EVPN Single-Homing

The EVPN Single-Homing feature utilizes the functionality defined in RFC 7432 (BGP MPLS-based Ethernet VPN), to achieve single-homing between a Provider Edge (PE) and a Customer Edge (CE) device.

- Information about EVPN Single-Homing, on page 1
- Prerequisites for EVPN Single-Homing, on page 5
- Restrictions for EVPN Single-Homing, on page 5
- How to Configure EVPN Single Homing, on page 6
- Configuration Examples for EVPN Single-Homing, on page 9
- Additional References for EVPN Single-Homing, on page 14
- Feature Information for EVPN Single-Homing, on page 14

Information about EVPN Single-Homing

Ethernet Multipoint Connectivity

To achieve Ethernet multipoint connectivity, MPLS deployments traditionally rely on Virtual Private LAN Services (VPLS). A VPLS service is built with a full-mesh of pseudowires between PE devices which are part of a Layer 2 broadcast domain. A VPLS PE device performs data-plane MAC learning. For MAC learning, the VPLS PE device uses local interfaces for traffic coming from the access network and uses pseudowires for the traffic coming from the core network.

EVPN Multipoint Solution

EVPN is the next generation of multipoint L2VPN solution that aligns operation principles of L3VPN with Ethernet services. Instead of relying solely on data plane for MAC Address learning, EVPN PE devices signal and learn MAC addresses over the core network using BGP, while still using data plane MAC-learning on the access side. Providers can configure BGP as a common VPN control plane for their ethernet offerings and leverage the advantages of Layer 3 VPN over VPLS. In Cisco IOS XE Fuji 16.8.1, only Single Homing functionality is supported from the feature set defined in RFC 7432.

EVPN Building Blocks

There are three fundamental building blocks for EVPN technology, EVPN Instance (EVI), Ethernet Segment (ES), EVPN BGP routes and extended communities:
• EVI is a VPN connection on a PE router. It is the equivalent of IP VPN Routing and Forwarding (VRF) in Layer 3 VPN. It is also known as MAC-VRF.

• ES is a connection with a customer site (device or network) and is associated with access-facing interfaces. They are assigned a unique ID that is referred to as Ethernet Segment Identifier (ESI). A site can be connected to one or more PEs. The ES connection has the same ESI in each of the PEs connected to the site.

• RFC7432 defines four new routes and four new extended communities to enable EPVN support. In Cisco IOS XE Fuji 16.8.x Software Release, Route Type 2 and Route Type 3 are supported.

In BGP MPLS-based EVPN, an EVI is configured for every PE device for each customer associated with the PE device. An example of a customer is the Customer Edge device that is attached to the PE device. Each EVI has a unique Route Distinguisher (RD) and one or more Route Targets (RT). The CE device can be a host, a switch or a router.

For EVPN Single-Homing feature, a CE device is attached to a single PE device and has an Ethernet Segment with ESI=0.

### Service Interfaces

The following types of EVPN VLAN service interfaces:

**VLAN-based Service Interface**

In VLAN-based service interface, each VLAN is associated to one bridge domain and one EVI.

*Figure 1: VLAN-Based Service Interface*

For VLAN-based Service Interface, Type 1 Route Distinguisher, a unique number used to distinguish identical routes in different VRFs, is used for EVIs as recommended by the RFC7432. The Route Distinguishers and Router Targets, which are used to share routes between different VRFs, are autogenerated to ensure unique Route Distinguisher numbers across EVIs.

**VLAN Bundle Service Interface**

In VLAN Bundle Service Interface, multiple VLANs share the same bridge table.
Each EVPN instance corresponds to multiple broadcast domains maintained in a single bridge table per MAC-VRF. For VLAN Bundle Service Interface service to work, MAC addresses must be unique across all VLANs for an EVI.

**VLAN-Aware Bundle Service Interface**

For VLAN-aware Bundle Service Interface, each VLAN is associated with one bridge domain, but there can be multiple bridge domains associated with one EVI.

