Virtual Extensible LAN (VXLAN)

A Practical guide to VXLAN solution

Toni Pasanen, CCIE 28158

The Network Times
Handbook Series



Virtual Extensible LAN VXLAN

A Practical guide to VXLAN solution

Part 1.

Toni Pasanen
CCIE#28158

Copyright © Toni Pasanen, All rights reserved.

Revision History

August 2019: *First Edition*

October 2019: *Second Edition*

The second edition includes additional chapter 19: “Tenant Routed Multicast (TRM)”

October 2019: Editing numering of examples and figures (ch. 15-17).

November 2019: Editing numering of examples and figures. (ch. 1-14, 18-19)

About the Author:

Toni Pasanen. CCIE No. 28158 (Routing and Switching), Presales Engineer at Fujitsu Finland. Toni started his IT-carrier in 1998 at Tieto, where he worked as a Service Desk Specialist moving via LAN team to Data Center team as a 3rd. Level Network Specialist. Toni joined Teleware (Cisco Learning partner) in 2004, where he spent two years teaching network technologies focusing routing/switching and MPLS technologies. Toni joined Tieto again in 2006, where he spent next six years as a Network Architect before joining Fujitsu. In his current role, Toni work closely with customers helping them in selecting the right network solutions not only from the technology perspective but also from the business perspective.

Dedications

To my lovely wife Tiina, thanks for pushing me forward when I was about the throw this procject out of the window (more than twice).

Table of Contents

Chapter 1: Underlay Network – OSPF Operation 1

Introduction 1
OSPF 2
Link-State Database (LSDB) optimization 3
Shortest-Path First (SPF)/Dijkstra Algorithm 8
SPF Run – Phase I: Building a Shortest-Path Tree 9
First iteration round 10
Second iteration round 11
Third iteration round 12
Fourth iteration round 13
Fifth iteration round 14
Sixth iteration round 15
Seventh iteration round 16
SPF Run – Phase II: Adding Leafs to Shortest-Path Tree 17
References: 19

Chapter 2: Underlay Network – Comparison of OSPF and IS-IS 20

Scenario-1: Interface loopback 50 down on Leaf-101 (IS-IS) 22
Scenario-2: Interface loopback 50 down on Leaf-101 (OSPF) 24
Scenario-3: OSPF Incremental SPF – L55 Down on Leaf-101 (Stub) 26
Scenario-4: OSPF Incremental SPF – Interface g0/3 Down on Spine-12 (transit link does not participate in SPT) 27
Scenario-5: IS-IS SPF – Interface g0/3 Down on Spine-12 (Full SPF computation) 28
Scenario-6: IS-IS Incremental SPF – Interface g0/3 Down on Spine-12 29
(transit link does not participate in SPT) 29
Conclusion 30

Chapter 3: Underlay Network: iBGP in Underlay Network 32

Next-Hop-Self consideration 36
Case-1: Next-hop-self is changed by RR Spine-11. 36
Case-2: RR Spine-11 does not change Next-hop-self. 37

Chapter 4: Underlay Network: Two-AS eBGP 42

Underlay Network Control Plane eBGP 42
Overlay Network Control Plane: eBGP 46
References: 58

Chapter 5: eBGP as an Underlay Network Routing Protocol: Multi-AS eBGP 59

Underlay Network Control Plane: IPv4 eBGP peering 59
Overlay Network Control Plane: L2VPN EVPN eBGP peering 61
References: 69
Chapter 6: Layer 2 Multi-Destination Traffic - Anycast-RP with PIM. 70
Step 1: Configuring Anycast-RP cluster 71
Step 2: Assign unique Cluster Member IP and define members 71
Step 3: Assign unique Cluster Member IP and define members 72
Configuring NVE interface 74
Anycast-PIM Control Plane Operation 75
Phase 1: PIM Join 75
Phase 2: PIM Registration 76
Phase 3: PIM Registration-Stop 78
Phase 4: Anycast-RP peer notification 78
Data Plane operation 82
ARP Request 82
ARP Reply 83
References: 85
Chapter 7: Layer 2 Multi-destination traffic - PIM BiDir. 86
Configuration 86
Control Plane Operation 87
References 90
Chapter 8: BGP EVPN VXLAN Configuration and building blocks. 91
BGP EVPN VXLAN Building Blocks for Intra-VNI switching 91
Underlay Network: OSPF configuration 92
Overlay Network: BGP L2VPN EVPN configuration 93
Overlay Network: NVE Peering 93
Overlay Network: Host Mobility Manager 94
Overlay Network: Anycast Gateway (AGW) 94
Overlay Network: VLAN based service 95
Overlay Network: TCAM modification 95
Intra-VNI service (L2VNI) in VXLAN Fabric 96
Tenant based Inter-VNI Routing (L3VNI) in VXLAN Fabric 100
References 103
Chapter 9: BGP EVPN VXLAN Control and Data Plane Operation. 104
MAC address learning process 105
Phase 1: MAC Address-Table update 105
Phase 2: L2RIB Update 106

Phase 3: BGP MAC Route Export on Local VTEP 107
Phase 4: BGP AFI L2EVPN MAC Route Import on Remote VTEP 110
Phase 5: MAC VRF on Remote VTEP 112
Phase 6: MAC Address Table on Remote VTEP 113
L2VNI: Intra-VNI Data Plane 113
ARP Request 114
ARP Reply 116
ICMP Request 118
ICMP Reply 120
Summary 121
MAC-IP address learning process 122
Phase 1: ARP Table on Local VTEP 123
Phase 2-3: MAC-IP on Local VTEP 123
Phase 4: BGP Route Export on Local VTEP 125
Phase 5: BGP Route Import on Remote VTEP 126
Phase 6: IP VRF on Remote VTEP 129
ARP-Suppression 130
Host route Advertisement: Inter-VNI routing (L3VNI) 132
Phase 1. Host Route in Local Routing Information Base (RIB) 133
Phase 2. Host Route BGP Process on Local VTEP 133
Phase 3. Host Route BGP Process on Remote VTEP 134
Phase 4. Installing Host Route into RIB of Remote VTEP 135
Data Plane operation 137
Phase 1. Switching in VNI30000 on VTEP-102 137
Phase 2. Routing from VNI30000 to VNI 10077 on VTEP-102 138
Phase 3. Routing from VNI10077 to VNI 10000 on VTEP-101 138
Summary 139
Prefix Advertisement 139
Phase 1: vmBeef start pinging to vmBebe 140
Phase 2: Local VTEP Leaf-101: ARP process 141
Phase 3: Remote VTEP Leaf-102: ARP process - Request 142
Phase 4: vmBebe: ARP process - Reply 143
Phase 5: remote VTEP switch Leaf-102: BGP Update 144
Phase 6: Local VTEP switch Leaf-102: BGP Update 144
Data Plane testing 148
Phase 1: vmBeef start pinging to vmBebe 149
Phase 2: Local VTEP Leaf-101: Routing 149
Phase 3-4: Remote VTEP Leaf-102: ARP request 150

Phase 5: vmBebe: ARP Reply 151
Phase 6: Remote VTEP Leaf-102: ICMP Request forwarding 152
Phase 7: vmBebe: ICMP reply 152
Phase 8-9: Remote VTEP Leaf-102: Routing decision and ICMP reply 152
Phase 10-11: Local VTEP Leaf-101: Routing decision and ICMP reply 153
Summary 156
References 157
Chapter 10: VXLAN fabric External Connections 158
eBGP Configuration between Border Leaf-102 and Ext-Ro01 158
Starting point 160
Chapter 11: Multihoming with vPC 190
Virtual Port Channel Configuration 190
Some other consideration for vPC: 197
VTEP redundancy with vPC 198
Advertising Primary IP address 204
References: 210
Chapter 12: Multihoming - vPC and Graceful Insertion and Removal (GIR) operation 211
Loopback addressing 211
Graceful Insertion and Removal (GIR) 212
Verifications. 213
Example-2 summary: BGP EVPN peering and NVE1 using the same Loopback interface. 218
Conclusion 219
References: 220
Chapter 13: Using vPC Peer Link as an Underlay Backup Path 221
Configuration 222
Verification 223
References: 227
Chapter 14: VXLAN Fabric Firewall Implementation 228
Protected segment 229
Non-Protected segment 230
Connectivity Testing 238
References: 240
Chapter 15: EVPN ESI Multihoming 241
Introduction 241

Ethernet Segment Identifier (ESI) and Port-Channel 242
Designated Forwarder (DF) 243
Designated Forwarder 246
References: 248
Chapter 16: EVPN ESI Multihoming - Fast Convergence and Load Balancing 249
Ethernet A-D per ES route - Fast Convergence in the all-Active mode 249
Fast Convergence 254
Load Balancing (Aliasing) 257
Summary 258
References: 259
Chapter 17: EVPN ESI Multihoming - Data Flows and link failures 260
Introduction 260
Intra-VNI (L2VNI): Unicast Traffic 262
Scenario 1: Link E1/2 down on Leaf-102 262
Scenario 2: Core link down on Leaf-102. 265
Intra-VNI (L2VNI): Broadcast, Unknown Unicast and Multicast (BUM) traffic 266
Scenario 1: Traffic flow from Designated Forwarder 266
Scenario 2: Traffic flow from non-Designated Forwarder 267
CHAPTER 18: VXLAN EVPN Multi-Site 269
Shared EVPN domain limitations 269
EVPN Multi-Site Architecture Introduction 270
Intra-Site EVPN Domain (Fabric) 271
Intra-Site NVE peering and VXLAN tunnels 272
Summary 278
Shared Common EVPN Domain Connections 278
Border Gateway setup 279
Multi-Destination traffic forwarding 287
Designated Forwarder 287
Ingress-Replication 293
Fabric Link Failure 299
Normal State 300
Fabric-Link Failure 302
Fabric-Link Recovery 304
DCI-Link Failure 307
Normal State 308
DCI Link Failure 309
DCI Link Recovery 310

References 312
Chapter 19: Tenant Routed Multicast in VXLAN Fabric 313
Underlay Multicast Routing 315
PIM neighbor establishment process 315
Shared Multicast Tree for Intra-VN 316
Joining to Intra-VN Shared Tree 316
Joining to Intra-VN Source-Specific Tree 318
Tenant Routed Multicast (TRM) Configuration 325
Define Anycast-RP 325
Enable TRM on leaf switches 325
Define the tenant-based Multicast group for Multicast traffic. 326
Prevent PIM neighbor establishment within a specific VLAN 326
BGP afi IPv4 MVPN peering (Leaf) 327
BGP afi IPv4 MVPN peering (Spine) 327
Tenant Routed Multicast (TRM) operation 328
Shared/Source-Specific tree for Inter-VN 328
Verification 330
TRM Control Plane operation. 332
IGMP membership report 332
MVPN Source-Active Auto-Discovery 333
Data Plane Operation 340
Ingress leaf operation 340
Spine operation 341
Egress leaf operation 341
Summary 342
References 343

About this book

This book is based on my personal blog "The Network Times" (nwktimes.blogspot.com). The first quarter of the book focuses on the Underlay Network solutions used in VXLAN fabric. It starts by explaining the operation of OSPF, focusing on the Dijkstra algorithm and the Shortest Path Tree calculation process. Then it discusses the differences between the OSPF and IS-IS routing protocols from the Underlay Network perspective. The first part also introduces three BGP based Underlay Network routing solutions with Single-AS solution, Dual-AS solution, and Multi-AS solution. After the Unicast Routing section, this book explains the Multicast Routing solution used for L2VNI specific L2BUM traffic forwarding by introducing the Anycast-RP with PIM and the PIM BiDir solutions.

The focus of the second quarter is VXLAN with BGP EVPN Control Plane. This part of the book explains the basic building blocks and configurations needed for both Layer2 and Layer3 services. It also discusses the BGP EVPN Control Plane operation by showing how a BGP EVPN NLRI information is advertised within a VXLAN fabric and how this information is used in Data Plane.

The third quarter discusses a multi-homing solution with vPC by explaining the vPC Multi-homing, vPC and GIR, and vPC Peer-Link as an Underlay Network Backup Path. In addition to vPC this section explains how to implement Firewall into VXLAN Fabric.

The Last quarter of this book starts by explains the standard based EVPN ESI Multi-homing solution. It also discusses the Data Center Interconnect (DCI) solution based on the EVPN Multi-Site architecture. In addition, this part introduces Tenant Routed Multicast (TRM) solution.

Each chapter includes various configuration and verification examples as well as traffic captures. The only physical device used in labs is my personal computer and the example labs are done by using both Cisco "Virtual Internet Routing Lab" (VIRL) and Emulated Virtual Environment - Next Generation (EVE-NG).

Disclaimers

The content of this book is based on the authors own experience and testing results. This book is meant to be neither a Design nor Implementation guide but rather it tries to give to readers a basic understanding of VXLAN. After reading this book, readers should do their own technology validation before using it in production environment.

Chapter 1: Underlay Network – OSPF Operation

Introduction

The main job for the Underlay Network from the EVPN VXLAN Fabric perspective is to offer resilient IP connectivity between VXLAN Tunnel End Point (VTEP) devices. It can also be used for Layer2 BUM (Broadcast, Unknown Unicast, and Multicast) traffic forwarding though this requires a Multicast Routing on an Underlay Network. The common routing protocol choices for the Underlay Network are OSPF and IS-IS, which are Link State Protocols and BGP, which in turn is a Path Vector Protocol.

The reason why this chapter is included in the content of this book is that by understanding the operation of the Link State Protocol it is hopefully easier to choose between the BGP and IGP when selecting routing protocol running in an Underlay Network.

This chapter explains the operation of the Dijkstra/Shortest Path First (SPF) algorithm that is used by Link State Protocols for calculating the Shortest-Path Tree. This chapter also discusses how the Shortest-Path Tree calculation process can be optimized by disabling the advertisement of unnecessary LSU information. In addition, this chapter illustrates how devices exchange the Link State information and based on that how they calculate the Shortest-Path Tree.

OSPF

OSPF router exchanges Link State Update (LSU) packets with adjacent routers. LSU packets carries the Link State Advertisement (LSA) where router describes its links. After successful validation of the received LSU, the router stores the Link descriptions into OSPF Link-State Database (LSDB) and floods it to adjacent routers. This way each router has a common topology view of an OSPF Area. After the LSDB synchronization process, routers run the Dijkstra algorithm to find the shortest/best path to each destination. The terms Dijkstra/SPF are used interchangeably in this chapter.

Figure 1-1 introduces the example topology used in this chapter. Switches from Leaf-101 to Leaf-105 represents VTEP switches, which each have an NVE (Network Virtualization Edge) Interface (192.168.50.x). In addition, there are two Core/Spine switches Spine-11 and Spine-12. The topology between VTEP switches Leaf-101, Leaf-102, and Spine-11 and Spine-12 follow the Spine-Leaf topology, which is widely used in Datacenter networks (excluding link between Spine switches). VXLAN based Network Virtualization Overlay (NVO) solutions have also become an alternative to the traditional Enterprise network architecture, where the Spanning-Tree Protocol is commonly used as a Control Plane protocol. Enterprise LAN cabling scheme might enforce to build a physical topology, which is not ideal.

All Inter-Switch links and Loopback addresses belong to OSPF Area 0 in the example network. The OSPF metric values shown in the figure are set statically by configuring an OSPF cost per interface. The IP addressing scheme of physical interfaces is based on switch numbering. As an example, the interface g0/1 towards Spine-11on Switch Leaf-102 has an IP address 10.102.11.102/24. Obviously, this is not the ideal solution and in real life, the use of mask /30, /31 or Unnumbered interface is a much elegant solution. OSPF Router-Ids are defined statically (192.168.0.sw-id). Note that there are no RID related Loopback Interfaces in this example network. This is not the recommended design but it reduces the LSAs in LSDB and makes it easier to explain the Dijkstra algorithm. In addition, there is a hostname-to-IP address mapping information and OSPF name-lookup enabled in each switch. This way the advertising OSPF routers are identified by its name instead of its IP address.

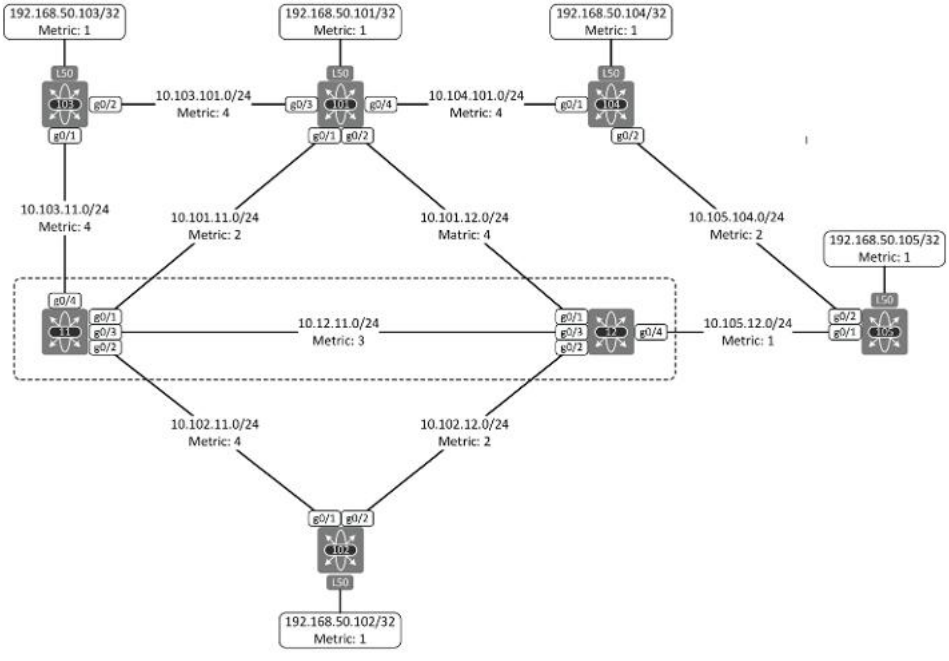


Figure 1-1: *Underlay Network Topology and Interface Addressing.*

Link-State Database (LSDB) optimization

The default OSPF network type in the Ethernet link is Broadcast. This means that routers connected to link select the Designated Router (DR) and Backup-designated Router (BDR) among themselves. Non-Designated routers send Router LSAs only to DR, which in turn floods them to other routers in the segment as a Network LSA (Type-2).

Figure 1-2 shows the LSA generation from the Switch Leaf-102 perspective, which is the DR for segments connected to both Spine-11 and Spine-12. Switch Leaf-102 originates the LSU packet, which carries both Router LSAs and Network LSAs and sends it to both adjacent switches.

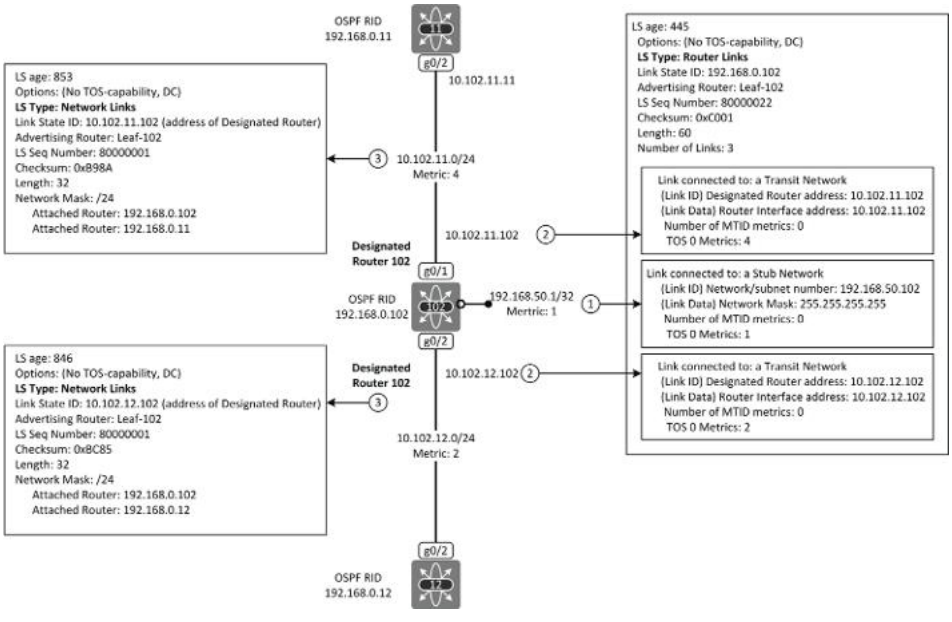


Figure 1-2: *Router LSA and Network LSAs originated by Leaf-102 (DR).*

The process of the DR/BDR election is useless in a segment where there are only two OSPF speakers. It also slightly increases the recovery times and generates an unnecessary Network LSAs. The first step in the LSDB optimization process is to change the OSPF Network type form Broadcast to Point-to-Point, where there is no DR/BDR election per segment and adjacent routers exchange only Router LSAs.

Figure 1-3 shows the Router LSAs generated by Leaf-102 after the OSPF network type is changed from Broadcast to Point-to-Point. Leaf-102 describes its neighbor routers and the local interfaces by using Link Type-1 Router LSA (shown as “Link connected: to another router”). This interface information is used for routing. In addition, Leaf-102 describes the subnet to which the routers are connected to by using the Link Type-3 Router LSA (shown as “connected to Stub Network”). Both of these LSAs are flooded to every switch inside an OSPF area. The sub-networks are *Transit networks* and are not used for data traffic, which means that they consume unnecessary hardware resources and decreases the convergence time. These unnecessary Transit Network LSAs can be hidden by using Prefix-Suppression.

