCAL	Bundle-Ether700.305, N/A
204	0022.0100.0003	L2VPN	Bundle-Ether700.305, N/A
204	0022.0100.0004	LOCAL	Bundle-Ether700.305, N/A
204	0022.0100.0004	L2VPN	Bundle-Ether700.305, N/A
204	0022.0100.0005	LOCAL	Bundle-Ether700.305, N/A
204	0022.0100.0005	L2VPN	Bundle-Ether700.305, N/A

Related Topics

- [EVPN E-Tree Using RT Constraints, on page 41](#)

Associated Commands

- `etree rt-leaf`
- `show l2route evpn mac all`

CE with Multihoming Active-Active and CE with Multihoming Active-Active