An EVPN instance consists of multiple broadcast domains where each VLAN has one bridge table. Multiple bridge tables (one per VLAN) are maintained by a single MAC-VRF that corresponds to the EVPN instance.

**Route Types**

For EVPN Single homing feature, Route Type 2 and Route Type 3 are supported, as defined by RFC 7432.

**Route Type 2 - MAC and IP Advertisement Route**

Type 2 Routes are used to advertise MAC addresses and their associated IP addresses. When a PE router learns the MAC address of a CE device that is connected to it locally, or a MAC address of a device behind the CE device, a MAC and IP advertisement route is created.

Following is the header format for the MAC and IP Advertisement Route packet:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Length (Octets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route Type</td>
<td>0x02</td>
<td>1</td>
</tr>
<tr>
<td>Length</td>
<td>Variable</td>
<td>1</td>
</tr>
<tr>
<td>EVI RD</td>
<td>Type 1 (IPv4 address) RD unique across all EVIs on the PE.</td>
<td>8</td>
</tr>
<tr>
<td>ESI</td>
<td>Ethernet Segment Identifier</td>
<td>10</td>
</tr>
<tr>
<td>Ethernet Tag</td>
<td>0 or valid Ethernet Tag</td>
<td>4</td>
</tr>
</tbody>
</table>
**Route Types**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Length (Octets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC Addr Len</td>
<td>48</td>
<td>1</td>
</tr>
<tr>
<td>MAC Address</td>
<td>Valid MAC address</td>
<td>6</td>
</tr>
<tr>
<td>IP Addr Length</td>
<td>IP address length in bits: 0 or 32 or 128</td>
<td>1</td>
</tr>
<tr>
<td>IP Address</td>
<td>Optional IP address</td>
<td>0 or 4 or 16</td>
</tr>
<tr>
<td>Label1</td>
<td>Valid downstream assigned label to perform forwarding to CE based on the destination MAC address</td>
<td>3</td>
</tr>
<tr>
<td>Label2</td>
<td>Specifies a second label</td>
<td>0-3</td>
</tr>
<tr>
<td>EVI RT</td>
<td>Type 0 (2byteAS) route target</td>
<td>8</td>
</tr>
<tr>
<td>MAC Mobility</td>
<td>0x0600: {1 byte Sticky bit} :0x00: {4 byte sequence number}</td>
<td>8</td>
</tr>
</tbody>
</table>

- MAC Address field is populated with the CE address.
- IP address field is optional with IP Address length set to 0 bits.
- For EVPN Single-Homing feature, ESI value is always set to Zero.
- In the Label field, Per-BD or Per-CE labels can be assigned.
  - Per-BD is used when PE advertises a single label for all MAC addresses learned in a given bridge domain.
  - Per-CE label assigns a separate label to each access port in the bridge domain.

**Route Type 3 - Inclusive Multicast Ethernet Tag Route**

Type 3 routes are used for transporting Broadcast, Unknown Unicast and Multicast (BUM) traffic to other PE devices across a given EVPN network instance.

The following is the header format for Type 3 routes:

**Table 2: Route Type 3 - Inclusive Multicast Ethernet Tag Route Header**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Length (Octets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route Type</td>
<td>0x03</td>
<td>1</td>
</tr>
<tr>
<td>Length</td>
<td>26 or 38</td>
<td>1</td>
</tr>
<tr>
<td>EVI RD</td>
<td>Type 1 (IPv4Addr) RD unique across all EVIs on the PE.</td>
<td>8</td>
</tr>
<tr>
<td>Ethernet Tag</td>
<td>0 or valid Ethernet Tag</td>
<td>4</td>
</tr>
</tbody>
</table>
### Prerequisites for EVPN Single-Homing

- EVI and Bridge domains must be in established state with associated MPLS labels.