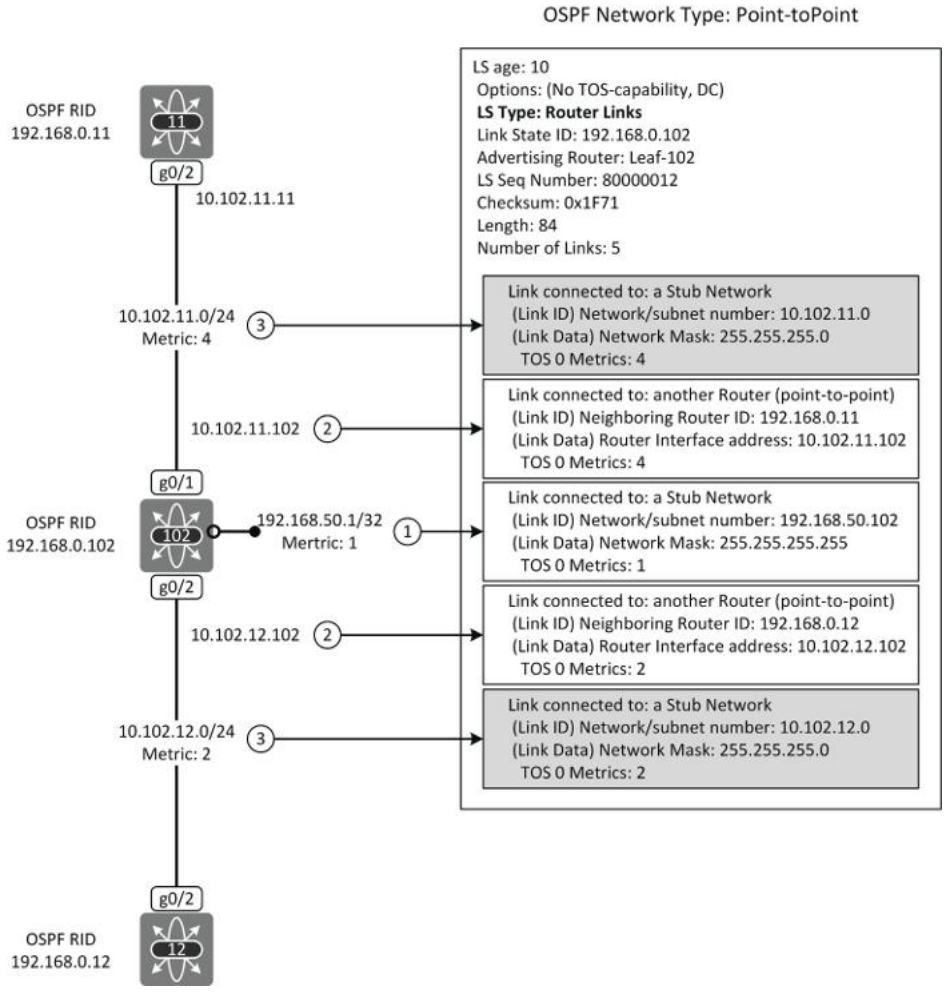


Figure 1-3: *Router LSA originated by Leaf-102.*

Example 1-1 illustrates the OSPF LSDB taken from Leaf-102 before prefix-suppression. The link count in Area 0 is 45.

```
Leaf-102#sh ip ospf database | b Link

Link ID         ADV Router      Age         Seq#       Checksum Link count
192.168.0.11    Spine-11        31          0x80000003 0x00701F 8
192.168.0.12    Spine-12        18          0x80000006 0x007C05 8
192.168.0.101   Leaf-101        24          0x80000006 0x003B53 9
192.168.0.102   Leaf-102        62          0x80000004 0x003B63 5
192.168.0.103   Leaf-103        58          0x80000004 0x009EE6 5
192.168.0.104   Leaf-104        52          0x80000003 0x00DB8A 5
192.168.0.105   Leaf-105        45          0x80000003 0x00FD74 5
```

Example 1-1: *OSPF LSDB from Leaf-102.*

Example 1-2 shows the routing table of switch 102 before prefix-suppression. Each Transit network is learned via OSPF.

```
Leaf-102#sh ip route ospf
<snipped>

        10.0.0.0/8 is variably subnetted, 12 subnets, 2 masks
O        10.12.11.0/24 [110/4] via 10.102.12.12, 00:32:53, GigabitEthernet0/2
O        10.101.11.0/24 [110/6] via 10.102.12.12, 00:33:07, GigabitEthernet0/2
                        [110/6] via 10.102.11.11, 00:33:07, GigabitEthernet0/1
O        10.101.12.0/24 [110/6] via 10.102.12.12, 00:32:53, GigabitEthernet0/2
O        10.103.11.0/24 [110/8] via 10.102.12.12, 00:33:07, GigabitEthernet0/2
                        [110/8] via 10.102.11.11, 00:33:07, GigabitEthernet0/1
O        10.103.101.0/24
           [110/10] via 10.102.12.12, 00:32:43, GigabitEthernet0/2
           [110/10] via 10.102.11.11, 00:32:43, GigabitEthernet0/1
O        10.104.101.0/24
           [110/9] via 10.102.12.12, 00:33:31, GigabitEthernet0/2
O        10.105.12.0/24 [110/3] via 10.102.12.12, 00:32:53, GigabitEthernet0/2
O        10.105.104.0/24
           [110/5] via 10.102.12.12, 00:33:31, GigabitEthernet0/2
      192.168.50.0/32 is subnetted, 5 subnets
O        192.168.50.101 [110/7] via 10.102.12.12, 01:30:05, GigabitEthernet0/2
                        [110/7] via 10.102.11.11, 01:29:25, GigabitEthernet0/1
O        192.168.50.103 [110/9] via 10.102.11.11, 01:29:25, GigabitEthernet0/1
O        192.168.50.104 [110/6] via 10.102.12.12, 01:29:35, GigabitEthernet0/2
O        192.168.50.105 [110/4] via 10.102.12.12, 01:29:45, GigabitEthernet0/2
```

Example 1-2: *OSPF routes in RIB of Leaf-102.*

Figure 1-4 shows the LSU packet and LSAs originated by Leaf-102 when the OSPF network type is changed to Point-to-Point in all Inter-Switch Links and the Transit networks are excluded from the LSAs by using prefix-suppression option. When these changes are done in all OSPF switches, the LSDB optimization is ready. Example 1-3 shows the OSPF configuration related to LSAs optimization.

```
Leaf-102(config)# interface g0/1
Leaf-102(config-if)# ip ospf network point-to-point
Leaf-102(config-if)# ip ospf prefix-suppression
```

Example 1-3: *Leaf-102 Interface g0/1 configuration.*

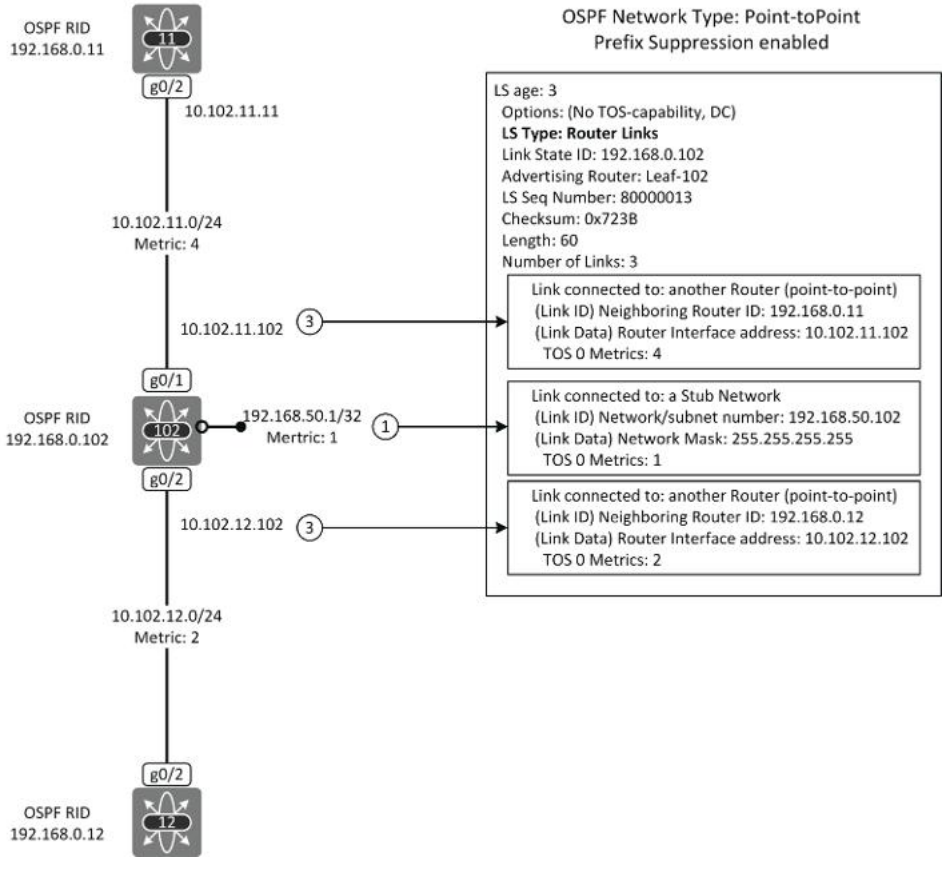


Figure 1-4: *Router LSA originated by Switch 102.*

Example 1-4 shows the OSPF LSDB taken from Leaf-102 after prefix-suppression. The Link count is now reduced from 45 to 25.

```
            OSPF Router with ID (192.168.0.102) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
192.168.0.11    Spine-11        29          0x80000004 0x00432B 4
192.168.0.12    Spine-12        9           0x80000007 0x000959 4
192.168.0.101   Leaf-101        21          0x80000007 0x007B05 5
192.168.0.102   Leaf-102        69          0x80000005 0x008E2D 3
192.168.0.103   Leaf-103        68          0x80000005 0x00A857 3
192.168.0.104   Leaf-104        65          0x80000004 0x00AF8F 3
192.168.0.105   Leaf-105        44          0x80000004 0x00B13E 3
```

Example 1-4: *OSPF LSDB from switch 102 after prefix suppression.*

Example 1-5 shows that there are no Transit networks after the prefix-suppression in Leaf-102 Routing Information Base (RIB). The highlighted portion of example 1-5 shows that even though the local Transit networks and related IP addresses are hidden from adjacent switches, they still exist in the local RIB.

```
Leaf-102#sh ip route ospf
<snipped>
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        10.102.11.0/24 is directly connected, GigabitEthernet0/1
L        10.102.11.102/32 is directly connected, GigabitEthernet0/1
C        10.102.12.0/24 is directly connected, GigabitEthernet0/2
L        10.102.12.102/32 is directly connected, GigabitEthernet0/2
      192.168.50.0/32 is subnetted, 5 subnets
O        192.168.50.101 [110/7] via 10.102.12.12, 00:28:15, GigabitEthernet0/2
                        [110/7] via 10.102.11.11, 00:27:55, GigabitEthernet0/1
C        192.168.50.102 is directly connected, Loopback50
O        192.168.50.103 [110/9] via 10.102.11.11, 00:27:55, GigabitEthernet0/1
O        192.168.50.104 [110/6] via 10.102.12.12, 00:28:05, GigabitEthernet0/2
O        192.168.50.105 [110/4] via 10.102.12.12, 00:28:05, GigabitEthernet0/2
```

Example 1-5: *Switch 102 Interface g0/1 configuration.*

Shortest-Path First (SPF)/Dijkstra Algorithm

Dijkstra/SPF algorithm is used for calculating a Shortest-Path Tree (SPT) topology in OSPF Area. A router starts the process by setting itself as a root of the tree. At the first stage, the router builds a Shortest-Path Tree between routers by using the Type-1 Link Description (point-to-point) which describes links to neighbor routers in Router LSA. When the Shortest-Path Tree is formed, the router calculates the distance to subnets connected to each router by using the Link Type-3 (Stub) Link Description in Router LSA.

Routers have two lists related to SPT calculation. The *Candidate List* (also known as a *Tentative List*) is the list that includes all routers that are currently examined by the router. The *Tree List* (also called *Path* or *Known List*) is the list, which includes all the routers participating in a final Shortest-Path Tree. In addition, a Link State Database (LSDB) is a source from where the information is pulled to calculation and actually it is sometimes called the *Unknown List*.

The next section describes the SPT calculation process from the switch Leaf-102 perspective.

SPF Run – Phase I: Building a Shortest-Path Tree

Figure 1-5 shows the initial situation where the switch Leaf-102 starts the Shortest-Path Tree calculation. Leaf-102 inserts itself into the Candidate-list with cost 0 and with next-hop pointing to itself. All other switches are in Unknown-list at this phase. The Path-list is empty at the initial situation. Table 1-1 illustrates the Unknown/Candidate/Path list progress in this stage.

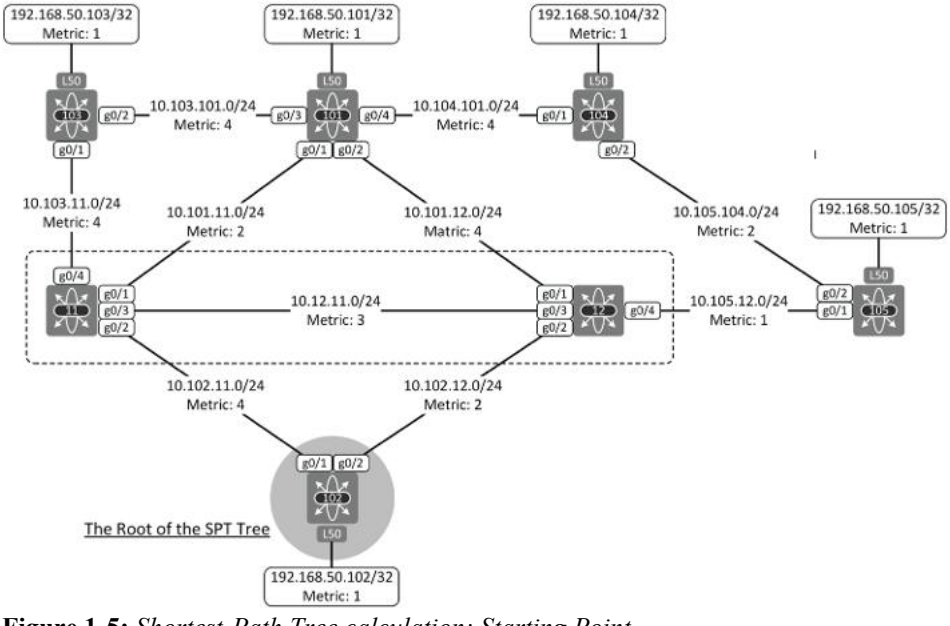


Figure 1-5: *Shortest-Path Tree calculation: Starting Point*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
Spine-11 (192.168.0.11)	Leaf-102, 0, Leaf-102	
Spine-12 (192.168.0.12)		
Leaf-101 (192.168.0.101)		
Leaf-103 (192.168.0.103)		
Leaf-104 (192.168.0.104)		
Leaf-105 (192.168.0.105)		

Table 1-1: *Shortest Path Tree: Starting point on Switch 102.*

First iteration round

Figure 1-6 shows the first SPF iteration round. Leaf-102 inserts itself to the Path List (table 1-2 step-A). Leaf-102 examines its self-originated Router LSA. It starts from the first Link Description (LD) found from the LSA. First LD is Link Type-3 (Stub) so Leaf-102 ignores it (1). The next entry describes the link to Spine-11 (Link Type-1), which is adjacent via local interface g0/1. Leaf-102 moves Spine-11 into the Candidate-list with cost 4 via interface g0/1 (2). The last LD describes the link to Spine-12 that is reachable via interface g0/2. The leaf-102 move also Spine-12 to the Candidate List (3). Leaf switches 101, 103-105 are still in the Unknown-list.

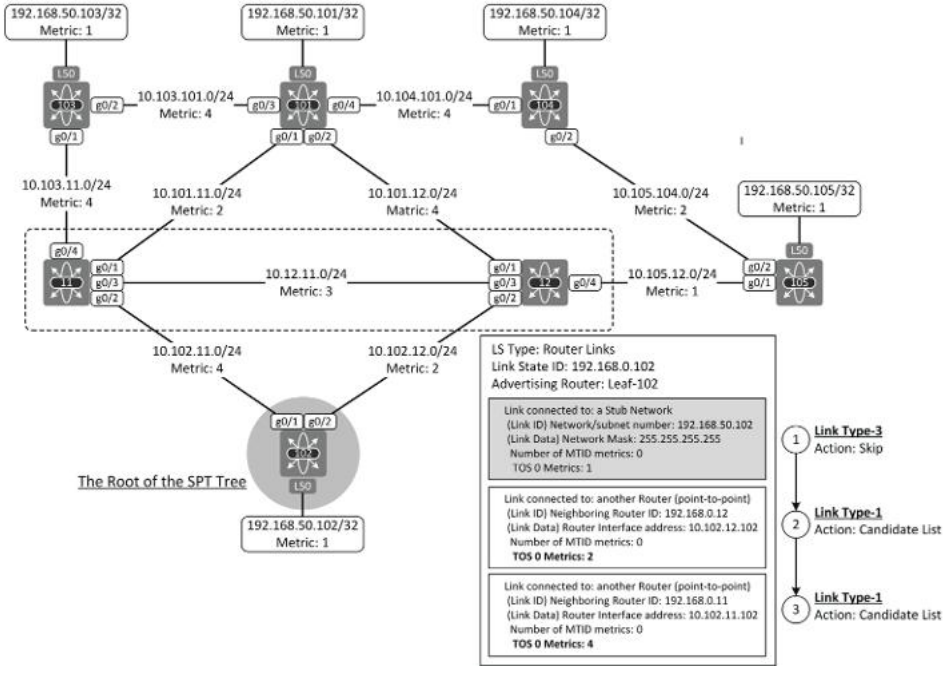


Figure 1-6: *Shortest-Path Tree calculation: First iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
		(A) Leaf-102, 0, Leaf-102
	(2) Spine-11, 4, Gi0/1	
Leaf-101 (192.168.0.101)	(3) Spine-12, 2, Gi0/2	
Leaf-103 (192.168.0.103)		
Leaf-104 (192.168.0.104)		
Leaf-105 (192.168.0.105)		

Table 1-2: *Shortest Path Tree: 1st. Iteration on Leaf-102.*

Second iteration round

Leaf-102 moves Spine-12 to the Path-list (B) because it has the lowest cost among all the devices listed on the Candidate-list (B). Leaf-102 starts checking the LSA advertised by Spine-12. First LD describes the link to Leaf-105 with cost 1. Leaf-102 inserts Leaf-105 into Candidate-list (1). Next LD describes the link to Spine-11 with cost 3 (total cost 5). There is already an entry with better cost in the Candidate-list (b), so Leaf-102 skip this LD (2). Next one describes to link back to Leaf-102, so Leaf-102 skip it also (3). Last LD points to Leaf-101 with cost 4 (total cost 8). This information is added to the Candidate-list by Leaf-102 (4). Leaf switches 103-104 are still in the Unknown-list.

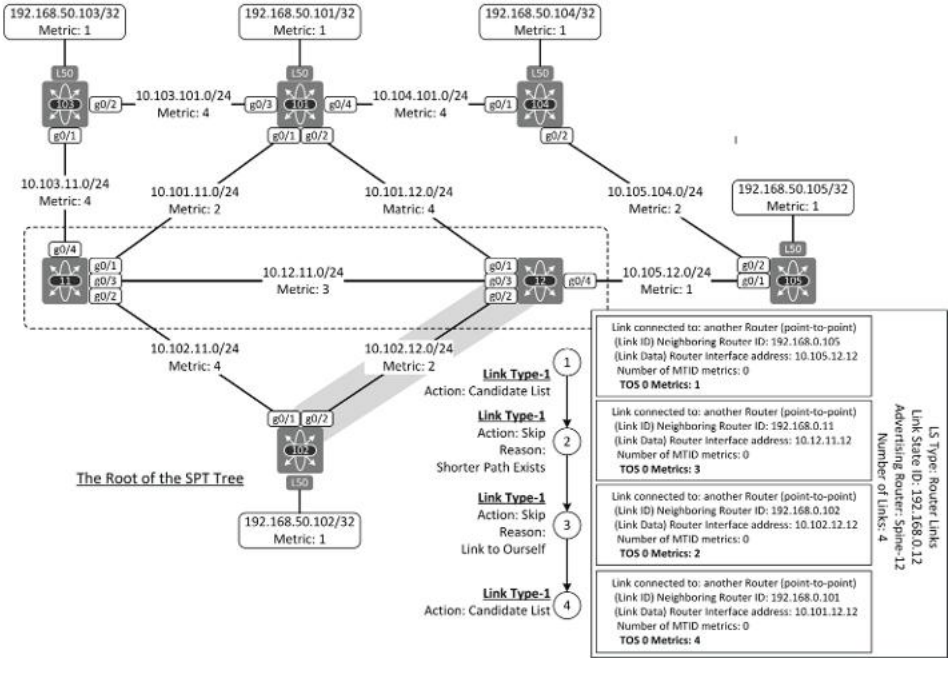


Figure 1-7: *Shortest-Path Tree calculation: Second iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
		Leaf-102, 0, Leaf-102
	(b) Spine-11, 4, Gi0/1	(B) Spine-12, 2, Gi0/2
	(1) Leaf-105, 3, Spine-12	
Leaf-103 (192.168.0.103)	(4) Leaf-101, 6, Spine-12	
Leaf-104 (192.168.0.104)		

Table 1-3: *Shortest-Path Tree calculation: Second iteration round on Leaf-102*

Third iteration round

Leaf-102 moves Leaf-105 to the Path-list (C) because it has the lowest cost among all devices listed on the Candidate-list. Leaf-102 starts checking the LSA advertised by Leaf-105. First LD describes the Stub Network (Link Type-3) and it is skipped (1). Next LD describes the link to Leaf-104. Leaf-102 add Leaf-104 into Candidate-list with total cost 5 (2+1+2) with Spine-12 as a next-hop. The last LD describes the link to Spine-12, which already is installed into Path-list with better cost (3). Only Leaf-103 is now in the Unknown-list.