### Restrictions for EVPN Single-Homing

- Route Type 1 and Route Type 4 are not supported.
- Per-EVI-based labelling in not supported.
- The number of bridge domains that are supported are 16000.
- The number of EFPs or service instances that are supported per physical interface are 8000.
- Stateful Switchover is not supported.
- Single-Homing feature is not supported with port channel interface between Provider Edge and Customer Edge devices.
- MAC mobility with duplicate MAC detection is not supported.
How to Configure EVPN Single Homing

Configuring EVPN

The above figure represents a simple EVPN network. Use the following steps to configure EVPN:

EVPN Configuration

```plaintext
enable
configure terminal
l2vpn evpn
    replication-type ingress
    router-id Loopback1
    mpls label mode per-ce

l2vpn evpn instance 10 vlan-based
    route-distinguisher 1.1.1.1:10
    route-target both 10:10
    no auto-route-target

member evpn-instance 10
    member GigabitEthernet 0/0/1 service-instance 10

member GigabitEthernet 0/0/1 service-instance 10
    no ip address
    service instance 10 ethernet
    encapsulation dot1q 200
```
Configuring L2VPN EVPN Globally and EVI on IOS-XE Router

l2vpn evpn
  replication-type ingress  ----> Enables ingress replication label
l2vpn evpn instance 10 vlan-based ----> Configures Vlan-based EVI 10
l2vpn evpn instance 20 vlan-bundle ----> Configures Vlan-bundled EVI 20
l2vpn evpn instance 30 vlan-aware ----> Configures Vlan-aware EVI 30

Configuring Bridge Domains on IOS-XE Router

bridge-domain 10
  mac aging-time 30
  member GigabitEthernet6 service-instance 10 ----> Links SI 10 on interface with Bridge-domain 10
  member evpn-instance 10 ----> Links EVI 10 with Bridge-domain 10
bridge-domain 20
  mac aging-time 30
  member GigabitEthernet6 service-instance 20 ----> Links SI 20 on interface with Bridge-domain 20
  member evpn-instance 20 ----> Links EVI 20 with Bridge-domain 20
bridge-domain 30
  mac aging-time 30
  member GigabitEthernet6 service-instance 30 ----> Links SI 30 on interface with Bridge-domain 30
  member evpn-instance 30 ethernet-tag 30 ----> Links EVI 30 with Bridge-domain 30

Configuring Access Interface on a Provider Edge

interface GigabitEthernet6
  no ip address
  negotiation auto
  service instance 10 ethernet ----> Enables service instance 10 under the physical interface
    encapsulation dot1q 10
  service instance 20 ethernet ----> Enables service instance 20 under the physical interface
    encapsulation dot1q 20-21
  service instance 30 ethernet ----> Enables service instance 30 under the physical interface
    encapsulation dot1q 30

Configuring EVPN Single-Homing

Use the following steps to configure EVPN Single-Homing:

Configuring BGP on Provider Edge Device, PE1

enable
configure terminal
Configuring BGP on Route Reflector

```plaintext
router bgp 100
  bgp router-id 10.1.1.1
  bgp log-neighbor-changes
  bgp graceful-restart
  neighbor 10.2.2.2 remote-as 100
  neighbor 10.2.2.2 update-source Loopback0
!
address-family ipv4
  neighbor 10.2.2.2 activate
  neighbor 10.2.2.2 send-community both
  neighbor 10.2.2.2 soft-reconfiguration inbound
exit-address-family

address-family l2vpn evpn
  neighbor 10.2.2.2 activate
  neighbor 10.2.2.2 send-community both
  neighbor 10.2.2.2 soft-reconfiguration inbound
exit-address-family
```

Configuring Customer Edge and Provider Edge Interfaces

CE1 configuration

```plaintext
interface GigabitEthernet6.10
  encapsulation dot1Q 10
  ip address 203.0.113.1 255.255.255.240

interface GigabitEthernet6.20
  encapsulation dot1Q 20
  ip address 203.0.113.17 255.255.255.240
```
interface GigabitEthernet6.30
  encapsulation dot1Q 30
  ip address 203.0.113.33 255.255.255.240

PE1 Configuration

interface GigabitEthernet6
  no ip address
  negotiation auto
  service instance 10 ethernet
  encapsulation dot1q 10

  service instance 20 ethernet
  encapsulation dot1q 20-21

  service instance 30 ethernet
  encapsulation dot1q 30

Configuration Examples for EVPN Single-Homing

Use the following command to verify that EVI and Bridge domains are in established state and to display associated MPLS labels:

```
show 12vpn evpn evi detail
```

**EVPN instance: 10 (VLAN Based) ----> VLAN Based EVI**
- RD: 10.1.1.1:10 (auto) ----> RD derived from Loopback0 of PE1
- Import-RTs: 100:10
- Export-RTs: 100:10
- Per-EVI Label: none
- State: Established ----> EVI state
- Encapsulation: mpls
- Bridge Domain: 10
- Ethernet-Tag: 0
- BUM Label: 23 ----> Broadcast/Unknown unicast/Multicast traffic label
- Per-BD Label: 22
- State: Established ----> Bridge-domain state
- Pseudoports:
  - GigabitEthernet6 service instance 10 ----> Local interface part of bridge-domain
  - GigabitEthernet7 service instance 10 ----> Local interface part of bridge-domain

**EVPN instance: 20 (VLAN Bundle) ----> VLAN Bundled EVI**
- RD: 10.1.1.1:20 (auto)
- Import-RTs: 100:20
- Export-RTs: 100:20
- Per-EVI Label: none
- State: Established
- Encapsulation: mpls
- Bridge Domain: 20
- Ethernet-Tag: 0
- BUM Label: 20
- Per-BD Label: 21
- State: Established
- Pseudoports:
  - GigabitEthernet6 service instance 20
  - GigabitEthernet7 service instance 20

**EVPN instance: 30 (VLAN Aware) ----> VLAN-Aware EVI**
- RD: 10.1.1.1:30 (auto)
- Import-RTs: 100:30
- Export-RTs: 100:30
Per-EVI Label: none
State: Established
Encapsulation: mpls
Bridge Domain: 30
Ethernet-Tag: 30
BUM Label: 18
Per-BD Label: 19
State: Established
Pseudoports:
  GigabitEthernet6 service instance 30
  GigabitEthernet7 service instance 30

Use the following command to verify that bridge domain has learnt the local MAC address:

PE1# show bridge-domain 10
Bridge-domain 10 (3 ports in all)
State: UP
Aging-Timer: 30 second(s) ----> MAC aging timer for bridge-domain
  GigabitEthernet6 service instance 10
  GigabitEthernet7 service instance 10
EVPN Instance 10
AED MAC address Policy Tag Age Pseudoport
  000C.29B0.3E16 forward static_r 0 OCE_PTR:0xe8eb04a0 ----> Remotely learnt MAC
  000C.29AF.F904 forward dynamic_c 29 GigabitEthernet6.EFP10 ----> MAC locally learnt
  000C.2993.130E forward dynamic_c 26 GigabitEthernet7.EFP10
  000C.29EE.EC0D forward static_r 0 OCE_PTR:0xe8eb0500

In the above output, MAC addresses with forward dynamic_c tags are locally learned addresses and MAC addresses with forward static_r tags are remote addresses learned through EVPN.

Use the following command to verify that EVPN manager has received the local MACs learned by the bridge domain:

PE1# show l2vpn evpn mac
MAC Address  EVI  BD  ESI Ether Tag Next Hop
------------- ----- ----- ------------------------ ---------- -------------
000C.2993.130e 10 10 0000.0000.0000.0000.0000 0 Gi7:10
000C.29AF.F904 10 10 0000.0000.0000.0000.0000 0 Gi6:10
000C.29b0.3E16 10 10 0000.0000.0000.0000.0000 0 10.7.7.7
000C.29ee.EC0D 10 10 0000.0000.0000.0000.0000 0 10.3.3.3