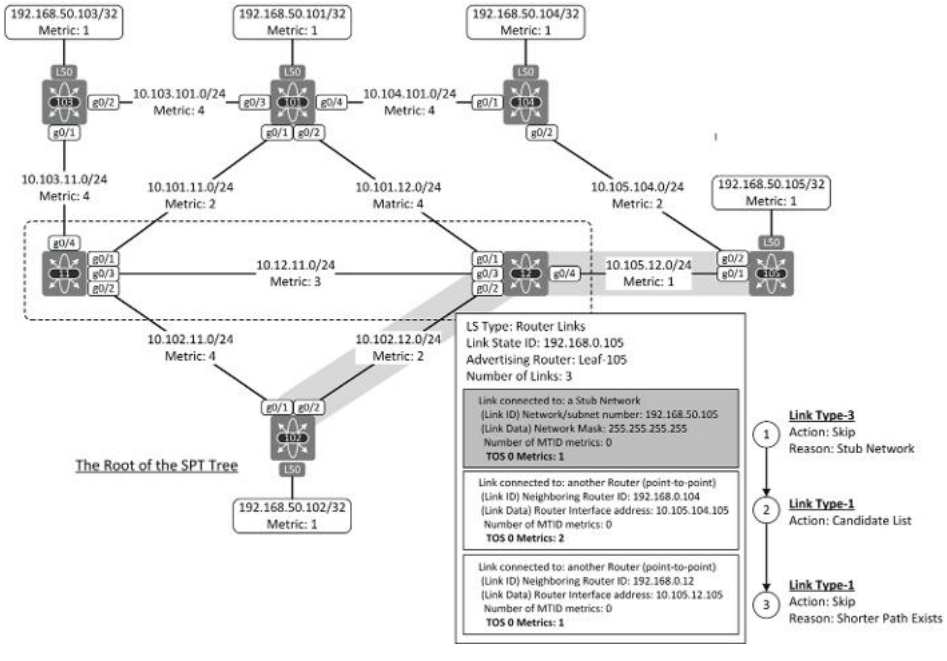


Figure 1-8: *Shortest-Path Tree calculation: Third iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
		Leaf-102, 0, Leaf-102
	Spine-11, 4, Gi0/1	Spine-12, 2, Gi0/2
	Leaf-101, 6, Spine-12	(C) Leaf-105, 3, Spine-12
Leaf-103 (192.168.0.103)	(2) Leaf-104, 5, Spine-12	

Table 1-4: *Shortest-Path Tree calculation: Third iteration round on Leaf-102.*

Fourth iteration round

Leaf-102 moves Spine-11 to the Path-list (D) because it has the lowest cost among all the devices listed on the Candidate-list. Leaf-102 starts checking the LSA originated by Spine-11. First LD describes the link to Leaf-103 (1). Leaf-102 add Leaf-103 to Candidate-list with total cost 8 (4+4) and Spine-11 as a next-hop. Next LD is about Spine-12 that already exists on the Path-list with a better cost. Leaf-102 skips this one. Next LD describes the link back to Leaf-102 so it is also skipped. The fourth LD describes the link to Leaf-101 with the same total cost 6 than what Spine-12 has described. Leaf-102 updates Candidate-list concerning Leaf-101, which now has two equal-cost paths via Spine-11 and Spine-12.

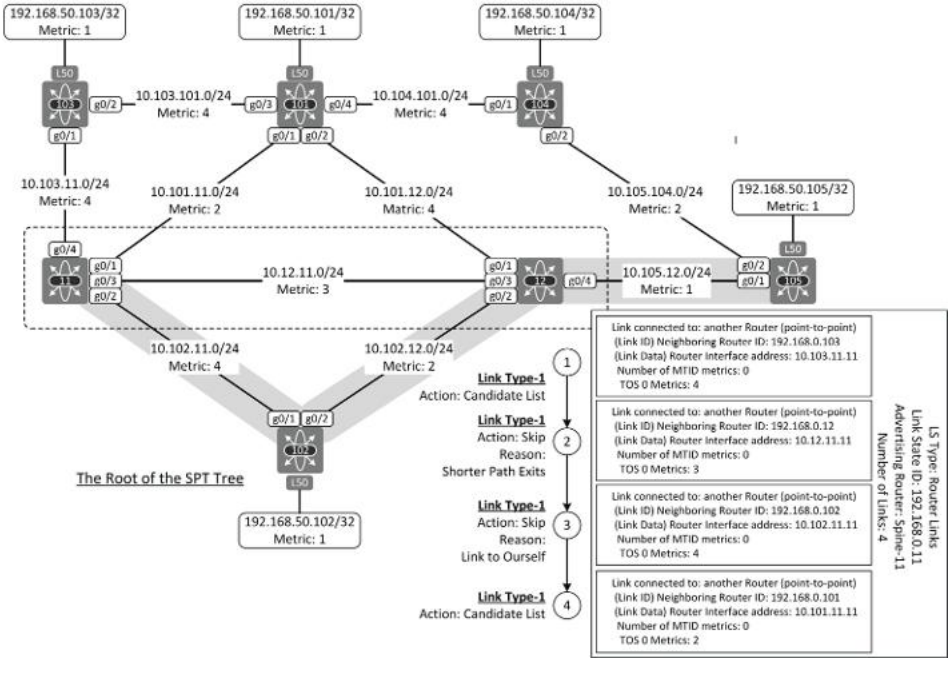


Figure 1-9: *Shortest-Path Tree calculation: Fourth iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
	Leaf-101, 6, Spine-11	Leaf-102, 0, Leaf-102
	Leaf-101, 6, Spine-12	Spine-12, 2, Gi0/2
	Leaf-104, 5, Spine-12	Leaf-105, 3, Spine-12
	(1) Leaf-103, 8, Spine-11	(D) Spine-11, 4, Gi0/1

Table 1-5: *Shortest-Path Tree calculation: Fourth iteration round on Leaf-102.*

Fifth iteration round

Leaf-102 moves Leaf-104 from the Candidate-list to Path (total cost 5, next-hop Spine-12) because it has the lowest cost among all the devices listed on the Candidate List (E). The first LD of the LSA generated by Leaf-104 is a Stub Network (Link Type-3) so it is ignored by Leaf-102. Next LD describes the link to Leaf-105 (total cost 5, next-hop Spine-12), which already is added to Path-list with a better cost. Leaf-102 ignores this one too. The last LD describes the link to Leaf-101 (total cost 9, next-hop Spine-12), which already is in Candidate-list with better total costs via Spine-11 and Spine-12, so also the last LD is ignored. None of the LD described in LSA originated by Leaf-104 does not end up to the Path-list.

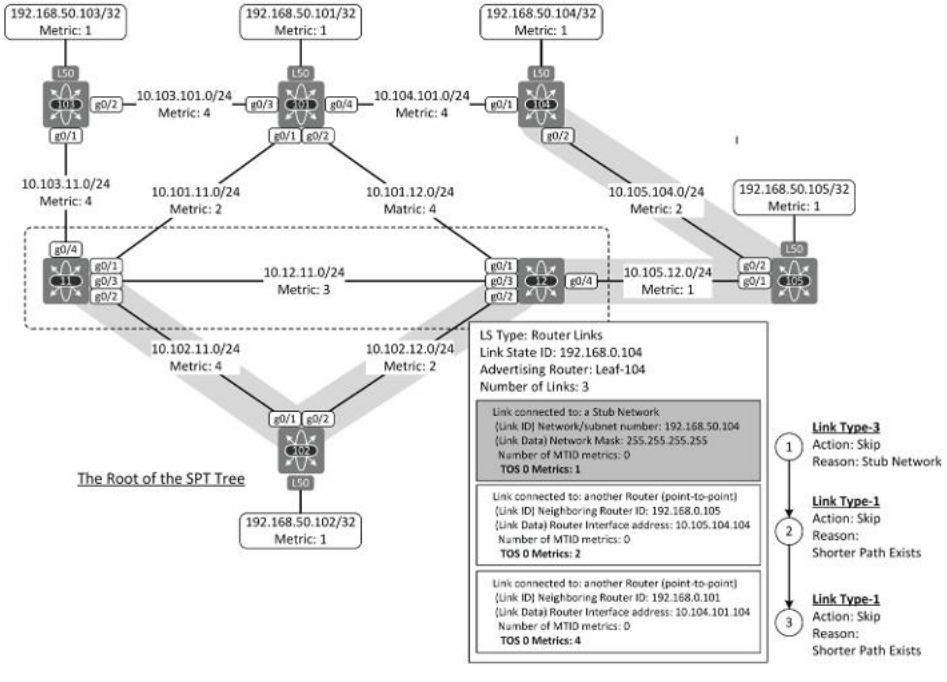


Figure 1-10: *Shortest-Path Tree calculation: Fifth iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
	Leaf-101, 6, Spine-11	Leaf-102, 0, Leaf-102
	Leaf-101, 6, Spine-12	Spine-12, 2, Gi0/2
	Leaf-103, 8, Spine-11	Leaf-105, 3, Spine-12
		Spine-11, 4, Gi0/1
		(E) Leaf-104, 5, Spine-12

Table 1-6: *Shortest-Path Tree calculation: Fifth iteration round on Leaf-102.*

Sixth iteration round

Leaf-102 moves Leaf-101 from the Candidate-list to Path-list with total cost 6 via Spine-11 and Spine-12 (F). First LD originated by Leaf-101 is Stub Network and it is skipped from Shortest-Path Tree calculation. The next four LDs describes the links to switches Leaf-103, Leaf-104, Spine-11, and Spine-12. Leaf-102 has a better path with lower costs related to these switches and this is why all four Link Descriptions are ignored links. Just like in case of Leaf-104, none of the LD described in LSA originated by Leaf-101 does not end up to the Path-list.

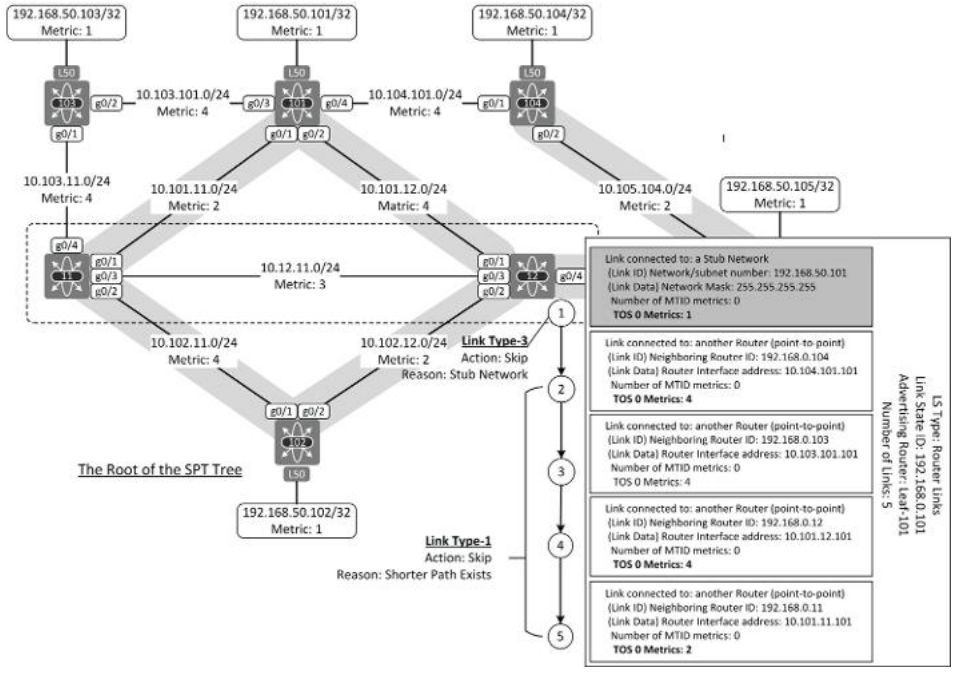


Figure 1-11: *Shortest-Path Tree calculation: Sixth iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
		Leaf-102, 0, Leaf-102
		Spine-12, 2, Gi0/2
	Leaf-103, 8, Spine-11	Leaf-105, 3, Spine-12
		Spine-11, 4, Gi0/1
		Leaf-104, 5, Spine-12
		(F) Leaf-101, 6, Spine-11 and Spine-12 (ECMP)

Table 1-7: *Shortest-Path Tree calculation: Sixth iteration round on Leaf-102.*

Seventh iteration round

Leaf-102 moves the last switch Leaf-103 Candidate-list (G) to the Path-list with a total cost of 8 and Spine-11 as a next-hop. The first LD describes the Stub Network, so it is excluded from Shortest-Path Tree calculation. The second LD describes the link to Leaf-101, which already is in Path-list with a better metric, so it is also left out from the SPT calculation. The last LDs describe the link to Spine-11, which also is in Path List with a better overall metric. At this stage, the Shortest-Path Tree is ready and Leaf-102 starts phase two where it calculates the Shortest-Paths to Stub Networks.

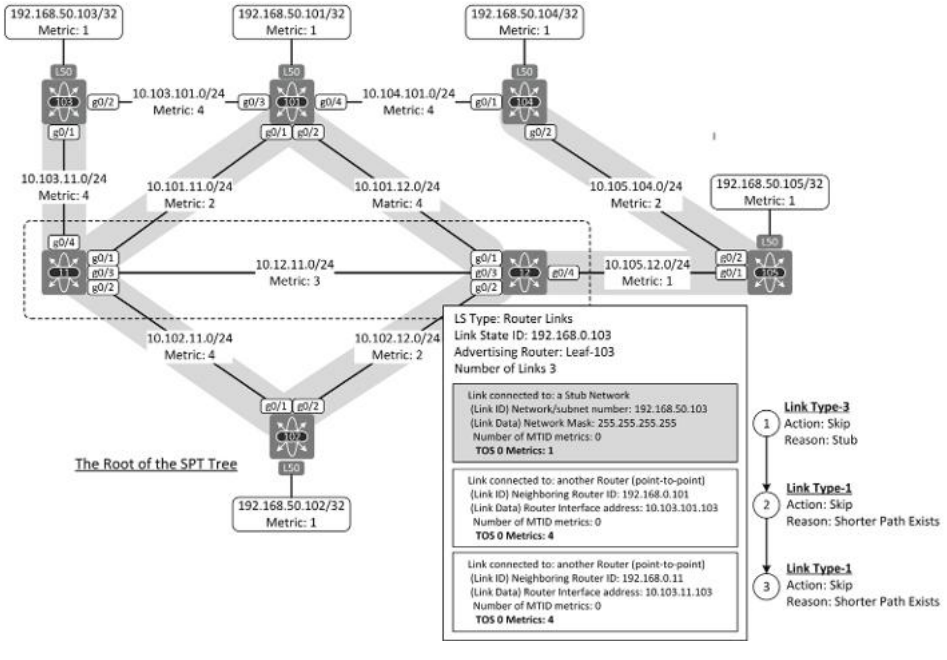


Figure 1-12: *Shortest-Path Tree calculation: Seventh iteration round on Leaf-102.*

Unknown List (Routers in LSDB)	**Candidate/Tentative List** (Dst, Cost, Next-Hop)	**Path/Known/Tree List** (Dst, Cost, Next-Hop)
		Leaf-102, 0, Leaf-102
		Spine-12, 2, Gi0/2
		Leaf-105, 3, Spine-12
		Spine-11, 4, Gi0/1
		Leaf-104, 5, Spine-12
		Leaf-101, 6, Spine-11 and Spine-12 (ECMP)
		(G) Leaf-103, 8, Spine-11

Table 1-8: *Shortest-Path Tree calculation: Seventh iteration round on Leaf-102.*

SPF Run – Phase II: Adding Leafs to Shortest-Path Tree

At first phase, Leaf-102 forms a Shortest-Path Tree (SPT) by using the Dijkstra/SPF algorithm. In the second Phase, Leaf-102 adds Stub Networks (leafs) to SPT. Leaf-102 starts by examining its self-originated Router LSA where there is a Link Description about Stub Network 192.168.50.102/32. This Stub Network is moved to Path-list (Path-list updates are not included in table 1-9 for simplicity). This information is also installed into Routing Information Base (RIB) as a connected network. Next, Leaf-102 examines the Router LSA originated by the Spine-12, which is the closest OSPF speaker found from the Path-list. There is no Link Description about Stub Networks, so neither Path List nor RIB is updated. Leaf-102 moves on to next closest OSPF speaker found from the Path List and checks the Router LSA originated by Leaf-105. The Router LSA originated by Leaf-105 includes the Stub Network 192.168.50.105/32. Leaf-102 adds it to Path-list and updates the RIB. Next, OSPF speaker on the Path-list is Spine-11, which Router LSA does not describe any Stub Network, so Leaf-102 moves on to check the Router LSA originated by Leaf-104. The Router LSA of Leaf-104 includes the Stub Network 192.168.50.104/32 which is inserted into the Path-list and into the RIB of Leaf-102. After this, Leaf-102 examines the Router LSA originated by Leaf-101. There is a Link Description about Stub Network 192.168.50.101/32. Leaf-101 is reachable via two equal-cost paths via Spine-11 and Spine-12 so the Stub Network is added into the Path-list and into the RIB with two next-hops. The last OSPF speaker listed in the Path List is Leaf-103, which has Stub Network 192.168.50.103/32 described in its Router LSA. Leaf-102 adds the Stub Network into its Path-list and into the RIB.

Now the Shortest-Path Tree with its Leafs is ready from the Leaf-102 perspective.

	Path/Known/Tree List (Dst, Cost, Next-Hop)	**Stub Networks** (Router LSA: LT-3)	**Routing Information Base** (Network [AD/C] NH Int,)
1	Leaf-102, 0, Leaf-102	192.168.50.102/32	192.168.50.102/32 directly connected
2	Spine-12, 2, Gi0/2		
3	Leaf-105, 3, Spine-12	192.168.50.105/32	192.168.50.105/32 [110/4] 10.102.12.12 via g0/2
4	Spine-11, 4, Gi0/1		
5	Leaf-104, 5, Spine-12	192.168.50.104/32	192.168.50.104/32 [110/6] 10.102.12.12 via g0/2
6	Leaf-101, 6, Spine-11 Spine-12	192.168.50.101/32	192.168.50.101/32 [110/5] 10.102.12.12 via g0/1 [110/5] 10.102.12.12 via g0/2
7	Leaf-103, 8, Spine-11	192.168.50.103/32	192.168.50.103/32 [110/9] 10.102.12.11 via g0/1

Table 1-9: *Adding leaves to Shortest-Path Tree calculation on Leaf-102.*

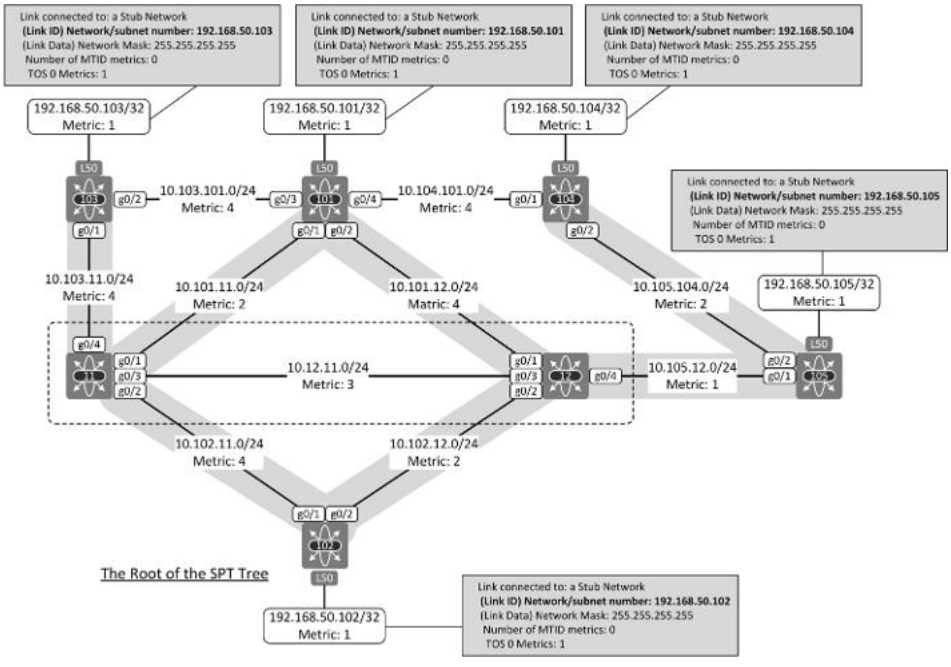


Figure 1-13: *Adding leafs to Shortest-Path Tree calculation on Leaf-102.*

So far, only the Shortest-Path Tree calculation process is introduced from the Leaf-102 perspective. All other switches in example network naturally runs the same process based on the information found from the Link State Database (LSDB). Even though the LSDBs in each OSPF speaker inside an OSPF area are identical (and they have to be) it does not mean that each OSPF speaker has an identical Shortest-Path Tree and RIB. The last table of this chapter illustrates the cost calculated by each Leaf switches. Spine switches are excluded since they do not have any Stub Networks.

To\From	Leaf-101	Leaf-102	Leaf-103	Leaf-104	Leaf-105
Leaf-101	0	6	4	4	5
Leaf-102	6	0	8	6	3
Leaf-103	4	8	0	4	9
Leaf-104	4	5	8	0	2
Leaf-105	5	3	9	2	0
192.168.50.101/32	0	7	5	5	6
192.168.50.102/32	7	0	9	7	4
192.168.50.103/32	5	9	0	5	10
192.168.50.104/32	5	6	9	0	3
192.168.50.105/32	6	4	10	3	0

Table 1-10: *Cost table between Leaf –to-Leaf and Leaf-to Stub Network.*

References:

[RFC 1195] R. Callon, “Use of OSI IS-IS for Routing in TCP/IP and Dual Environments”, RFC 1195, December 1990.

[RFC 2328] J. Moy, “OSPF Version 2”, RFC 2328, April 1998.

[RFC 6860] Y. Yang et al., “Hiding Transit-Only Networks in OSPF”, RFC 6860, January 2013.

[RFC 8365] A. Sajassi et al. “A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)”, RFC 8363, March 2018.

[ISIS-OSPF] M. Bhatia et al., “IS-IS and OSPF Difference Discussion”, Draft-bhatia-manral-isis-ospf-01” January 2006.

Chapter 2: Underlay Network – Comparison of OSPF and IS-IS

This chapter discusses the differences between the OSPF and the IS-IS from the Network Virtualization Overlay (NVO) solution, especially from the VXLAN network perspective. First, this chapter shortly introduces some of the differences between these two protocols (terminology, timers, and LSAs). Next, this chapter explains the default behavior of the Shortest Path First (SPF) by explaining first the IS-IS reaction when Stub Network goes down. Then the same event is explained from the OSPF perspective. This chapter also introduces the OSPF reaction when an *Incremental SPF (iSPF)* is enabled, and the interface on a link that is not belonging to the Shortest-Path Tree (SPT) goes down. The same event is also discussed with and without iSPF concerning IS-IS.