PE1# show l2vpn evpn mac detail
MAC Address: 000C.2993.130e
EVPN Instance: 10
Bridge Domain: 10
Ethernet Segment: 0000.0000.0000.0000.0000
Ethernet Tag ID: 0
Next Hop(s): GigabitEthernet7 service instance 10
Label: 22
Sequence Number: 0
MAC only present: Yes
MAC Duplication Detection: Timer not running

MAC Address: 000C.29ee.EC0D
EVPN Instance: 10
Bridge Domain: 10
Ethernet Segment: 0000.0000.0000.0000.0000
Ethernet Tag ID: 0
Next Hop(s): 10.3.3.3
Local Address: 10.1.1.1
Label: 19
Sequence Number: 0
MAC only present: Yes
MAC Duplication Detection: Timer not running

In the above output, the next hop address of the remote MAC is the address of the provider edge device, if it is learnt remotely or the local interface if MAC address is learnt locally.

Use the following command to verify that Layer 2 Routing Information Base (RIB) has the required MAC info:

```bash
PE1# show l2vpn l2route evpn mac
```

<table>
<thead>
<tr>
<th>EVI</th>
<th>ETag</th>
<th>Prod</th>
<th>Mac Address</th>
<th>Next Hop(s)</th>
<th>Seq Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0</td>
<td>L2VPN</td>
<td>000C.2993.130E</td>
<td>Gi7:10</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>L2VPN</td>
<td>000C.29AF.F904</td>
<td>Gi6:10</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>BGP</td>
<td>000C.29B0.3E16</td>
<td>L:19 IP:10.7.7.7</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>BGP</td>
<td>000C.29EE.EC0D</td>
<td>L:19 IP:10.3.3.3</td>
<td>0</td>
</tr>
</tbody>
</table>

Remote MACs are learnt through BGP. In the above command output, the producer is BGP and local MACs are learned through Layer 2 VPN.

Use the following command to verify that Layer 2 FIB has received the MAC information from Layer 2 RIB, and bridge-domain and MFI are configured.

```bash
PE1# show l2fib bridge-domain 10 detail
```

Bridge Domain : 10
Reference Count : 18
Replication ports count : 4
Unicast Address table size : 4
IP Multicast Prefix table size : 4

Flood List Information :
Olist: Id 9225, Port Count 4

Port Information :
Serv Inst: Gi6:10
Serv Inst: Gi7:10
EVPN MPLS Encap: pathlist 107
EVPN MPLS Encap: pathlist 101

Unicast Address table information :
Mac: 000c.2993.130e, Adjacency: Serv Inst: Gi7:10
Mac: 000c.29af.f904, Adjacency: Serv Inst: Gi6:10
Mac: 000c.29b0.3e16, Adjacency: EVPN MPLS Encap: pathlist 98
Mac: 000c.29ee.ec0d, Adjacency: EVPN MPLS Encap: pathlist 104

IP Multicast Prefix table information :
Source: *, Group: 224.0.0.0/4, IIF: , Adjacency: Olist: 9226, Ports: 0
Source: *, Group: 224.0.0.0/24, IIF: , Adjacency: Olist: 9226, Ports: 4
Use the following command to verify that the information on BGP route type 3 is sent to L2RIB:

```
PE1# show l2vpn l2route evpn imet
```

<table>
<thead>
<tr>
<th>EVI</th>
<th>ETAG</th>
<th>Prod</th>
<th>Router IP Addr</th>
<th>Type</th>
<th>Label</th>
<th>Tunnel ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0</td>
<td>BGP</td>
<td>10.3.3.3</td>
<td>6</td>
<td>18</td>
<td>10.3.3.3</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>BGP</td>
<td>10.7.7.7</td>
<td>6</td>
<td>18</td>
<td>10.7.7.7</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>L2VPN</td>
<td>10.1.1.1</td>
<td>6</td>
<td>23</td>
<td>10.1.1.1</td>
</tr>
</tbody>
</table>

Use the following command to verify MPLS forwarding:

```
PE1# show mpls forwarding-table
```

<table>
<thead>
<tr>
<th>Local</th>
<th>Outgoing Prefix</th>
<th>Bytes Label</th>
<th>Outgoing Label or Tunnel Id</th>
<th>Switched interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>No Label</td>
<td>evpn(mc:bd 30)</td>
<td>305042</td>
<td>point2point</td>
</tr>
<tr>
<td>19</td>
<td>No Label</td>
<td>evpn(u:cb:bd 30)</td>
<td>7684</td>
<td>point2point</td>
</tr>
<tr>
<td>20</td>
<td>No Label</td>
<td>evpn(m:cb:bd 20)</td>
<td>542588</td>
<td>point2point</td>
</tr>
<tr>
<td>21</td>
<td>No Label</td>
<td>evpn(u:cb:bd 20)</td>
<td>13786</td>
<td>point2point</td>
</tr>
<tr>
<td>22</td>
<td>No Label</td>
<td>evpn(u:cb:bd 10)</td>
<td>6638</td>
<td>point2point</td>
</tr>
<tr>
<td>23</td>
<td>No Label</td>
<td>evpn(m:cb:bd 10)</td>
<td>277740</td>
<td>point2point</td>
</tr>
<tr>
<td>24</td>
<td>Pop Label</td>
<td>192.0.2.2-A</td>
<td>0</td>
<td>G11 192.0.2.2</td>
</tr>
<tr>
<td>25</td>
<td>Pop Label</td>
<td>192.0.2.2-A</td>
<td>0</td>
<td>G11 192.0.2.2</td>
</tr>
<tr>
<td>16001</td>
<td>Pop Label</td>
<td>10.3.3.3/32</td>
<td>0</td>
<td>G11 192.0.2.2</td>
</tr>
<tr>
<td>16002</td>
<td>Pop Label</td>
<td>10.2.2.2/32</td>
<td>0</td>
<td>G11 192.0.2.2</td>
</tr>
<tr>
<td>16004</td>
<td>Pop Label</td>
<td>10.7.7.7/32</td>
<td>0</td>
<td>G11 192.0.2.2</td>
</tr>
</tbody>
</table>

```
PE1# show ip bgp l2vpn evpn route-type 2
```

BGP routing table entry for [2][10.1.1.1:10][0][48][000C2993130E][0][*]/20, version 43

Advertised to update-groups:

2
Refresh Epoch 1

Local (via default) from 0.0.0.0 (10.1.1.1)
Origin incomplete, localpref 100, weight 32768, valid, sourced, local, best
EVPN ESI: 000000000000000000000000, Label1 22
Extended Community: RT:100:10
rx pathid: 0, tx pathid: 0x0

BGP routing table entry for [2][10.1.1.1:10][0][48][000C29B03E16][0][*]/20, version 116

Paths: (1 available, best #1, table evi_10)
Not advertised to any peer
Refresh Epoch 3

Local, (received & used), imported path from [2][10.7.7.10][0][48][000C29B03E16][0][*]/20 (global)
10.7.7.7 (metric 30) (via default) from 10.2.2.2 (10.2.2.2)
Origin incomplete, metric 0, localpref 100, valid, internal, best
EVPN ESI: 000000000000000000000000, Label1 19
Extended Community: RT:100:10
Originator: 10.7.7.7, Cluster list: 10.2.2.2
rx pathid: 0, tx pathid: 0x0

BGP routing table entry for [2][10.1.1.1:10][0][48][000C29B03E16][0][*]/20, version 116

Paths: (1 available, best #1, table evi_10)
Not advertised to any peer
Refresh Epoch 3