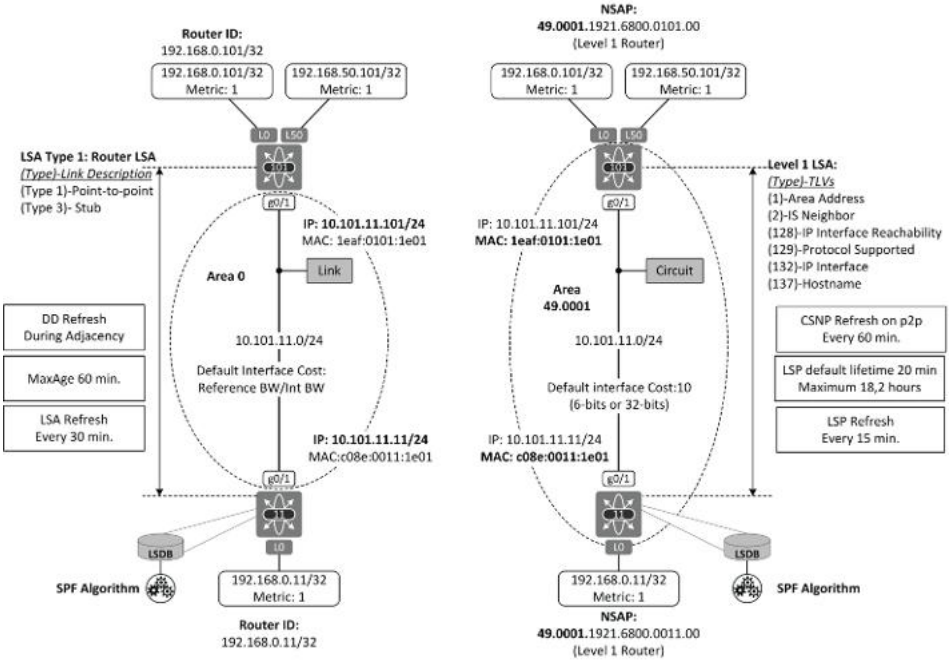


Figure 2-1: *Comparison of OSPF and IS-IS.*

Figure 2-1 illustrates some of the differences between the OSPF and the IS-IS Link-State Protocols. OSPF routers are identified by Router-Id (Usually a Loopback 0) while IS-IS routers are identified by NSAP address (Network Service Access Point). There is no relationship in the OSPF RID and OSPF Area numbering, while the first 13 bits in an NSAP address defines an IS-IS area and the next 48 bits identify a router (usually a Loopback 0). OSPF uses Link State Updates (LSU) packets, which carry LSAs (there are several different types of LSAs), which in turn describes links connected to the router. IS-IS uses a single Link State PDU (LSP), which carries different TLVs (Type, Length, Value) fields where router describes things like its hostname (RFC5301), the IS-IS area, IP addresses of links, and IPv4 prefixes that are directly connected to the router. From the router processing perspective fixed fields in OSPF LSAs do

not require as much processing effort than variable length fields in IS-IS TLVs. OSPF uses Database Description (DD) to give a shortcut of its Links State Database (LSDB) to its' peer during adjacency negotiation (Exchange state). IS-IS uses Complete Sequence Number Packet (CSNP) for the same purpose but it is refreshed every 60 minutes by default on P2P links. OSPF LSA has fixed 60 minutes lifetime that cannot be changed and each LSA is refreshed every 30 minutes by default. IS-IS LSP lifetime is 20 minutes by default and it can change all the way up to 18.2 hours. IS-IS LSPs are refreshed every 15 minutes by default and it launches the SPF algorithm (range 1-65535 seconds). To form an OSPF adjacency (and maintain it) between OSPF speaker the following attributes in Hello-packets must match; area-id, timers (hello and dead interval), authentication, network mask, stub-flags, options, Interface MTU and OSPF network type. In addition, the interface IP address and OSPF router-id has to be unique in the adjacent router. While OSPF has nine attributes that have to match, IS-IS only requires four matching attributes: Interface MTU, Levels, The Area-Id in Level 1 router and authentication. Also, System-Id has to be unique. Among these things, there are differences in area structure. OSPF area boundary falls on the router while in IS-IS the link is the border. The default metric in OSPF is calculated based on defined reference bandwidth/Interface bandwidth while in IS-IS the default interface metric is 10. In addition, IS-IS can signal memory overload by using "overload-bit", OSPF does not have this option. There are many other differences between the OSPF and the IS-IS but there are also similarities. All router inside an area, whether we are speaking about OSPF or IS-IS, has to have a common LSDB. Both protocols use the SPF algorithm to form a loop-free Shortest-Path Tree (SPT) between each router inside an area as well as adding leafs (networks) to SPT. Based on the calculation result, routers also updates their Routing Information Bases (RIB) as well as Forwarding Information Bases (FIB) if there are no better route sources like static routes available.

Figure 2-2 shows the example topology where both OSPF and IS-IS are running on each router and both protocols calculate the Shortest-Path (Links participating SPT are highlighted with grey color). The topology is based on the Spine-Leaf model with the difference that there is a link between the two Spine switches. The purpose of Spine-to-Spine link is to demonstrate the SPF process when the link that is not participating in Shortest-Path Tree goes down. This kind of cabling structure might be relevant in some enterprise networks. There are six scenarios in this chapter. First, this chapter explains the operation of the SPF algorithm from the IS-IS perspective when the loopback 50 interface on Leaf-101 goes down (Partial SPF). Then the same event is monitored from the OSPF perspective (Full SPF). Next, this section shows how the OSPF SPF algorithm can be optimized (Incremental SPF). Hence, this section explains how OSPF reacts when the link between Spine-11 and Spine-12 goes down when iSPF is enabled. The last section explains the same operation from the IS-IS perspective and also introduces the IS-IS SPF optimization using iSPF.

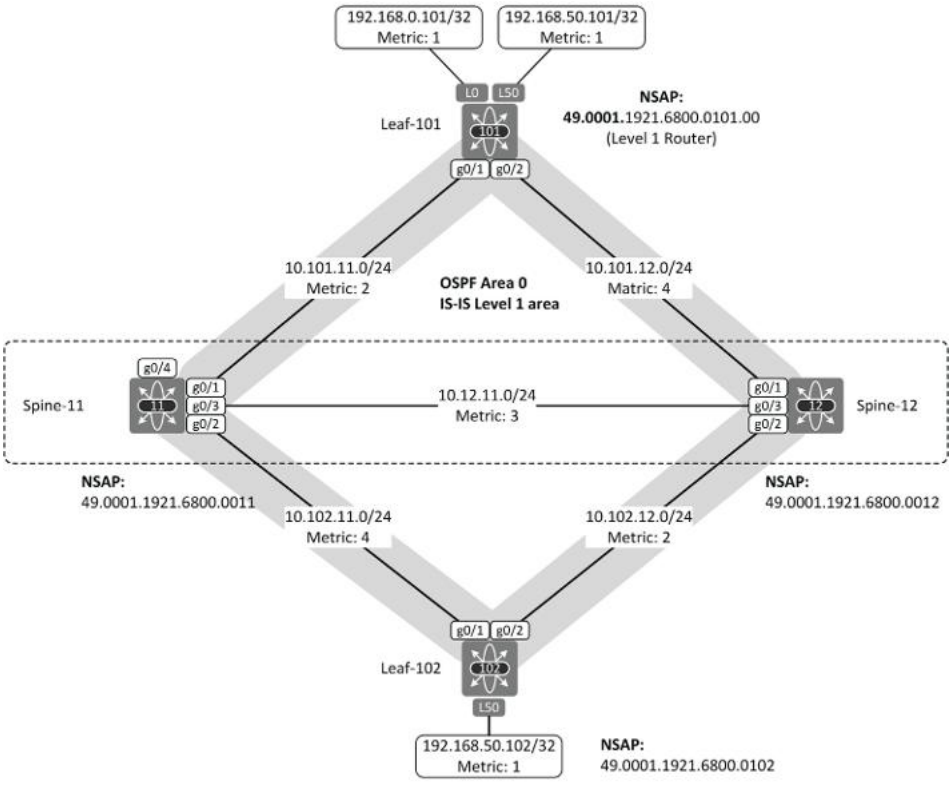


Figure 2-2: *Example topology.*

Scenario-1: Interface loopback 50 down on Leaf-101 (IS-IS)

Figure 2-3 illustrates the situation where the Loopback 50 interface (192.168.50.101/32) state is moved from Up to Down on Leaf-101. As a reaction to this event, Leaf-101 generates a new LSP and sends it to IS-IS neighbors. The "*IP Internal Reachability Information*" (Type-128) describes the network that is directly connected to the IS-IS router. The IP network 192.168.50.101 is excluded from this LSP. The packet capture is taken from the interface g0/1 on Leaf-101. LSP is flooded to Leaf-102 by both Spine switches. When Leaf-102 receives the LSP, instead of executing full SPF run by processing each LSA from its LSDB it only processes the new LSP. Leaf-102 ages out the old LSP from the LSDB and installs the information received on the newest LSP into LSDB. Also, it removes routes describing the reachability of net 192.168.50.101 from its RIB and FIB. IS-IS does not run the full SPF where the whole SPT is recalculated when the Leaf route is changed. This is why the term *Partial SPF* is used when speaking about the SPF run concerning IS-IS.

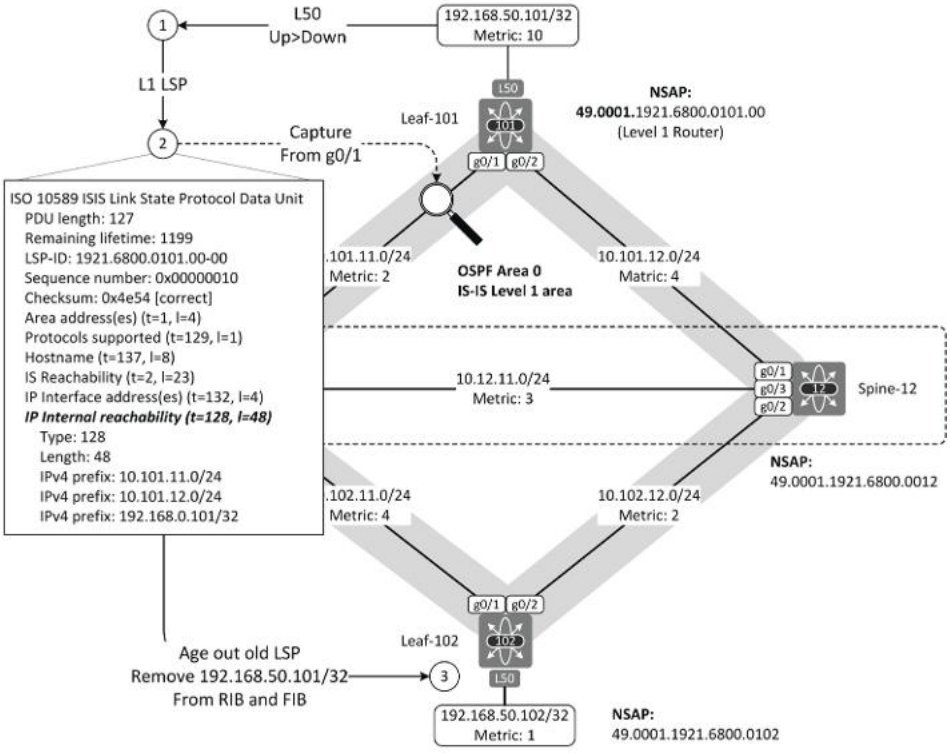


Figure 2-3: *Interface Loopback 50 DOWN on Leaf-101 (IS-IS).*

Example 2-1 shows the debug information in Leaf-102 when it receives the new LSP originated by Leaf-101.

```
Leaf-102#debug isis spf-events
IS-IS SPF events debugging is on for IPv4 unicast topology base
ISIS-SPF: L1 LSP 3 (1921.6800.0101.00-00) flagged for recalculation
from 252321F
ISIS-SPF: LSP 3 (1921.6800.0101.00-00) Type STD
ISIS-SPF: spf_result: next_hop_parents:0x1050133C root_distance:6,
parent_count:2, parent_index:4 db_on_paths:1
ISIS-SPF: Calculating routes for L1 LSP 3 (1921.6800.0101.00-00)
ISIS-SPF: lsptype:0, current_lsp(1921.6800.0101.00-00)(3)
current_lsp:0xDABACA8, lsp_fragment:0xDABACA8 calling isis_walk_lsp
ISIS-SPF: Aging L1 LSP 3 (1921.6800.0101.00-00), version 15
```

Example 2-1: "*Debug isis spf-events" on Leaf-102*

Example 2-2 shows the LSDB concerning information received from Leaf-101. NLPID (Network Layer Protocol Identifier) describes the supported protocols (0xCC = IPv4). NLPID is shown in packet capture as "Supported Protocols" in figure 2-3. Also, the hostname of the originating router is carried in TLVs. To be able to see hostname in OSPF LSDB, hostname-to-IP address mapping information has to be configured to each router.

```
Leaf-102#show isis database level-1 Leaf-101.00-00 detail
<snipped>
Leaf-101.00-00        0x00000010   0x4E54                1178/1198      0/0/0
  Area Address: 49.0001
  NLPID:        0xCC
  Hostname: Leaf-101
  Metric: 4          IS Spine-12.00
  Metric: 2          IS Spine-11.00
  IP Address:   192.168.0.101
  Metric: 2          IP 10.101.11.0 255.255.255.0
  Metric: 4          IP 10.101.12.0 255.255.255.0
  Metric: 4          IP 10.103.101.0 255.255.255.0
  Metric: 0          IP 192.168.0.101 255.255.255.255
```

Example 2-2: "*show isis database level-1 Leaf-101.00-00 detail" on Leaf-102*

Scenario-2: Interface loopback 50 down on Leaf-101 (OSPF)

Figure 2-4 describes the same scenario from the OSPF perspective. As soon as the interface Loopback 50 state is changed from Up to Down, Leaf-101 generates a Link State Update (LSU) packet that carries a Router-LSA including information about current usable links. Since interface Loopback 50 is down, there are only two P2P (type-1) link descriptions describing the connection to Spine switches. When Leaf-102 receives the LSU it notifies that this is the newest Router-LSA. Leaf-102 then executes the full Shortest-Path Tree calculation even though the core of the tree has not changed.

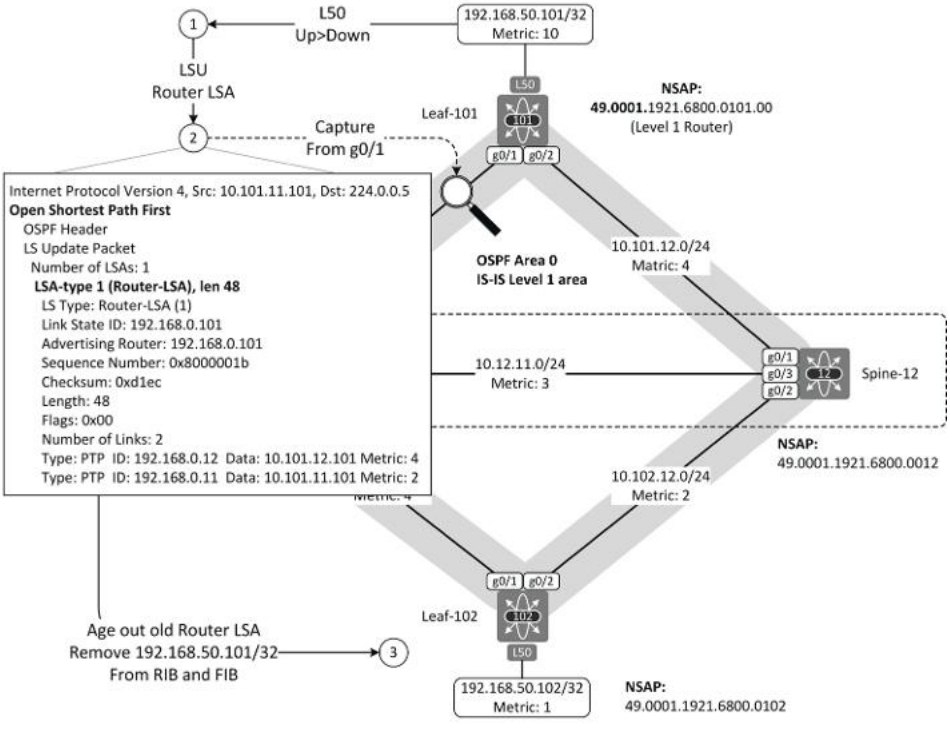


Figure 2-4: *Interface Loopback 50 DOWN on Leaf-101 (OSPF).*

The first highlighted (black and white) shows the start of the SPF algorithm. Leaf -102 re-builds the whole SPT by going through each router-LSA found on its LSDB. It moves itself as a root for the SPT. Then it finds out the closest OSPF peer which is Spine-12 and processes LSA originated by the Spine-12.

```
Leaf-102#debug ip ospf spf intra
OSPF SPF intra debugging is on
04:50:38: OSPF-1 SPF  : Detect change in LSA type 1, LSID 192.168.0.101 from
192.168.0.101 area 0
04:50:43: OSPF-1 MON  : Begin SPF at 17443.145ms, process time 19052ms
04:50:43: OSPF-1 INTRA: Running SPF for area 0, SPF-type Full
04:50:43: OSPF-1 INTRA: Initializing to run spf
04:50:43: OSPF-1 INTRA: spf intra() - rebuilding the tree
04:50:43: OSPF-1 INTRA:  It is a router LSA 192.168.0.102. Link Count 2
04:50:43: OSPF-1 INTRA:   Processing link 0, id 192.168.0.12, link data
10.102.12.102, type 1
04:50:43: OSPF-1 SPF  :    Add better path to LSA ID 192.168.0.12, gateway
10.102.12.12, dist 2
04:50:43: OSPF-1 INTRA:    Putting LSA on the clist LSID 192.168.0.12, Type 1,
Adv Rtr. 192.168.0.12
04:50:43: OSPF-1 SPF  :    Add path: next-hop 10.102.12.12, interface
GigabitEthernet0/2
04:50:43: OSPF-1 INTRA:   Processing link 1, id 192.168.0.11, link data
10.102.11.102, type 1
04:50:43: OSPF-1 SPF  :    Add better path to LSA ID 192.168.0.11, gateway
10.102.11.11, dist 4
04:50:43: OSPF-1 INTRA:    Putting LSA on the clist LSID 192.168.0.11, Type 1,
Adv Rtr. 192.168.0.11
04:50:43: OSPF-1 INTRA:      Upheap LSA ID 192.168.0.11, Type 1, Adv
192.168.0.11 on clist from index 2 to 2
04:50:43: OSPF-1 SPF  :    Add path: next-hop 10.102.11.11, interface
GigabitEthernet0/1
04:50:43: OSPF-1 INTRA:      Downheap LSA ID 192.168.0.11, Type 1, Adv
192.168.0.11 on clist from index 1 to 1
04:50:43: OSPF-1 INTRA:  It is a router LSA 192.168.0.12. Link Count 3
04:50:43: OSPF-1 INTRA:   Processing link 0, id 192.168.0.11, link data
10.12.11.12, type 1
04:50:43: OSPF-1 INTRA:   Ignore newdist 5 olddist 4
04:50:43: OSPF-1 INTRA:   Processing link 1, id 192.168.0.102, link data
10.102.12.12, type 1
04:50:43: OSPF-1 INTRA:   Ignore newdist 4 olddist 0
04:50:43: OSPF-1 INTRA:   Processing link 2, id 192.168.0.101, link data
10.101.12.12, type 1
04:50:43: OSPF-1 SPF  :    Add better path to LSA ID 192.168.0.101, gateway
10.101.12.101, dist 6
04:50:43: OSPF-1 INTRA:    Putting LSA on the clist LSID 192.168.0.101, Type 1,
Adv Rtr. 192.168.0.101
04:50:43: OSPF-1 INTRA:      Upheap LSA ID 192.168.0.101, Type 1, Adv
192.168.0.101 on clist from index 2 to 2
04:50:43: OSPF-1 SPF  :    Add path: next-hop 10.102.12.12, interface
GigabitEthernet0/2
04:50:43: OSPF-1 INTRA:      Downheap LSA ID 192.168.0.101, Type 1, Adv
192.168.0.101 on clist from index 1 to 1
04:50:43: OSPF-1 INTRA:  It is a router LSA 192.168.0.11. Link Count 3
04:50:43: OSPF-1 INTRA:   Processing link 0, id 192.168.0.12, link data
10.12.11.11, type 1
04:50:43: OSPF-1 INTRA:   Ignore newdist 7 olddist 2
04:50:43: OSPF-1 INTRA:   Processing link 1, id 192.168.0.102, link data
10.102.11.11, type 1
```

```
04:50:43: OSPF-1 INTRA:   Ignore newdist 8 olddist 0
04:50:43: OSPF-1 INTRA:   Processing link 2, id 192.168.0.101, link data
10.101.11.11, type 1
04:50:43: OSPF-1 INTRA:   Add equal-length path to 192.168.0.101, dist 6
04:50:43: OSPF-1 INTRA:    LSA already on the clist LSID 192.168.0.101, Type 1,
Adv Rtr. 192.168.0.101
04:50:43: OSPF-1 SPF  :    Add path: next-hop 10.102.11.11, interface
GigabitEthernet0/1
04:50:43: OSPF-1 INTRA:      Downheap LSA ID 192.168.0.101, Type 1, Adv
192.168.0.101 on clist from index 1 to 1
04:50:43: OSPF-1 INTRA:  It is a router LSA 192.168.0.101. Link Count 2
04:50:43: OSPF-1 INTRA:   Processing link 0, id 192.168.0.12, link data
10.101.12.101, type 1
04:50:43: OSPF-1 INTRA:   Ignore newdist 10 olddist 2
04:50:43: OSPF-1 INTRA:   Processing link 1, id 192.168.0.11, link data
10.101.11.101, type 1
04:50:43: OSPF-1 INTRA:   Ignore newdist 8 olddist 4
04:50:43: OSPF-1 INTRA: Adding Stub nets
04:50:43: OSPF-1 INTRA: Entered intra-area route sync for area 0
04:50:43: OSPF-1 INTRA: Entered intra-area route sync for area 0
04:50:43: OSPF-1 MON  : End SPF at 17443.159ms, Total elapsed time 14ms
```

Example 2-3: "*debug ip ospf spf intra" on Leaf-102.*

Scenario-3: OSPF Incremental SPF – L50 Down on Leaf-101 (Stub)

OSPF SPT calculation can be optimized by using incremental SPF. The configuration is simple; the command **ispf** is added under the OSPF process. Incremental SPF is activated and SPF is run right after the command.

Example 2-4 illustrates the reaction of Leaf-102 when the interface Loopback 50 goes down on Leaf-101 and the iSPF is enabled on Leaf-102. Instead of running a full SPF, Leaf-102 runs Incremental SPF because the only change compared to the old and new LSA originated by Leaf-101 concerns Stub Network.

```
Leaf-102#debug ip ospf spf intra
OSPF SPF intra debugging is on
06:13:33: OSPF-1 SPF  : Detect change in LSA type 1, LSID 192.168.0.101 from
192.168.0.101 area 0
06:13:33: OSPF-1 INTRA: Insert LSA to New_LSA list type 1, LSID 192.168.0.101,
from 192.168.0.101 area 0
06:13:38: OSPF-1 MON  : Begin SPF at 22418.763ms, process time 23574ms
06:13:38: OSPF-1 INTRA: Running SPF for area 0, SPF-type Incremental
06:13:38: OSPF-1 INTRA: Initializing to run spf
06:13:38: OSPF-1 INTRA: Running incremental SPF for area 0
06:13:38: OSPF-1 INTRA: iSPF: Processing node 1/192.168.0.101/192.168.0.101
from the New List
06:13:38: OSPF-1 INTRA: iSPF:  Checking parents
06:13:38: OSPF-1 INTRA: iSPF:   trying to find a link to parent
1/192.168.0.11/192.168.0.11 in the new LSA
06:13:38: OSPF-1 INTRA: iSPF:    ...found
06:13:38: OSPF-1 INTRA: iSPF:   trying to find a link to parent
1/192.168.0.12/192.168.0.12 in the new LSA
06:13:38: OSPF-1 INTRA: iSPF:    ...found
06:13:38: OSPF-1 INTRA: iSPF:  No change in parents
06:13:38: OSPF-1 INTRA: iSPF: Scanning new LSA of node
1/192.168.0.101/192.168.0.101
```

```
06:13:38: OSPF-1 INTRA: iSPF:    node 1/192.168.0.101/192.168.0.101 is NOT a
parent of node 1/192.168.0.12/192.168.0.12
06:13:38: OSPF-1 INTRA: iSPF:    node 1/192.168.0.12/192.168.0.12 is parent of
node 1/192.168.0.101/192.168.0.101
06:13:38: OSPF-1 INTRA: iSPF:    node 1/192.168.0.101/192.168.0.101 is NOT a
parent of node 1/192.168.0.11/192.168.0.11
06:13:38: OSPF-1 INTRA: iSPF:    node 1/192.168.0.11/192.168.0.11 is parent of
node 1/192.168.0.101/192.168.0.101
06:13:38: OSPF-1 INTRA: iSPF:    init all stub routes on delete list of node
1/192.168.0.101/192.168.0.101
06:13:38: OSPF-1 INTRA: iSPF: initializing node 0/192.168.50.101/192.168.0.101
06:13:38: OSPF-1 INTRA: iSPF:    process all stub-routes of node
1/192.168.0.101/192.168.0.101
06:13:38: OSPF-1 INTRA: iSPF: Checking lost links of node
1/192.168.0.101/192.168.0.101
06:13:38: OSPF-1 INTRA: iSPF: Re-attaching nodes on orphans
06:13:38: OSPF-1 INTRA: spf_intra() - rebuilding the tree
06:13:38: OSPF-1 INTRA: Adding Stub nets
06:13:38: OSPF-1 INTRA: Entered intra-area route sync for area 0
06:13:38: OSPF-1 INTRA: Entered intra-area route sync for area 0
06:13:38: OSPF-1 MON  : End SPF at 22418.771ms, Total elapsed time 8ms
```

Example 2-4: "*debug ip ospf spf intra" on Leaf-102.*

Scenario-4: OSPF Incremental SPF – Interface g0/3 Down on Spine-12 (transit link does not participate in SPT)

Incremental SPF (iSPF) also works with the transit links that are not part of the core of the SPT. When the interface g0/3 in Spine-12 goes down, Spine-12 will originate a new LSU packet where it sends fresh link descriptions. Since the link does not belong to SPT from the Leaf-102 perspective, Leaf-102 does not calculate the whole tree, it just processes the router LSA received from Spine-11 and adjust the LSDB based on it.

```
Leaf-102#debug ip ospf spf intra
OSPF SPF intra debugging is on
06:41:37: OSPF-1 SPF  : Detect change in LSA type 1, LSID 192.168.0.12 from
192.168.0.12 area 0
06:41:37: OSPF-1 INTRA: Insert LSA to New_LSA list type 1, LSID 192.168.0.12,
from 192.168.0.12 area 0
06:41:42: OSPF-1 MON  : Begin SPF at 24102.468ms, process time 25012ms
06:41:42: OSPF-1 INTRA: Running SPF for area 0, SPF-type Incremental
06:41:42: OSPF-1 INTRA: Initializing to run spf
06:41:42: OSPF-1 INTRA: Running incremental SPF for area 0
06:41:42: OSPF-1 INTRA: iSPF: Processing node 1/192.168.0.12/192.168.0.12 from
the New List
06:41:42: OSPF-1 INTRA: iSPF:  Checking parents
06:41:42: OSPF-1 INTRA: iSPF:   trying to find a link to parent
1/192.168.0.102/192.168.0.102 in the new LSA
06:41:42: OSPF-1 INTRA: iSPF:    ...found
06:41:42: OSPF-1 INTRA: iSPF:  No change in parents
06:41:42: OSPF-1 INTRA: iSPF: Scanning new LSA of node
1/192.168.0.12/192.168.0.12
06:41:42: OSPF-1 INTRA: iSPF:    node 1/192.168.0.12/192.168.0.12 is NOT a
parent of node 1/192.168.0.102/192.168.0.102
06:41:42: OSPF-1 INTRA: iSPF:    node 1/192.168.0.102/192.168.0.102 is parent
of node 1/192.168.0.12/192.168.0.12
```

```
06:41:42: OSPF-1 INTRA: iSPF:    node 1/192.168.0.102/192.168.0.102 is parent
of node 1/192.168.0.12/192.168.0.12
06:41:42: OSPF-1 INTRA: iSPF:    node 1/192.168.0.12/192.168.0.12 is parent of
node 1/192.168.0.101/192.168.0.101
06:41:42: OSPF-1 INTRA: iSPF:    init all stub routes on delete list of node
1/192.168.0.12/192.168.0.12
06:41:42: OSPF-1 INTRA: iSPF:    process all stub-routes of node
1/192.168.0.12/192.168.0.12
06:41:42: OSPF-1 INTRA: iSPF: Checking lost links of node
1/192.168.0.12/192.168.0.12
06:41:42: OSPF-1 INTRA: iSPF:    node 1/192.168.0.12/192.168.0.12 is NOT a
parent of node 1/192.168.0.11/192.168.0.11
06:41:42: OSPF-1 INTRA: iSPF: Re-attaching nodes on orphans
06:41:42: OSPF-1 INTRA: spf_intra() - rebuilding the tree
06:41:42: OSPF-1 INTRA: Adding Stub nets
06:41:42: OSPF-1 INTRA: Entered intra-area route sync for area 0
06:41:42: OSPF-1 INTRA: Entered intra-area route sync for area 0
06:41:42: OSPF-1 MON  : End SPF at 24102.474ms, Total elapsed time 6ms
```

Example 2-5: "*debug ip ospf spf intra" on Leaf-102.*

Scenario-5: IS-IS SPF – Interface g0/3 Down on Spine-12 (Full SPF computation)

Example 2-6 explains how ISIS reacts when interface g0/3 on Spine-12 goes down. Leaf-101 runs a full SPF and re-builds an SPT. When comparing debug output received from IS-IS to OSPF, the IS-IS is a bit more descriptive. The Output clearly shows how the Leaf-102 installs itself from the TENT list into PATH list. Then it moves its own IS-IS peers to the TENT list. Then Leaf-102 removes the peer with the lowest metric from the TENT list, and adds it into the PATH list. As a next step, Spine-12 peer Leaf-101 is moved into the TENT list (which already includes Spine-11). Then, the metric of these two are compared and the Spine-11 is moved into PATH list based on the lowest metric value. The process moves on and it is stopped when the TENT list is empty.

```
08:45:33: ISIS-SPF:  Compute L1 SPT
08:45:33: ISIS-Stats:  Compute L1 SPT
08:45:33: ISIS-SPF: Starting level-1 SPF with 1 nodes into TENT
08:45:33: ISIS-SPF: Move 1921.6800.0102.00-00 to PATHS, metric 0
08:45:33: ISIS-SPF: Remove the node from the TENT list lsp(1921.6800.0102.00-
00)(1):0xF584E10
08:45:33: ISIS-SPF: Attempt to add each adj of the node to tent via add lsp
routes, lsp(1921.6800.0102.00-00)(1), lsptype:0
08:45:33: ISIS-SPF: lsptype:0, current_lsp(1921.6800.0102.00-00)(1)
current_lsp:0xF584E10, lsp_fragment:0xF584E10 calling isis_walk_lsp
08:45:33: ISIS-SPF: Added neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: considering adj to 1921.6800.0012 (GigabitEthernet0/2)
metric 2, level 1, circuit 3, adj 1
08:45:33: ISIS-SPF:   (accepted)
08:45:33: ISIS-Spf: Add 1921.6800.0012.00-00 to TENT, metric 2
08:45:33: ISIS-Spf:   Next hop 1921.6800.0012 (GigabitEthernet0/2)
08:45:33: ISIS-SPF: Added neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: considering adj to 1921.6800.0011 (GigabitEthernet0/1)
metric 4, level 1, circuit 3, adj 1
08:45:33: ISIS-SPF:   (accepted)
08:45:33: ISIS-Spf: Add 1921.6800.0011.00-00 to TENT, metric 4
08:45:33: ISIS-Spf:   Next hop 1921.6800.0011 (GigabitEthernet0/1)
08:45:33: ISIS-SPF: Move 1921.6800.0012.00-00 to PATHS, metric 2
```

```
08:45:33: ISIS-SPF: Remove the node from the TENT list lsp(1921.6800.0012.00-
00)(2):0xFEF85C8
08:45:33: ISIS-SPF: Attempt to add each adj of the node to tent via add lsp
routes, lsp(1921.6800.0012.00-00)(2), lsptype:0
08:45:33: ISIS-SPF: lsptype:0, current_lsp(1921.6800.0012.00-00)(2)
current_lsp:0xFEF85C8, lsp_fragment:0xFEF85C8 calling isis_walk_lsp
08:45:33: ISIS-SPF: Failed to add neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: Added neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-Spf: Add 1921.6800.0101.00-00 to TENT, metric 6
08:45:33: ISIS-Spf:   Next hop 1921.6800.0012 (GigabitEthernet0/2)
08:45:33: ISIS-SPF: Move 1921.6800.0011.00-00 to PATHS, metric 4
08:45:33: ISIS-SPF: Remove the node from the TENT list lsp(1921.6800.0011.00-
00)(4):0xDABACA8
08:45:33: ISIS-SPF: Attempt to add each adj of the node to tent via add lsp
routes, lsp(1921.6800.0011.00-00)(4), lsptype:0
08:45:33: ISIS-SPF: lsptype:0, current_lsp(1921.6800.0011.00-00)(4)
current_lsp:0xDABACA8, lsp_fragment:0xDABACA8 calling isis_walk_lsp
08:45:33: ISIS-SPF: Added neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-Spf: Add 1921.6800.0101.00-00 to TENT, metric 6
08:45:33: ISIS-Spf:   Next hop 1921.6800.0011 (GigabitEthernet0/1)
08:45:33: ISIS-Spf:   Next hop 1921.6800.0012 (GigabitEthernet0/2)
08:45:33: ISIS-SPF: Failed to add neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: Failed to add neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: Move 1921.6800.0101.00-00 to PATHS, metric 6
08:45:33: ISIS-SPF: Remove the node from the TENT list lsp(1921.6800.0101.00-
00)(3):0xDABB208
08:45:33: ISIS-SPF: Attempt to add each adj of the node to tent via add lsp
routes, lsp(1921.6800.0101.00-00)(3), lsptype:0
08:45:33: ISIS-SPF: lsptype:0, current_lsp(1921.6800.0101.00-00)(3)
current_lsp:0xDABB208, lsp_fragment:0xDABB208 calling isis_walk_lsp
08:45:33: ISIS-SPF: Failed to add neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: Failed to add neighbor lsp to the tent list, link_is_p2p:1
08:45:33: ISIS-SPF: Aging L1 LSP 1 (1921.6800.0102.00-00), version 41
08:45:33: ISIS-SPF: Aging L1 LSP 2 (1921.6800.0012.00-00), version 43
08:45:33: ISIS-SPF: Aging L1 LSP 3 (1921.6800.0101.00-00), version 64
08:45:33: ISIS-SPF: Aging L1 LSP 4 (1921.6800.0011.00-00), version 44
08:45:33: ISIS-Stats: SPF only compute time 0.014
08:45:33: ISIS-Stats: IPv4 RIB only compute time 0.000
08:45:33: ISIS-Stats: Complete L1 SPT,
08:45:33: ISIS-Stats:  Compute time 0.014/0.014, 4/0 nodes, 4/0 links, 0
suspends
```

Example 2-6: *debug isis sfp-events on Leaf-102.*

Scenario-6: IS-IS Incremental SPF – Interface g0/3 Down on Spine-12 (transit link does not participate in SPT)

It is also possible to optimize the SPF algorithm in IS-IS by using Incremental SPF. The configuration is simple; add the command **ispf level-1 20** under the IS-IS routing process (default value when beginning iSPF is 120 seconds). Unlike in OSPF, IS-IS configuration allows the administrator to define when the Incremental SPF is activated (when activated, SPF is done immediately).

Example 2-7 describes the SPF process in Leaf-102 when the incremental SPF is enabled and the interface g0/3 on Spine-12 state changes from up to down. Note that there is a new LSP packet received from both Spine-11 (Adding LSP: 4) and Spine-12 (Adding LSP: 2). For simplicity processing of LSA received from Spine-12 is highlighted with grey color.

```
08:51:41: ISIS-SPF:  Compute L1 SPT
08:51:41: ISIS-Stats:  Compute L1 SPT
08:51:41: ISIS-Stats:
ISIS-Stats: Starting incremental SPF for level-1
08:51:41: ISIS-SPF: I-SPF: Adding LSP: (2) to NewLSP, metric: 2
08:51:41: ISIS-SPF: I-SPF: Adding LSP: (2) to 1035B410, metric: 2. From 24FD63D
08:51:41: ISIS-SPF: I-SPF: Adding LSP: (4) to NewLSP, metric: 4
08:51:41: ISIS-SPF: I-SPF: Adding LSP: (4) to 1035B410, metric: 4. From 24FD63D
08:51:41: ISIS-SPF: I-SPF: Entering in read_lsp with node: (2)
08:51:41: ISIS-SPF: I-SPF: Delta distance to (1) is 4
08:51:41: ISIS-SPF: I-SPF: Delta distance to (3) is 0
08:51:41: ISIS-SPF: I-SPF: Searching (2) orphans
08:51:41: ISIS-SPF: L1 LSP 2 (1921.6800.0012.00-00) flagged for recalculation
from 2500635
08:51:41: ISIS-SPF: I-SPF: Entering in read_lsp with node: (4)
08:51:41: ISIS-SPF: I-SPF: Delta distance to (3) is 0
08:51:41: ISIS-SPF: I-SPF: Delta distance to (1) is 8
08:51:41: ISIS-SPF: I-SPF: Delta distance to (2) is 5
08:51:41: ISIS-SPF: I-SPF: Searching (4) orphans
08:51:41: ISIS-SPF: L1 LSP 4 (1921.6800.0011.00-00) flagged for recalculation
from 2500635
08:51:41: ISIS-SPF: I-SPF: Entering isis_ispf_reattach_node
08:51:41: ISIS-SPF: Aging L1 LSP 1 (1921.6800.0102.00-00), version 44
08:51:41: ISIS-SPF: Aging L1 LSP 2 (1921.6800.0012.00-00), version 46
08:51:41: ISIS-SPF: Aging L1 LSP 3 (1921.6800.0101.00-00), version 67
08:51:41: ISIS-SPF: Aging L1 LSP 4 (1921.6800.0011.00-00), version 47
08:51:41: ISIS-Stats: SPF only compute time 0.007
08:51:41: ISIS-Stats: IPv4 RIB only compute time 0.000
08:51:41: ISIS-Stats: Complete L1 SPT,
08:51:41: ISIS-Stats:  Compute time 0.007/0.007, 0/0 nodes, 0/0 links, 0
suspends
```

Example 2-7: "*debug isis sfp-events" on Leaf-102.*

Conclusion

SPF algorithm on both OSPF and IS-IS is possible to optimize by using Incremental SPF, which means that when either Stub Network or non-SPT Transit Network goes down, routers do not have to run full SPF. Though when any link/network comes up, the full SPF is calculated by both protocols.

When discussing which Link-State Protocol is better for the Underlay Network from the VXLAN perspective, there is no clear answer. Some IS-IS properties like adjustable LSP lifetime, the new property requires only a new TLV (not totally new LSA type) and fewer attributes that have to match to form and keep an IS-IS adjacency up might make IS-IS a bit better than OSPF. The question is; are these properties relevant from the VXLAN fabric Underlay Network perspective, at the end of the day there are only couple of loopback addresses (if unnumbered uplinks are used) per router that needs to be reachable throughout the infrastructure. Though it is possible that there are hundreds or even thousands of switches in VXLAN fabric. Both OSPF and IS-IS are suitable for Underlay Network (Authors opinion). Use the one that you know better.

References:

[RFC 1195] R. Callon, “Use of OSI IS-IS for Routing in TCP/IP and Dual Environments”, RFC 1195, December 1990.

[RFC 2328] J. Moy, “OSPF Version 2”, RFC 2328, April 1998.

[RFC 4970] A. Lindem et al., “Extensions to OSPF for Advertising Optional Router Capabilities”, RFC 4970, July 2007.

[RFC 5301] D. McPherson et al., “Dynamic Hostname Exchange Mechanism for IS-IS”, RFC 5301, October 2008.

[RFC 5340] R. Coltun et al., “OSPF for IPv6”, RFC 5340, July 2008.

[RFC 6860] Y. Yang et al., “Hiding Transit-Only Networks in OSPF”, RFC 6860, January 2013.

[RFC 8365] A. Sajassi et al. “A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)”, RFC 8363, March 2018.

[ISIS-OSPF] M. Bhatia et al., “IS-IS and OSPF Difference Discussion”, Draft-bhatia-manral-isis-ospf-01” January 2006

Chapter 3: Underlay Network: iBGP in Underlay Network

Using BGP instead of OSPF or IS-IS for Underlay Network routing in BGP VXLAN fabric simplifies the Control Plane operation because there is only one routing protocol running on fabric switches. However, there are some tradeoffs too. The BGP only solution requires a minimum of two different BGP Address-Families (afi) per switch, one for the Underlay (IPv4 Unicast) and one for the Overlay (L2VPN EVPN). In addition, if Border Leaf switches are connected to the MPLS network, there is a third BGP afi for VPNv4. In some cases, multi-afi BGP makes troubleshooting a bit more complex compared to a single-afi solution where BGP is used only in Overlay Network. The focus of this chapter is VXLAN fabric Underlay Network with iBGP routing.

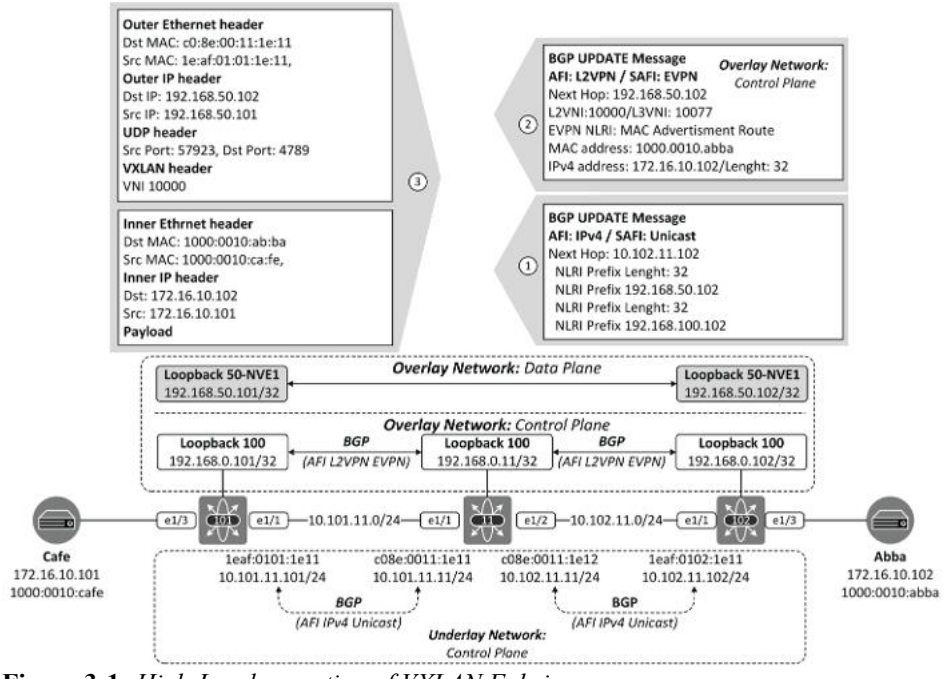


Figure 3-1: *High-Level operation of VXLAN Fabric*

Figure 3-1 illustrates a high-level operation of a Control and Data Plane in VXLAN fabric. *First*, there is an address-family (afi) IPv4 Unicast BGP peering between the physical interfaces. Switches exchanges BGP Update packets, which carries an IPv4 Network Layer Reachability Information (IPv4 NLRI) about their logical Loopback Interfaces 100 (Overlay Control Plane-BGP) and 50 (Overlay Data Plane-VXLAN). This is a part of the Underlay Network operation. *Second*, all switches have an afi L2VPN EVPN peering between their Loopback 100 interfaces. Leaf switches advertise their EVPN specific NLRIs such as local hosts MAC/IP address with L2/L3 Virtual Network Identifier (VNI) over this peering. *Third*, Data between the hosts belonging to the same Layer-2 segment but located in different Leaf switches are sent over the interface NVE1 (Network Virtualization Edge) with VXLAN encapsulation. The IP address of an NVE1 interface is taken from the interface Loopback 50 and it is used for VXLAN tunneling between VTEP switches.

Route Propagation

Figure 3-2 describes the BGP processes from the Underlay Network perspective focusing on the propagation of Loopback 100 and 50 interfaces connected to Leaf-102.