Local, (received & used), imported path from [2][10.7.7.10][0][48][000C29B03E16][0][*]/20 (global)
10.7.7.7 (metric 30) (via default) from 10.2.2.2 (10.2.2.2)
Origin incomplete, metric 0, localpref 100, valid, internal, best
EVPN ESI: 00000000000000000000, Label 19
Extended Community: RT:100:10
Originator: 10.7.7.7, Cluster list: 10.2.2.2
rx pathid: 0, tx pathid: 0x0
BGP routing table entry for [2][10.1.1.1:10][0][48][000C29EEEC0D][0][*]/20, version 134
Paths: (1 available, best #1, table evi_10)
Not advertised to any peer
Refresh Epoch 3
Local, (received & used), imported path from [2][10.3.3.3:10][0][48][000C29EEEC0D][0][*]/20 (global)
  10.3.3.3 (metric 30) (via default) from 10.2.2.2 (10.2.2.2)
    Origin incomplete, metric 0, localpref 100, valid, internal, best
    Extended Community: RT:100:10
    Originator: 10.3.3.3, Cluster list: 10.2.2.2
    rx pathid: 0, tx pathid: 0x0
EVPN ESI: 00000000000000000000, Label 19
Extended Community: RT:100:10
Originator: 10.3.3.3, Cluster list: 10.2.2.2
rx pathid: 0, tx pathid: 0x0

PE1# show ip bgp l2vpn evpn route-type 3
BGP routing table entry for [3][10.1.1.1:10][0][32][10.1.1.1]/17, version 41
Paths: (1 available, best #1, table evi_10)
Advertised to update-groups:
2
Refresh Epoch 1
Local
  :: (via default) from 0.0.0.0 (10.1.1.1)
    Origin incomplete, localpref 100, weight 32768, valid, sourced, local, best
    Extended Community: RT:100:10
    PMSI Attribute: for EVPN, Flags: 0x0, Tunnel type: 6, length 4, label: 23 (vni 368)
tunnel parameters: 0101 0101
  rx pathid: 0, tx pathid: 0x0
BGP routing table entry for [3][10.1.1.1:10][0][32][10.3.3.3]/17, version 137
Paths: (1 available, best #1, table evi_10)
Not advertised to any peer
Refresh Epoch 3
Local, (received & used), imported path from [3][10.3.3.3:10][0][32][10.3.3.3]/17 (global)
  10.3.3.3 (metric 30) (via default) from 10.2.2.2 (10.2.2.2)
    Origin incomplete, metric 0, localpref 100, valid, internal, best
    Extended Community: RT:100:10
    Originator: 10.3.3.3, Cluster list: 10.2.2.2
    PMSI Attribute: for EVPN, Flags: 0x0, Tunnel type: 6, length 4, label: 18 (vni 288)
tunnel parameters: 0303 0303
  rx pathid: 0, tx pathid: 0x0
BGP routing table entry for [3][10.1.1.1:10][0][32][10.7.7.7]/17, version 122
Paths: (1 available, best #1, table evi_10)
Not advertised to any peer
Refresh Epoch 3
Local, (received & used), imported path from [3][10.7.7.7:10][0][32][10.7.7.7]/17 (global)
  10.7.7.7 (metric 30) (via default) from 10.2.2.2 (10.2.2.2)
    Origin incomplete, metric 0, localpref 100, valid, internal, best
    Extended Community: RT:100:10
    Originator: 10.7.7.7, Cluster list: 10.2.2.2
    PMSI Attribute: for EVPN, Flags: 0x0, Tunnel type: 6, length 4, label: 18 (vni 288)
tunnel parameters: 0707 0707
  rx pathid: 0, tx pathid: 0x0
Additional References for EVPN Single-Homing

Standards and RFCs

<table>
<thead>
<tr>
<th>Standard</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 7432</td>
<td>BGP MPLS-Based Ethernet VPN</td>
</tr>
</tbody>
</table>

Feature Information for EVPN Single-Homing

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVPN Single-Homing</td>
<td>Cisco IOS XE Fuji 16.8.x</td>
<td>The EVPN Single-Homing feature utilizes the BGP MPLS-based Ethernet VPN (EVPN) functionality to achieve single-homing between a Provider Edge and a Customer Edge device. The following command was introduced or modified: address-family l2vpn, l2vpn evpn, member (bridge-domain), show ip bgp l2vpn evpn, show l2vpn evpn, show l2vpn l2route</td>
</tr>
</tbody>
</table>