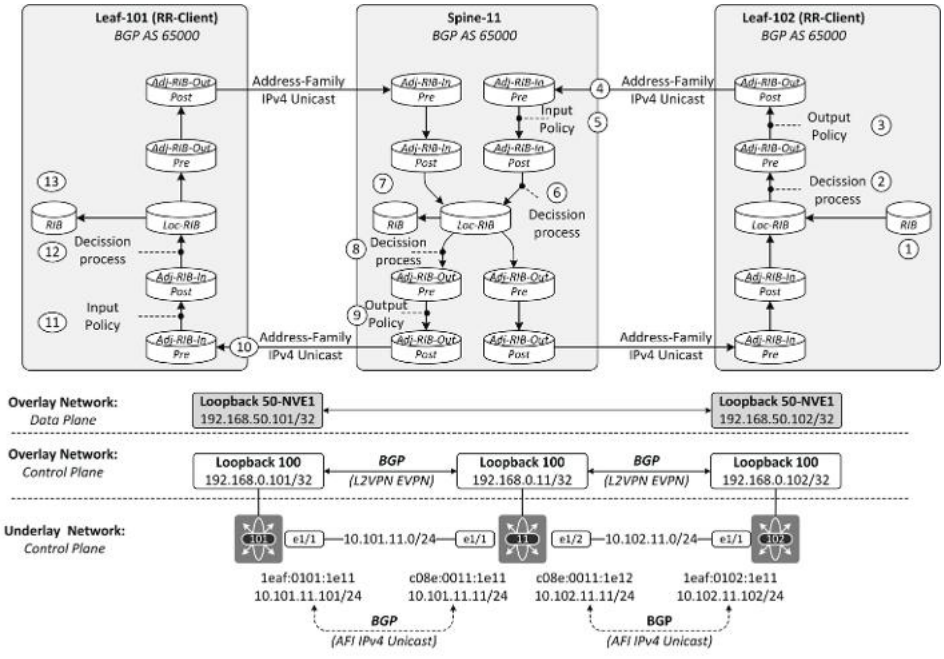


Figure 3-2: *Underlay BGP process.*

Step-1: Leaf-102 injects the IP addresses of Loopback 100 (192.168.100.102) and 50 (192.168.50.102) into the BGP table called *Loc-RIB*. This can be done for example by redistributing connected networks through the route-map or by using the network –clause under the BGP process.

```
Leaf-102# sh ip bgp | be Network
   Network            Next Hop   Metric LocPrf   Weight Path
*>i192.168.50.101/32  10.102.11.11       100      0       i
*>l192.168.50.102/32  0.0.0.0            100 32768       i
*>i192.168.100.11/32  10.102.11.11       100      0       i
*>i192.168.100.101/32 10.102.11.11       100      0       i
*>l192.168.100.102/32 0.0.0.0            100 32768       i
*>i192.168.238.0/29   10.102.11.11       100      0       i
```

Example 3-1: *Loc-RIB (IPv4 Unicast) table on Leaf-102*

Step-2: Leaf-102 installs routes from the *Loc-RIB* into the *Adj-RIB-Out (Pre-Policy)* table. Note that there are dedicated *Pre-Policy* and *Post-Policy Adj-RIB-Out* tables towards each BGP peer. *Adj-RIB-Out (Pre-Policy)* table is a peer type-specific table (iBGP, eBGP, RR-Client, Confederation and Route-Server) and includes peer type-specific routes. IPv4 Unicast peering type between Leaf-102 and Spine-11 is internal BGP (iBGP), so routes learned from the other iBGP peer are not eligible for this *Adj-RIB-Out (Pre-Policy)* table. Leaf-102 attach the iBGP specific Path Attributes into each installed route excluding the Next-Hop Path Attribute which is not set in this phase.

Step-3: Leaf-102 sends routes from the *Adj-RIB-Out (Pre-Policy)* table through the Policy Engine into *the Adj-RIB-Out (Post-Policy)* table. In this example, Leaf-102 does not filter any routes or change the attached BGP Path Attributes, it just adds the Next-Hop Path Attribute introducing itself as a next-Hop. Leaf-102 sends BGP Update to Spine-11 over IPv4 Unicast peering.

```
Leaf-102# show ip bgp neighbors 10.102.11.11 advertised-routes       Network
Next Hop Metric LocPrf   Weight Path
*>l192.168.50.102/32  0.0.0.0          100     32768    i
*>l192.168.100.102/32 0.0.0.0          100     32768    i
```

Example 3-2: *Adj-RIB-Out (Post-Policy) table on Leaf-102*

Step-4: Spine-11 receives a BGP Update message from Leaf-102 and installs routes into peer-specific *Adj-RIB-In (Pre-Policy)* table without any modification.

Step-5: Spine-11 installs routes from the *Adj-RIB-In (Pre-Policy)* table through the Policy Engine into the *Adj-RIB-In (Post-Policy)* table without filtering prefixes or without changing Path Attributes carried in received BGP update.

Step-6: All Adj-RIB tables (In-Out/Pre-Post) are peer-specific and switch might have several BGP peers. Only the best route is installed into Loc-RIB based on the BGP Path selection procedure. Spine-11 installs the routes received from the Leaf-102 into *Loc-RIB* since there is no better path available via other BGP peer Leaf-101. In addition, it increases the BGP table version by 1.

```
Spine-11# sh ip bgp 192.168.100.102
BGP routing table information for VRF default, address family IPv4 Unicast
BGP routing table entry for 192.168.100.102/32, version 14
<snipped>
  Advertised path-id 1
  Path type: internal, path is valid, is best path, in rib
  AS-Path: NONE, path sourced internal to AS
    10.102.11.102 (metric 0) from 10.102.11.102 (192.168.0.102)
      Origin IGP, MED not set, localpref 100, weight 0

  Path-id 1 advertised to peers:
    10.101.11.101
```

Example 3-3: *BGP Loc-RIB in Spine-11.*

Step-7: Because there is no better route source than iBGP and routes are valid (reachable Next-Hop), Spine-11 installs routes from *Loc-RIB* also into the *RIB*.

Step-8: Leaf-101 and Leaf-102 are Route-Reflector clients of Spine-11. This way Spine-11 is able to forwards BGP updates messages received from iBGP peer Leaf-102 to another iBGP peer Leaf-101 and the other way around. As a first step, Spine-11 installs routes into Leaf-101 IPv4 unicast peering specific *Adj-RIB-Out (Pre-Policy)* table.

Step-9: Spine-11 installs routes from the *Adj-RIB-Out (Pre-Policy)* table into the Adj-RIB-Out (Post-Policy) table through the BGP Policy Engine. Because there is no operation defined in BGP Policy Engine, routes are installed without modification.

```
Spine-11# sh ip bgp neigh 10.101.11.101 advertised-routes

Peer 10.101.11.101 routes for address family IPv4 Unicast:
BGP table version is 10, Local Router ID is 192.168.0.11
<snipped>

   Network          Next Hop        Metric   LocPrf  Weight Path
*>i192.168.50.102/32  10.102.11.102          100      0      i
*>l192.168.100.11/32  0.0.0.0                100      32768  i
*>i192.168.100.102/32 10.102.11.102          100      0      i
*>l192.168.238.0/29   0.0.0.0                100      32768  i
```

Example 3-4: *Adj-RIB-Out (Post-Policy) table on Spine-11.*

When routes are sent to Leaf-101 by Spine-11, the command ***next-hop-self*** under the afi IPv4 Unicast section considering Leaf-101 neighbor parameters changes the Next-Hop Path Attribute to 10.101.11.11 (interface E1/1 of Spine-11). Example 3-5 illustrates partial BGP configuration on Spine-11.

Note, there is a dedicated section concerning next-hop-self after this section.

```
  neighbor 10.101.11.101
    remote-as 65000
    description ** BGP Underlay to Leaf-101 **
    address-family ipv4 unicast
      route-reflector-client
      next-hop-self
```

Example 3-5: *Spine-11 BGP configuration.*

Note! This solution breaks the recommendation on RFC 4456 (section 10) "*In addition, when a RR reflects a route, it SHOULD NOT modify the following path attributes: NEXT_HOP, AS_PATH, LOCAL_PREF, and MED. Their modification could potentially result in routing loops*".

In addition to Next-Hop modification, Spine-11 adds BGP Path Attributes Originator-Id (value: 192.168.0.102) and Cluster_List (value: 192.168.0.11) into BGP Update Message (Capture 3-1).

```
Ethernet II, Src: c0:8e:00:11:1e:11, Dst: 1e:af:01:01:1e:11
IPv4,Src: 10.101.11.11, Dst: 10.101.11.101
Transmission Control Protocol,
Src Port: 26601, Dst Port: 179, Seq: 58, Ack: 39, Len: 69
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 69
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 46
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
```

```
        Path Attribute - ORIGINATOR_ID: 192.168.0.102
        Path Attribute - CLUSTER_LIST: 192.168.0.11
        Path Attribute - MP_REACH_NLRI
         Address family identifier: IPv4 (1)
         Subsequent Address family identifier: Unicast (1)
         Next hop network address
           Next Hop: 10.101.11.11
         Network layer reachability information
           192.168.100.102/32
```

Capture 3-1: *BGP Update sent to Leaf-101 by Spine-11.*

Step-10-13: Leaf-101 receives the BGP Update from Spine-11. The import process follows the same principle that was described in Steps 4-6, Route is installed into *Adj-RIB-in (Pre-Policy)* table and from there through the Policy Engine into *Adj-RIB-in (Post-Policy).* After Best Path decision process route is installed via Loc-RIB into RIB.

Next-Hop-Self consideration

Just for a recap, Figure 3-3 illustrates three different peerings in BGP EVPN VXLAN fabric. The BGP IPv4 Unicast peering between switches is used in Underlay Network for advertising Loopback addresses which in turn are used in Overlay Network. Loopback 100 is used for BGP L2VPN EVPN. Network Virtual Edge (NVE) Interfaces uses Loopback 50 address for NVE tunnel peering but also for the source IP address in the outer IP header for all VXLAN encapsulated packets. VXLAN encapsulated Ethernet frames are sent over the VXLAN tunnel between NVE peers if the host MAC address is known. If the host MAC address is not known, it belongs to L2BUM traffic category (Broadcast, Unknown Unicast, and Multicast). In Multicast enabled core the destination IP address for L2BUM traffic is the Multicast Group address used in this specific VNI. In case of IP Only core, the Ingress-Replication is used for L2BUM and the copy of ingress L2BUM traffic is sent to every VTEP participating in the same L2VNI.

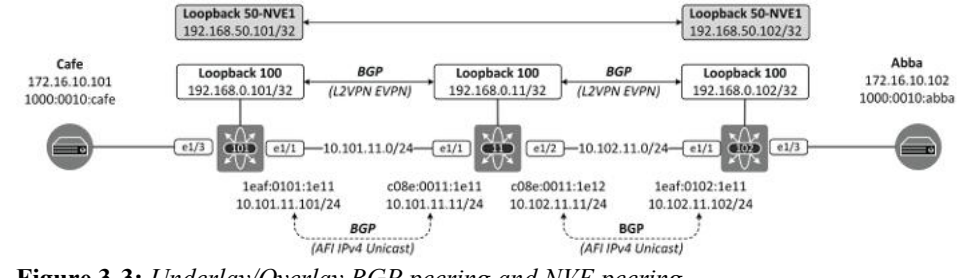


Figure 3-3: *Underlay/Overlay BGP peering and NVE peering.*

Case-1: Next-hop-self is changed by RR Spine-11.

In this case, Spine-11 changes the next-hop address when it forwards the BGP Update message received from Leaf-102 to Leaf-101. Example 3-6 illustrates the Loc-BGP table of Leaf-101. It shows that the Leaf-102 Loopback address 192.168.50.102 is reachable via 10.101.11.11 (Spine-11).

```
Leaf-101# sh ip bgp | i Net|192.168.50.102
   Network            Next Hop       Metric LocPrf Weight Path
*>i192.168.50.102/32  10.101.11.11          100          0 i
```

Example 3-6: *Leaf-101 Loc-BGP table.*

Example 3-7 shows that the Next-Hop address is also installed into the RIB.

```
Leaf-101# sh ip route 10.101.11.11 | b 10.1
10.101.11.11/32, ubest/mbest: 1/0, attached
    *via 10.101.11.11, Eth1/1, [250/0], 00:35:30, am
```

Example 3-7: *Leaf-101 RIB.*

Example 3-8 shows that the state of the connection between NVE1 interfaces of Leaf-101 and Leaf-102 is Up.

```
Leaf-101# sh nve peers
Interface Peer-IP     State LearnType Uptime   Router-Mac
---- ---------------  ----- ----      -------- --------------
nve1 192.168.50.102   Up    CP        00:28:39 5e00.0002.0007
```

Example 3-8: *Leaf-101Loc-BGP table.*

Example 3-9 shows that the host Cafe connected to Leaf-101 is able to ping host Abba which is connected to Leaf-102.

```
Cafe#ping 172.16.10.102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.102, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 19/28/36 ms
```

Example 3-9: *Leaf-101Loc-BGP table.*

Case-2: RR Spine-11 does not change Next-hop-self.

Now the Spine-11 does not change the next-hop when it forwards BGP Update received from Leaf-102 to Leaf-101. Example 3-10 shows that Leaf receives the route but since the next hop is not available, it is not a valid route (no * in front of the entry).

```
Leaf-101# sh ip bgp | i Net|192.168.50.102
   Network             Next Hop      Metric LocPrf  Weight Path
  i192.168.50.102/32  10.102.11.102         100          0 i
```

Example 3-10: *Leaf-101Loc-BGP table (no next-hop-self on Spine-11).*

Example 3-11 shows that there is no routing information concerning the IP address 192.168.50.102 and its Next-Hop address 10.102.11.102 (Interface E1/1 on Leaf-102) RIB of Leaf-101.

```
Leaf-101# sh ip route 192.168.50.102
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
```

```
'%<string>' in via output denotes VRF <string>

Route not found

Leaf-101# sh ip route 10.102.11.102
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

Route not found
Leaf-101#
```

Example 3-11: *RIB of Leaf-101.*

Also, the state of the NVE peering is now Down as can be seen from the example 3-12.

```
Leaf-101# sh nve peers
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- --------------   ----- --------- -------- -----------
nve1      192.168.50.102    Down  CP        0.000000 n/a
```

Example 3-12: *RIB of Leaf-101.*

However. ping still works between the Café and Abba (Example 3-13).

```
Cafe#ping 172.16.10.102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.102, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 26/36/54 ms
```

Example 3-13: *Ping from 172.16.10.101 (Café) to 172.16.10.102 (Abba)*

The reason for this can be found when comparing captures taken from the Interface E1/1 of Leaf-101 while pinging from Café to Abba. Capture 3-2 taken from the Interface E1/1 of Leaf-101 shows the encapsulated ICMP request packet sent by host Cafe (172.16.10.101) to host Abba (172.16.10.102) in solution, where Spine-11 changes the Next-Hop address. The destination MAC address in outer header belongs to interface E1/1 of Spine-11. The outer destination IP address is the IP address of Interface NVE1 of Leaf-102 (figure 3-4). When Spine-11 receives this packet, it routes the packet based on the Outer IP address.

```
Ethernet II,Src: 1e:af:01:01:1e:11 ,Dst: c0:8e:00:11:1e:11
IPv4, Src: 192.168.50.101, Dst: 192.168.50.102
User Datagram Protocol, Src Port: 57251, Dst Port: 4789
Virtual eXtensible Local Area Network
Ethernet II, Src: 10:00:00:10:ca:fe, Dst: 10:00:00:10:ab:ba
Internet Protocol Version 4,
Src: 172.16.10.101, Dst: 172.16.10.102
Internet Control Message Protocol
```

Capture 3-2: *Capture when Next-Hop-Self is used in Spine-11*

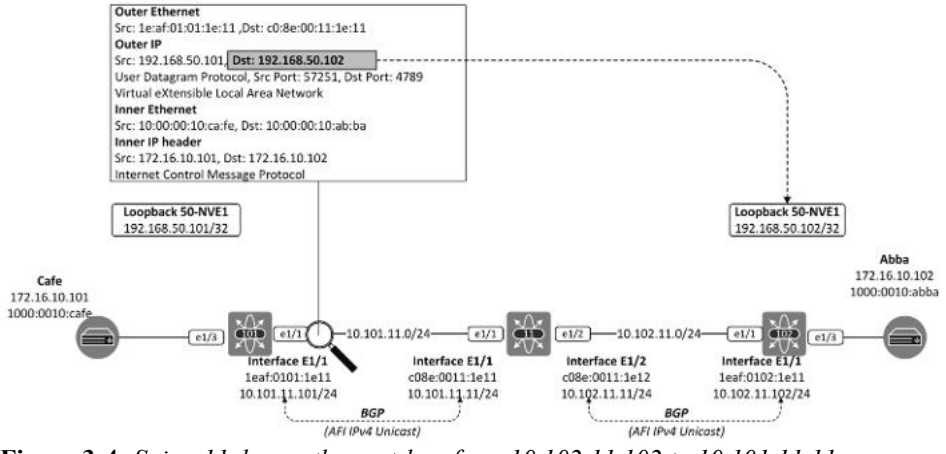


Figure 3-4: *Spine-11change the next-hop from 10.102.11.102 to 10.101.11.11.*

Capture 3-3 taken from the Interface E1/1 of Leaf-101 shows the encapsulated ICMP request packet sent by host Cafe (172.16.10.101) to host Abba (172.16.10.102) when the Spine-11 does NOT change the Next-Hop address. The previous example 3-12 shows that when next-hop is not reachable, the NVE peering state is down. Note that the BGP EVPN peering between Leaf-101 and Spine-11 as well as Leaf-102 and Spine-11 is still up and Leaf-101 receives the BGP Update originated by Leaf-102 but now with an invalid Next-Hop address. In this case, traffic falls to Unknown Unicast category and it is forwarded towards Multicast RP (Spine-11) of Mcast group 238.0.0.10

```
Ethernet II,
Src: 1e:af:01:01:1e:11 , Dst: IPv4mcast_0a (01:00:5e:00:00:0a)
Internet Protocol Version 4, 01:00:5e:00:00:0a
Src: 192.168.50.101, Dst: 238.0.0.10
User Datagram Protocol, Src Port: 57528, Dst Port: 4789
Virtual eXtensible Local Area Network
Ethernet II,
Src: 10:00:00:10:ca:fe, Dst: 10:00:00:10:ab:ba
Internet Protocol Version 4,
Src: 172.16.10.101, Dst: 172.16.10.102
Internet Control Message Protocol
```

Capture 3-3: *Capture when Next-Hop-Self is not used in Spine-11*

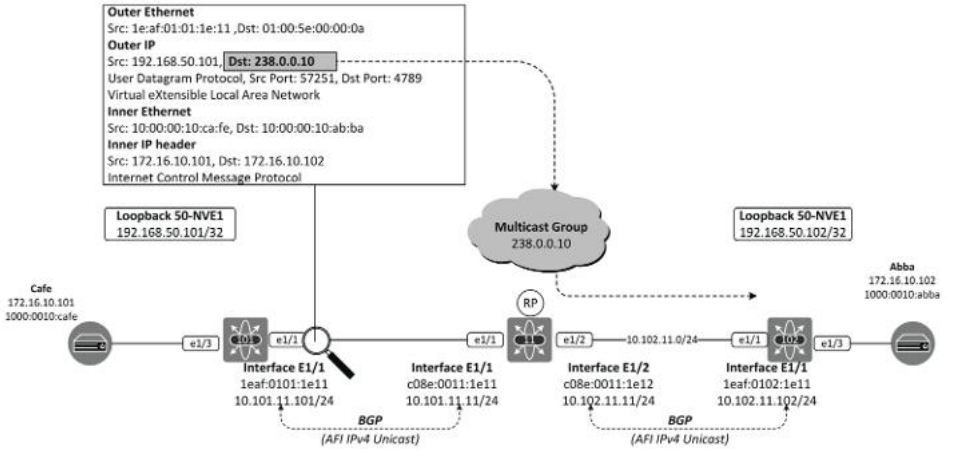


Figure 3-5: *Spine-11 does not change the next-hop address10.102.11.102.*

Spine-11 forwards packet based on its mroute table shown in example 3-14. Both Leaf -101 and Leaf-102 are joined to Mcast group 238.0.0.10 and interfaces towards them are listed in OIL (Outgoing Interface List). This way the ICMP request reaches Leaf-102 and this is how Data Plane still works, though it does not work as expected!

```
Spine-11# sh ip mroute
IP Multicast Routing Table for VRF "default"
<snipped>
(*, 238.0.0.10/32), bidir, uptime: 01:57:43, pim ip
  Incoming interface: loopback238, RPF nbr: 192.168.238.1
  Outgoing interface list: (count: 3)
    Ethernet1/2, uptime: 01:57:36, pim
    loopback238, uptime: 01:57:43, pim, (RPF)
    Ethernet1/1, uptime: 01:57:43, pim
```

Example 3-14: *Mroute table on Spine-11.*

If the hardware on Spine-11 switch does not support next-hop address modification in BGP Route-Reflector, the original Inter-Switch link used as Next-Hop can be advertised by BGP. This way the Next-Hop Addresses are reachable, the state of NVE peering remains Up and the data is sent as known Unicast. Example 3-13 shows that the route to 192.168.50.102 is now valid and the Next-Hop address 10.102.11.102 is reachable when the Inter-Switch links are also advertised.

```
Leaf-101# sh ip bgp | i Net|192.168.50.102
*>i192.168.50.102/32  10.102.11.102          100          0 i
Leaf-101# sh ip route 10.102.11.102 sec 10.102.11.0/24
<snipped>
10.102.11.0/24, ubest/mbest: 1/0
    *via 10.101.11.11, [200/0], 00:02:05, bgp-65000, internal,
```

Example 3-15: *Loc-BGP table and RIB on Leaf-101.*

References

[RFC 4456] T. Bates et al., “BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)”, RFC 4456, April 2006.

Chapter 4: Underlay Network: Two-AS eBGP

This chapter explains the Two-AS eBGP solution in VXLAN Fabric, where Leaf switches share the same AS while Spine switches use their own AS. This chapter also discusses how the default operating model used in eBGP peering has to be modified in order to achieve a routing solution required by VXLAN Fabric. These modifications are mainly related to BGP loop prevention model and BGP next-hop path-attribute processing.

Figure 4-1 illustrates the topology used in this chapter. Leaf-101 and Leaf-102 both belong to BGP AS 65000, while Spine-11 belongs to BGP AS 65099. Loopback interfaces L100 are used for Overlay Network BGP peering and Loopback 50 are used for NVE peering. Underlay Network eBGP peering is done between physical interfaces. Host Cafe and Abba are both in the same network 172.16.10.0/24 which VLAN 10 is mapped into L2VNI 10000.

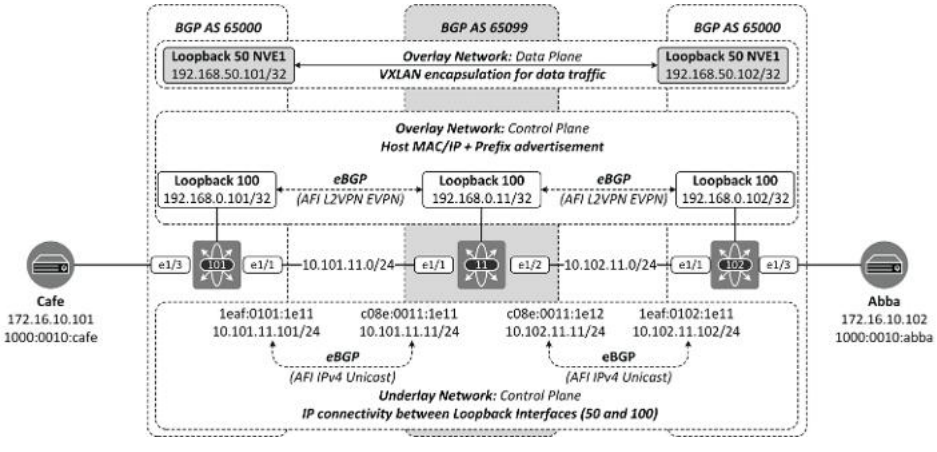


Figure 4-1: *Example topology and addressing.*

Underlay Network Control Plane eBGP

Figure 4-2 illustrates the Underlay Network addressing scheme and the peering model. BGP IPv4 peering is configured between the physical interfaces. Examples 1-1 through the 1-3 shows the basic BGP IPv4 peering configurations of switches. Both Leaf-101 and Leaf-102 advertise the IP addresses of Loopback 50 and Loopback 100 to Spine-11 while Spine-11 only advertises the IP address of Loopback 50 to leaf switches.

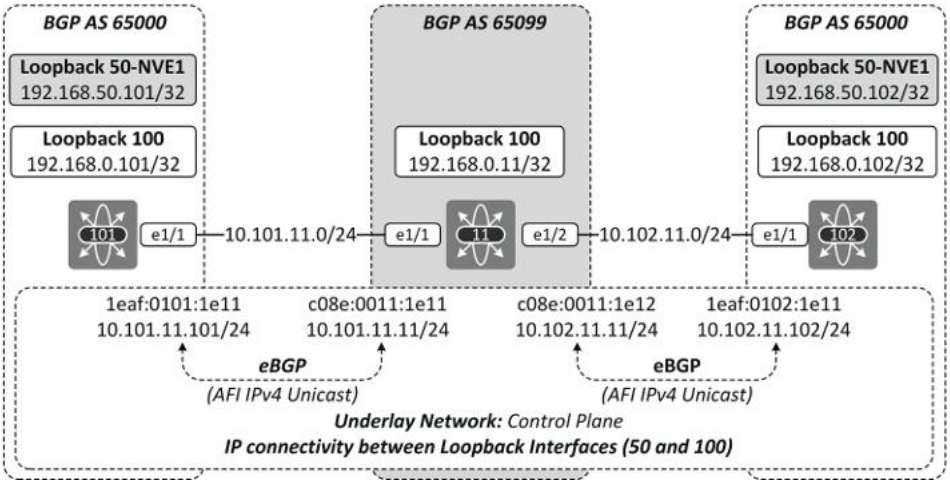


Figure 4-2: *VXLAN Fabric Underlay Network eBGP IPv4 peering.*

Examples 4-1 shows the BGP configuration of Leaf-101. It has an eBGP peering relationship with only Spine-11. IP addresses of interface Loopback 50 and Loopback 100 are advertised into BGP.

```
router bgp 65000
  router-id 192.168.0.101
  address-family ipv4 unicast
    network 192.168.50.101/32
    network 192.168.100.101/32
  neighbor 10.101.11.11
    remote-as 65099
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
```

Example 4-1: *Leaf-101 basic BGP configuration.*

Examples 4-2 shows the BGP configuration of Leaf-102.

```
router bgp 65000
  router-id 192.168.0.102
  address-family ipv4 unicast
    network 192.168.50.102/32
    network 192.168.100.102/32
  neighbor 10.102.11.11
    remote-as 65099
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
```

Example 4-2: *Leaf-102 basic BGP configuration.*

Example 4-3 shows the BGP configuration of Spine-11. It also advertises the IP address of interface Loopback 100 into BGP.

```
router bgp 65000
  router-id 192.168.0.11
  address-family ipv4 unicast
    network 192.168.100.101/32
  neighbor 10.101.11.101
    remote-as 65000
    description ** BGP Underlay to Leaf-101 **
    address-family ipv4 unicast
  neighbor 10.102.11.102
    remote-as 65000
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
```

Example 4-3: *Spine-11 basic BGP configuration.*

At this stage, the BGP peering between Leaf-101 and Spine-11 is up as can be seen from example 4-4.

```
Leaf-101# sh ip bgp summ | beg Neigh
Neighbor        V  AS MsgRcvd   MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.101.11.11    4 65099    2969    2954       11    0    0 02:27:42 2
```

Example 4-4: *Leaf-101 BGP peering.*

However, there are only local routes and routes originated by Spine-11 in Leaf-101 BGP table.

```
Leaf-101# sh ip bgp
<snipped>

   Network              Next Hop       Metric  LocPrf   Weight Path
*>l192.168.50.101/32    0.0.0.0                100       32768 i
*>e192.168.100.11/32    10.101.11.11                         0 65099 i
*>l192.168.100.101/32   0.0.0.0                100       32768 i
*>e192.168.238.0/29     10.101.11.11                         0 65099 i
```

Example 4-5: *Leaf-101 BGP routes.*

There are two reasons why BGP Updates originated by Leaf-102 does not end up in the BGP table of Leaf-101. First, Spine-11 does not forward BGP updates received from Leaf-102 to Leaf-101 at this moment. Example 4-6 shows that only self-originated routes are advertised to Leaf-101 by Spine-11. This is because the AS-PATH Path Attribute carried in BGP Update message includes the AS 65000 that is also used in Leaf-101 specific IPv4 BGP peering AS configuration. This is the default loop-prevention mechanism of BGP.

```
Spine-11# sh ip bgp neighbors 10.101.11.101 advertised-routes
<snipped>
   Network            Next Hop            Metric     LocPrf     Weight Path
*>l192.168.100.11/32  0.0.0.0                            100      32768 i
*>l192.168.238.0/29   0.0.0.0                            100      32768 i
```

Example 4-6: *Routes advertised to Leaf-101 by Spine-11.*

Disabling the peer-AS verification process before sending BGP Update with command "***disable-peer-as-check***" (example 4-7) changes this default behavior.

```
router bgp 65099
  neighbor 10.101.11.101
    address-family ipv4 unicast
      disable-peer-as-check
```

Example 4-7: *Disabling peer-AS verification on Spine-11.*

As can be seen from the example 4-8, Spine-11 now forwards BGP Update received from Leaf-102 to Leaf-101.

```
Spine-11# sh ip bgp neighbors 10.101.11.101 advertised-routes
<snipped>
Network               Next Hop       Metric LocPrf    Weight Path
*>e192.168.50.102/32  10.102.11.102                        0 65000 i
*>l192.168.100.11/32  0.0.0.0               100         32768 i
*>e192.168.100.102/32 10.102.11.102                        0 65000 i
*>l192.168.238.0/29   0.0.0.0               100         32768 i
```

Example 4-8: *Routes advertised to Leaf-101 by Spine-11.*

The second reason why routes do not end up into Leaf-101 BGP table is that even though Leaf-101 receives routes, it rejects them. BGP process discards BGP Updates messages learned from an eBGP peer, which carries receiving device AS Area information in its AS-Path list. This is also a default BGP loop prevention mechanism. This default behavior can be bypassed with "***allowas-in***" command under a peer-specific configuration (example 4-9).

```
router bgp 65000
  neighbor 10.101.11.11
    address-family ipv4 unicast
      allowas-in 3
```

Example 4-9: *Allow-as in on Leaf-101.*

After this addition, Leaf-101 accepts and installs routes originated by Leaf-102 into BGP table (example 4-10).

```
Leaf-101# sh ip bgp
<snipped>
   Network            Next Hop    Metric   LocPrf   Weight Path
*>l192.168.50.101/32  0.0.0.0                100      32768 i
*>e192.168.50.102/32  10.101.11.11                        0 65099 65000 i
*>e192.168.100.11/32  10.101.11.11                        0 65099 i
*>l192.168.100.101/32 0.0.0.0                100      32768 i
*>e192.168.100.102/32 10.101.11.11                        0 65099 65000 i
*>e192.168.238.0/29   10.101.11.11                        0 65099 i
```

Example 4-10: *Allow-as in on Leaf-101.*

The IP connectivity between the Leaf switches can now be verified by pinging between the Loopback interfaces (example 4-11).

```
Leaf-101# ping 192.168.100.102 source 192.168.100.101 count 2
<snipped>
64 bytes from 192.168.100.102: icmp_seq=0 ttl=253 time=9.268 ms
64 bytes from 192.168.100.102: icmp_seq=1 ttl=253 time=6.586 ms

<snipped>

Leaf-101# ping 192.168.50.102 source 192.168.50.101 count 2
<snipped>
64 bytes from 192.168.50.102: icmp_seq=0 ttl=253 time=27.166 ms
64 bytes from 192.168.50.102: icmp_seq=1 ttl=253 time=17.275 ms
```

Example 4-11: *ping from Leaf-101 to Leaf-102.*

Overlay Network Control Plane: eBGP

Figure 4-3 illustrates the Overlay Network addressing scheme and peering topology. BGP L2VPN EVPN peering is configured between Loopback 100 interfaces. Examples 4-12 and 4-13 shows the basic BGP L2VPN EVPN afi peering configurations of switches.

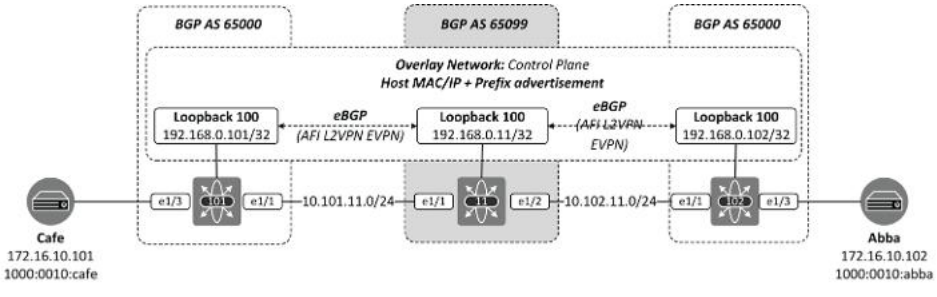


Figure 4-3: *VXLAN Fabric Overlay Network eBGP L2VPN EVPN peering.*

Both Leaf-switches can use the same configuration template if BGP L2VPN EVPN peering is configured between Loopback Interfaces. BGP sets TTL for BGP OPEN message to one by default. When peering between logical interfaces instead of the directly connected physical interface, the default TTL value one has to be manually increased by one with "***ebgp-multihop 2"*** command. In addition, peering between the logical loopback interfaces requires the update-source IP address modification since the IP address of the outgoing physical interface is used as a source IP for BGP messages sent to the external peer by default. This is achieved by using "***update-source loopback 100"*** command under the peer-specific configuration section. In addition, the same BGP loop-prevention mechanism that rejects routes with own AS-number applies also in Overlay Network, and "***allowas-in"*** is needed.

```
router bgp 65000
  neighbor 192.168.100.11
    remote-as 65099
    description ** BGP Overlay to Spine-11 **
    update-source loopback100
    ebgp-multihop 2
    address-family l2vpn evpn
      allowas-in 3
      send-community
      send-community extended
```

Example 4-12: *Basic BGP L2VPN EVPN configuration on Leaf-101and Leaf-102.*

Example 4-13 illustrates the Spine-11 BGP L2VPN EVPN peering configuration with Leaf-101. The "***disable-peer-as-check***" command is needed in Overlay BGP L2VPN EVPN peering just like it was needed in an Underlay BGP IPv4 peering.

```
router bgp 65099
  neighbor 192.168.100.101
    remote-as 65099
    description ** BGP Overlay to Leaf-101 **
    update-source loopback100
    ebgp-multihop 2
    address-family l2vpn evpn
      disable-peer-as-check
      send-community
      send-community extended
```

Example 4-13: *Basic BGP L2VPN EVPN peering configuration on Leaf-102.*

Now the BGP L2VPN EVPN peering is up, though Spine-11 has not installed any routes from neither Leaf-101 nor Leaf-102 into its BGP table.

```
Spine-11# sh bgp l2vpn evpn summary
BGP summary information for VRF default, address family L2VPN EVPN
BGP router identifier 192.168.0.11, local AS number 65099
BGP table version is 4, L2VPN EVPN config peers 2, capable peers 2
0 network entries and 0 paths using 0 bytes of memory
BGP attribute entries [0/0], BGP AS path entries [0/0]
BGP community entries [0/0], BGP clusterlist entries [0/0]

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.100.101 4 65000       6       6        4    0    0 00:00:13 0
192.168.100.102 4 65000       6       6        4    0    0 00:00:03 0
```

Example 4-14: *show bgp l2vpn evpn summry.*

L2VPN EVPN NLRIs are imported/exported based on Route-Target (RT) values. In Leaf-101and Leaf-102, there is an EVPN instance where the import/export policy has been defined (example 4-15).

```
Leaf-101# sh run bgp | sec evpn
<snipped>
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
    route-target both auto evpn
```

Example 4-15: *evpn vni 10000 Route-Target import/export policy on Leaf-101.*

There is no local EVPN instance configured on Spine-11, therefore it does not forward EVPN updates received from one eBGP peer to another eBGP peer. This rule applies to eBGP peering. In order to Spine-11 operate like a route-reflector, the command "***retain route-target***" is needed under global BGP L2VPN EVPN address-family (example 4-16). This way also the next-hop address carried in the update is retained.

```
Spine-11(config)# router bgp 65099
Spine-11(config-router)# address-family l2vpn evpn
Spine-11(config-router-af)# retain route-target all
```

Example 4-16: *retain route-target all command on Spine-11.*

Now, when the BGP L2VPN EVPN NLRIs are recent to Spine by Leaf-101…

```
Leaf-101# clear bgp l2vpn evpn 192.168.100.11 soft out
```

Example 4-17: *clear bgp l2vpn evpn on Leaf-101.*

… the MAC-only and MAC-IP NLRIs are received and installed into the BGP table of Spine-11. Note! Timestamps are removed (entries are updated from bottom to top).

```
Spine-11# sh bgp internal event-history events | i cafe

RIB: [L2VPN EVPN] Add/delete
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/112,
flags=0x200, evi_ctx invalid, in_rib: no

RIB: [L2VPN EVPN] Add/delete
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/144,
flags=0x200, evi_ctx invalid, in_rib: no

BRIB: [L2VPN EVPN]
(192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/112
(192.168.100.101)): returning from bgp_brib_add, reeval=0new_path: 1, change:
1, undelete: 0, history: 0, force: 0, (pflags=0x40002020) rnh_flag_change 0

BRIB: [L2VPN EVPN]
(192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/112
(192.168.100.101)): bgp_brib_add: handling nexthop, path->flags2: 0x80000

BRIB: [L2VPN EVPN] Created new path to
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/112 via
192.168.0.101 (pflags=0x40000000, pflags2=0x0)

BRIB: [L2VPN EVPN] Installing prefix
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/112
```

```
(192.168.100.101) via 192.168.50.101 label 10000 (0x0/0x0) into BRIB with
extcomm Extcommunity: RT:65000:10000 ENCAP:8

BRIB: [L2VPN EVPN]
(192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/144
(192.168.100.101)): returning from bgp_brib_add, reeval=0new_path: 1, change:
1, undelete: 0, history: 0, force: 0, (pflags=0x40002020) rnh_flag_c

BRIB: [L2VPN EVPN]
(192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/144
(192.168.100.101)): bgp_brib_add: handling nexthop, path->flags2: 0x80000

BRIB: [L2VPN EVPN] Created new path to
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/144
via 192.168.0.101 (pflags=0x40000000, pflags2=0x0)

BRIB: [L2VPN EVPN] Installing prefix
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/144
(192.168.100.101) via 192.168.50.101 label 10000 (0x0/0x0) into BRIB with
extcomm Extcommunity: RT:65000:10000 RT:65000:10077 ENC
```

Example 4-18: "*sh bgp internal event-history events | i café" on Spine-11.*

This can also be verified by checking the BGP table.

```
Spine-11# sh bgp l2vpn evpn 1000.0010.cafe
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.0.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216, version 93
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
  AS-Path: 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.101 (192.168.0.101)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 advertised to peers:
    192.168.100.102
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 90
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
  AS-Path: 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.101 (192.168.0.101)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
```

Example 4-19: *sh bgp l2vpn evpn 1000.0010.cafe on Spine-11.*

Example 4-20 shows that Leaf-102 has received BGP Update from Spine-11. Closer examination BGP table shows that the next-hop is Spine-11 though it should be Leaf-101.

```
Leaf-102# sh bgp l2vpn evpn 1000.0010.cafe
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.0.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216, version 19
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported to 1 destination(s)
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.100.11 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 not advertised to any peer

BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 4
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported to 3 destination(s)
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.100.11 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.0.102:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216, version 20
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path, in rib
             Imported from
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.100.11 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 not advertised to any peer
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 5
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
```

```
  Path type: external, path is valid, is best path, in rib
             Imported from
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.100.11 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.0.102:3    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 6
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported from
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.100.11 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007

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

Example 4-20: *show bgp l2vpn evpn 1000.0010.cafe on Leaf-102.*

In addition, the "***show nve peer detail***" command shows that the NVE peering is between Leaf-102 and Spine-11 while it should be between Leaf-102 and Leaf-101 (192.168.50.101). The reason for this is that Spine-11 changes the next-hop to its own IP address when it forwards BGP Update originated by Leaf-101 to Leaf-102 and the NVE peer information is taken from the next-hop field of L2VPN EVPN BGP Update.

```
Leaf-102# sh nve peers detail
Details of nve Peers:
----------------------------------------
Peer-Ip: 192.168.100.11
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 00:31:36
    Router-Mac          : 5e00.0000.0007
    Peer First VNI      : 10000
    Time since Create   : 00:31:36
    Configured VNIs     : 10000,10077
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10000,10077
    vni assignment mode : SYMMETRIC
    Peer Location       : N/A
```

Example 4-21: *show nve peers detail on Leaf-102.*

This means that there is no L2/L3 connectivity between host Café and host Abba as can be seen from example 4-22. Figure 4-4 illustrates the ICMP process.

```
Cafe#ping 172.16.10.102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.102, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Example 4-22: *ping from host Café to host Abba.*

Capture 4-1 is taken from the link between Leaf-101 and Spine-11 while the host Café tries to ping host Abba. First, since the hosts are in same subnet 172.16.10.0/24, host Cafe has to resolve the MAC address of host Abba. It sends an ARP request (L2 broadcast).

```
Ethernet II, Src: 10:00:00:10:ca:fe, Dst: ff:ff:ff:ff:ff:ff
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 10:00:00:10:ca:fe
    Sender IP address: 172.16.10.101
    Target MAC address: 00:00:00:00:00:00
    Target IP address: 172.16.10.102
```

Capture 4-1: *ARP request from host Cafe.*

ARP suppression is implemented in VNI 10000 in both Leaf switches. Since Leaf-101 knows the MAC address of host Abba (learned via BGP), it replies to ARP request by sending an ARP Reply as a unicast straight to the host Cafe.

```
Ethernet II, Src: 10:00:00:10:ab:ba, Dst: 10:00:00:10:ca:fe
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 10:00:00:10:ab:ba
    Sender IP address: 172.16.10.102
    Target MAC address: 10:00:00:10:ca:fe
    Target IP address: 172.16.10.101
```

Capture 4-2: *ARP reply from Leaf-101.*

Now host Cafe has resolved the MAC/IP of host Abba and it sends an ICMP request towards host Abba. Leaf-101 receives the ICMP request and makes a routing decision based on L2 RIB, where the next-hop incorrectly points to Spine-11 (example 4-23).

```
Leaf-101# sh l2route mac all

<snipped>
Topology    Mac Address    Prod   Flags         Seq No     Next-Hops
----------- -------------- ------ ------------- ---------- ----------------
10          1000.0010.abba BGP    SplRcv        0          192.168.100.11
10          1000.0010.cafe Local  L,            0          Eth1/3
77          5e00.0002.0007 VXLAN  Rmac          0          192.168.100.11
```

Example 4-23: *ping from host Café to host Abba.*

Leaf-101 encapsulates the frame and sets the outer destination IP to 192.168.100.11 (Capture 4-3). When Spine-11 receives the packet, it does not have any idea what to do with it and it rejects the packet.

```
Ethernet II, Src: 1e:af:01:01:1e:11, Dst: c0:8e:00:11:1e:11
Internet Protocol Version 4, Src: 192.168.50.101, Dst: 192.168.100.11
User Datagram Protocol, Src Port: 54810, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: Private_10:ca:fe (10:00:00:10:ca:fe), Dst: Private_10:ab:ba
(10:00:00:10:ab:ba)
Internet Protocol Version 4, Src: 172.16.10.101, Dst: 172.16.10.102
Internet Control Message Protocol
```

Capture 4-3: *forwarded frame by Leaf-101.*

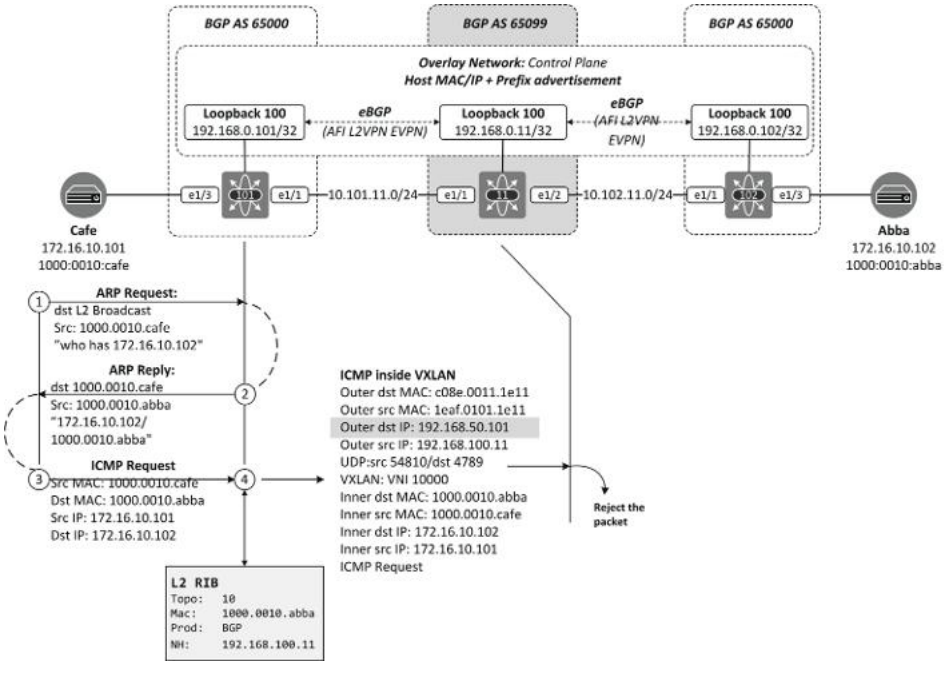


Figure 4-4: *ICMP process.*

In order to fix this, Spine-11 has to send L2VPN EVPN BGP Updates without modifying the Next-Hop Path Attribute. First, there is a route-map that prevents next-hop modification. This route map is then taken into action.

```
route-map DO-NOT-MODIFY-NH permit 10
  set ip next-hop unchanged
!
router bgp 65099
  router-id 192.168.0.11
  address-family ipv4 unicast
    network 192.168.100.11/32
    network 192.168.238.0/29
  address-family l2vpn evpn
    nexthop route-map DO-NOT-MODIFY-NH
    retain route-target all
  neighbor 10.101.11.101
    remote-as 65000
    description ** BGP Underlay to Leaf-101 **
    address-family ipv4 unicast
      disable-peer-as-check
  neighbor 10.102.11.102
    remote-as 65000
    description ** BGP Underlay to Leaf-102 **
    address-family ipv4 unicast
      disable-peer-as-check
  neighbor 192.168.100.101
    remote-as 65000
    description ** BGP Overlay to Leaf-101 **
    update-source loopback100
    ebgp-multihop 2
    address-family l2vpn evpn
      disable-peer-as-check
      send-community
      send-community extended
      route-map DO-NOT-MODIFY-NH out
  neighbor 192.168.100.102
    remote-as 65000
    description ** BGP Overlay to Leaf-102 **
    update-source loopback100
    ebgp-multihop 2
    address-family l2vpn evpn
      disable-peer-as-check
      send-community
      send-community extended
      route-map DO-NOT-MODIFY-NH out
```

Example 4-24: *Spine-11 bgp final configuration.*

After the change, Leaf-101 learns MAC/IP routes with the correct next-hop information.

```
Leaf-101# sh bgp l2vpn evpn 1000.0010.abba
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.0.101:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216, version 395
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path, in rib
             Imported from
192.168.0.102:32777:[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.102 (metric 0) from 192.168.100.11 (192.168.0.11)
```

```
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 not advertised to any peer
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.102]/272, version 369
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path, in rib
             Imported from
192.168.0.102:32777:[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.102]/272
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.102 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0002.0007

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.0.102:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216, version 394
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported to 1 destination(s)
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.102 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 not advertised to any peer
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.102]/272, version 367
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported to 3 destination(s)
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.102 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0002.0007

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.0.101:3    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.102]/272, version 370
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
```

```
  Path type: external, path is valid, is best path
             Imported from
192.168.0.102:32777:[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.102]/272
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.102 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0002.0007

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

Example 4-25: *BGP table on Leaf-101 concerning host Abba.*

The NVE peer information (example 4-26), as well as L2 routing information is L2 RIB (example 4-27) are now as expected.

```
Leaf-102# sh nve peer detail
Details of nve Peers:
--------------------------------------
Peer-Ip: 192.168.50.101
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 01:03:58
    Router-Mac          : 5e00.0000.0007
    Peer First VNI      : 10000
    Time since Create   : 01:03:58
    Configured VNIs     : 10000,10077
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10000,10077
    vni assignment mode : SYMMETRIC
    Peer Location       : N/A
```

Example 4-26: *NVE peer information on Leaf-102.*

```
Leaf-101# sh l2route mac all

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

Topology    Mac Address    Prod   Flags         Seq No     Next-Hops
----------- -------------- ------ ------------- ---------- ----------------
10          1000.0010.abba BGP    SplRcv        0          192.168.50.102
10          1000.0010.cafe Local  L,            0          Eth1/3
77          5e00.0002.0007 VXLAN  Rmac          0          192.168.50.102
```

Example 4-27: *L2 RIB on Leaf-101.*

Host Cafe is now able to ping host Abba.

```
Cafe#ping 172.16.10.102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.102, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 19/25/33 ms
```

Example 4-28: *ping from host Cafe to host Abba.*

As a final verification, capture 4-4 shows that ICMP packets from Cafe to Abba are now sent inside the VXLAN encapsulation with correct outer IP address 192.168.100.102.

```
Ethernet II, Src: 1e:af:01:01:1e:11, Dst: c0:8e:00:11:1e:11
IPv4, Src: 192.168.50.101, Dst: 192.168.50.102
User Datagram Protocol, Src Port: 59959, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: 10:00:00:10:ca:fe, Dst: 10:00:00:10:ab:ba
IPv4, Src: 172.16.10.101, Dst: 172.16.10.102
Internet Control Message Protocol
```

Capture 4-4: *VXLAN encapsulated ICMP packets.*

References:

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Chapter 5: eBGP as an Underlay Network Routing Protocol: Multi-AS eBGP

This chapter introduces the Multi-AS eBGP solution in VXLAN Fabric. In this solution, a single AS number is assigned to all spine switches while each leaf switches (or pair of vPC switches) have a unique BGP AS number. This solution neither requires "***allowas-in***" command in leaf switches nor "***disable-peer-check***" command in spine switches, which are required in Two-AS solution. The "***retain-route-target all***" command and BGP L2VPN EVPN address family peer-specific route-map with "***set ip next-hop-unchanged***" option are still needed on the spine switch. This chapter also explains the requirements and processes for L2 EVPN VNI specific route import policy when automated derivation of Route-Targets is used. The same IP/MAC address scheme is used in this chapter than what was used in the previous chapter "*VXLAN Underlay Routing - Part IV: Two-AS eBGP*" but the Leaf-102 now belongs to BGP AS 65001.

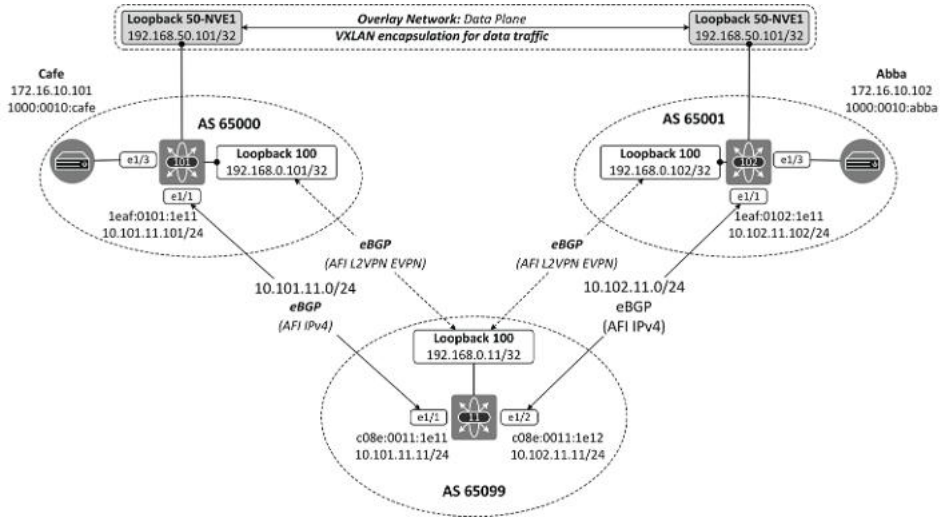


Figure 5-1: *The MAC/IP addressing scheme and eBGP peering model.*

Underlay Network Control Plane: IPv4 eBGP peering

Spine-11 belongs to BGP AS 65099 and it has IPv4 BGP peering with AS external neighbors Leaf-101 on AS 65000 and Leaf-102 on AS 65001. Both Leaf switches advertise the NLRIs about their Loopback 100 (used for overlay BGP peering) and Loopback 50 (used for NVE interfaces) to Spine-11. Spine-11 advertised the NLRI information about its interface Loopback 100. In addition, Spine-11 forwards the NLRI information received from Leaf-101 to Leaf-102 and another way around. The basic BGP configuration is shown in examples 5-1 to 5-3.

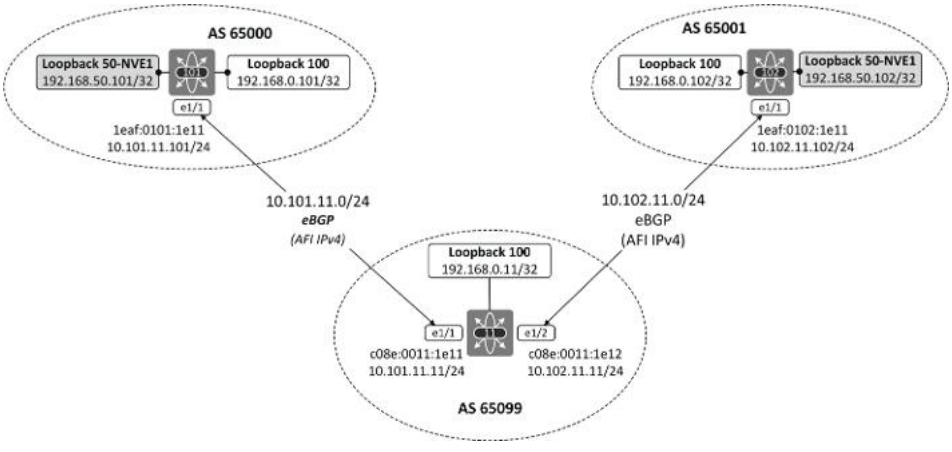


Figure 5-2: *VXLAN Fabric Underlay Network eBGP IPv4 peering.*

```
router bgp 65000
  router-id 192.168.0.101
  address-family ipv4 unicast
    network 192.168.50.101/32
    network 192.168.100.101/32
  neighbor 10.101.11.11
    remote-as 65099
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
```

Example 5-1: *Leaf-101 basic IPv4 BGP peering configuration.*

```
router bgp 65001
  router-id 192.168.0.102
  address-family ipv4 unicast
    network 192.168.50.102/32
    network 192.168.100.102/32
  neighbor 10.102.11.11
    remote-as 65099
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
```

Example 5-2: *Leaf-102 basic IPv4 BGP peering configuration.*

```
router bgp 65000
  router-id 192.168.0.11
  address-family ipv4 unicast
    network 192.168.100.101/32
  neighbor 10.101.11.101
    remote-as 65000
    description ** BGP Underlay to Leaf-101 **
    address-family ipv4 unicast
  neighbor 10.102.11.102
    remote-as 65000
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
```

Example 5-3: *Spine-11 basic IPv4 BGP peering configuration.*

Example 5-4 shows that Spine-11 has received two routes for both IPv4 BGP peers.

```
Spine-11# sh ip bgp summary
BGP summary information for VRF default, address family IPv4 Unicast
BGP router identifier 192.168.0.11, local AS number 65099
BGP table version is 96, IPv4 Unicast config peers 2, capable peers 2
6 network entries and 6 paths using 1392 bytes of memory
BGP attribute entries [3/480], BGP AS path entries [2/12]
BGP community entries [0/0], BGP clusterlist entries [0/0]

Neighbor             V       AS  MsgRcvd  MsgSent     TblVer    InQ  OutQ  Up/Down
State/PfxRcd
10.101.11.101    4 65000     274     264       96     0     0 00:00:19 2
10.102.11.102    4 65001     286     273       96     0     0 00:00:27 2
```

Example 5-4: *show ip bgp summary on Spine-11.*

Example 5-5 shows that Leaf-101 has received and installed routes originated by Leaf-102 into the BGP table.

```
Leaf-101# sh ip bgp | i .102
*>e192.168.50.102/32  10.101.11.11                          0 65099 65001 i
*>e192.168.100.102/32 10.101.11.11                          0 65099 65001 i
```

Example 5-5: *show ip bgp on Spine-11.*

Example 5-6 shows that there is an IP connectivity between the Loopback IP addresses of Leaf-101 and Leaf-102.

```
Leaf-101# ping 192.168.100.102 source 192.168.100.101 count 2
<snipped>
64 bytes from 192.168.100.102: icmp_seq=0 ttl=253 time=7.896 ms
64 bytes from 192.168.100.102: icmp_seq=1 ttl=253 time=6.913 ms
<snipped>
Leaf-101# ping 192.168.50.102 source 192.168.50.101 count 2
<snipped>
64 bytes from 192.168.50.102: icmp_seq=0 ttl=253 time=6.922 ms
64 bytes from 192.168.50.102: icmp_seq=1 ttl=253 time=10.413 ms
<snipped>
```

Example 5-6: *IP connectivity verification from Leaf-101 to Leaf-102.*

Overlay Network Control Plane: L2VPN EVPN eBGP peering

While Underlay Network IPv4 BGP peering is used for IP connectivity between devices, the Overlay L2VPN EVPN BGP peering is used to advertise EVPN related NLRIs. This section explains how automated derivation of Route-Targets (RT) can be achieved even though switches in different AS area generates a different RT value.

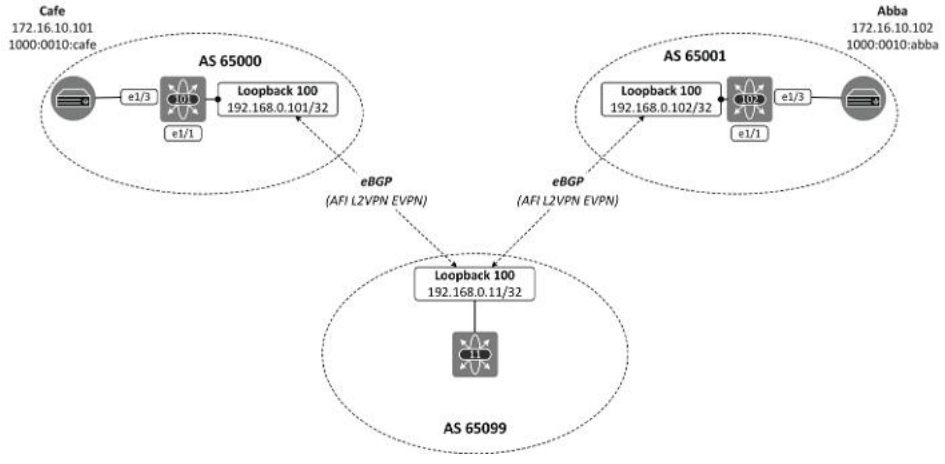


Figure 5-3: *VXLAN Fabric Overlay Network eBGP L2VPN EVPN peering.*

The basic EVPN L2VNI 10000 configuration on both leaf switches is illustrated in example 5-7.

```
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
```

Example 5-7: *IP connectivity verification from Leaf-101 to Leaf-102.*

The format of auto RT is "AS number:L2VNI". Therefore, Leaf-101 export routes with RT 65000:10000 and import routes with the same RT. Leaf-102 in turn export routes with RT 65001:10000 and import routes with the same RT. This means that neither leaf switch does import routes originated by the remote leaf. The solution is to use L2VPN EVPN BGP peer-specific command "***rewrite-evpn-rt-asn***". This command will change the AS number part from the RT to local AS on received BGP Updates. The next section explains how it works.

Example 5-8 shows that Spine-11 has received a BGP Update from Leaf-101 about NLRI of host Café (IP:172.16.10.101/MAC:1000.0010.cafe). Two RTs are included in BGP Update. The first one 65000:10000 is L2VNI specific RT and it used for intra-VN connection. The second one 65000:10077 is used inside specific tenant for Inter-VNI communication (L3VNI).

```
Spine-11# sh bgp l2vpn evpn 1000.0010.cafe
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.0.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216,
 version 293
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
  AS-Path: 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.101 (192.168.0.101)
```

```
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 advertised to peers:
    192.168.100.102
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 244
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
  AS-Path: 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.101 (192.168.0.101)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 advertised to peers:
    192.168.100.102
```

Example 5-8: *BGP table on Spine-11.*

Spine-11 also advertises routes to Leaf-102.

```
Spine-11# sh bgp l2vpn evpn neighbors 192.168.100.102 advertised-routes

Peer 192.168.100.102 routes for address family L2VPN EVPN:
BGP table version is 299, Local Router ID is 192.168.0.11
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
i
njected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.0.101:32777
*>e[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                      192.168.50.101                                 0 65000 i
*>e[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
                      192.168.50.101                                 0 65000 i

Route Distinguisher: 192.168.0.102:32777
```

Example 5-9: *Advertised NLRIs to Leaf-102 by Spine-11.*

However, the NLRI information is not installed from BGP Adj-RIB-In into Loc-RIB on Leaf-102. (received only)

```
Leaf-102# sh bgp l2vpn evpn 1000.0010.cafe
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.0.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216, version 0
Paths: (1 available, best #0)
```

```
Flags: no flags set

  Path type: external, path is valid, received only
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/248, version 0
Paths: (1 available, best #0)
Flags: no flags set

  Path type: external, path is valid, received only
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
```

Example 5-10: *show bgp l2vpn evpn 1000.0010.cafe on Leaf-102.*

The import policy has to match with Route-Target value Extended Community in BGP Update message in order to NLRI information can be installed from the peer-specific Adj-RIB-In into Loc-RIB. The RT of received BGP Update can be modified by using BGP L2VPN EVPN peer-specific command "***rewrite-evpn-rt-asn***". It changes the RT value of incoming BGP Update before installing it into Adj-RIB-In. Example 1-11 shows the configuration.

```
router bgp 65001
  router-id 192.168.0.102
  address-family ipv4 unicast
    network 192.168.50.102/32
    network 192.168.100.102/32
  address-family l2vpn evpn
  neighbor 10.102.11.11
    remote-as 65099
    description ** BGP Underlay to Spine-11 **
    address-family ipv4 unicast
  neighbor 192.168.100.11
    remote-as 65099
    description ** BGP Overlay to Spine-11 **
    update-source loopback100
    ebgp-multihop 2
    address-family l2vpn evpn
      send-community extended
      soft-reconfiguration inbound always
      rewrite-evpn-rt-asn
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
```

Example 5-11: *BGP configuration on Leaf-102.*

Adding command towards Spine-11 on Leaf-101 and Leaf-102 does not yet full fill the import policy requirements. Before RT modification, the BGP process compares the Route-Target AS number part to sending BGP peer AS number defined in a peer specific configuration. If the AS part in RT is the same as neighbor AS, the BGP process modifies the RT and import the NLRI into Loc-RIB if not the update is ignored. Therefore, also Spine-11 has to manipulate the RT value for BGP Updates that are received from Leaf-101. Figure 5-4 illustrates the situation where Spine-11 forwards BGP Update exported by Leaf-101 without RT manipulation. Leaf-102 does not install NLRI into Loc-RIB because the configured AS number for BGP L2VPN EVPN peer Spine-11 is different compared to Route-Target AS part of BGP Update received from Spine-11.

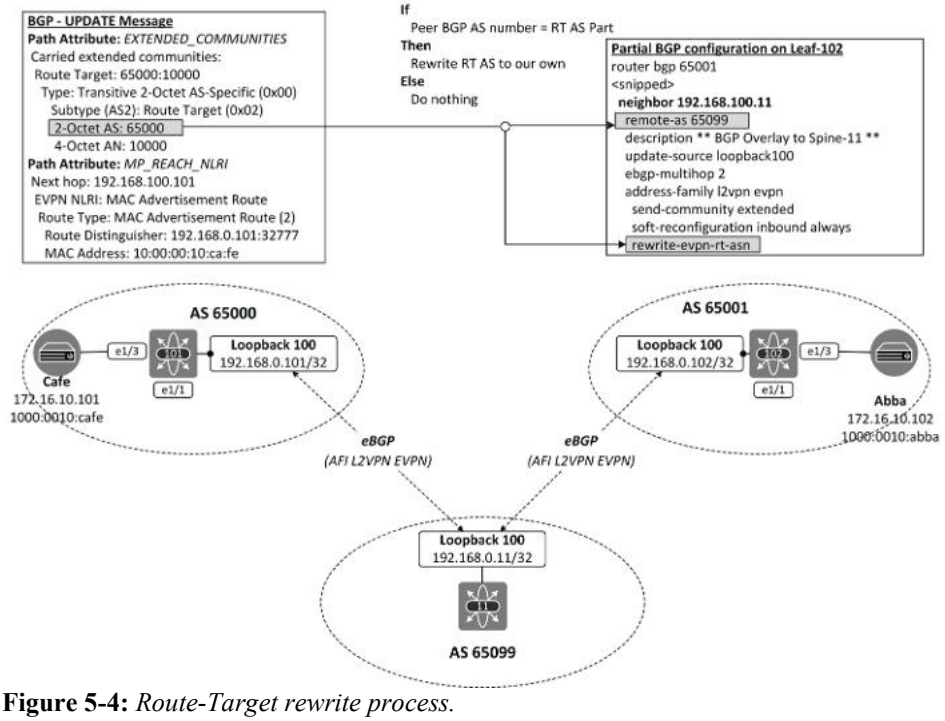


Figure 5-4: *Route-Target rewrite process.*

When the command "***rewrite-evpn-rt-asn***" is also added into Spine-11, the configuration towards Leaf-101 and Leaf-102, leaf switches are able to change the RT value carried in received BGP Updates and install the NLRIs into the BGP Loc-RIB table. Figure 5-5 illustrates the overall process.

Step-1:
Leaf-101 sends BGP Update with RT 65000:10000 to Spine-11.

Step-2:
Spine-11 receives the BGP Update. It compares the BGP AS part from RT to configured BGP AS number towards Leaf-101.

Step-3:
Because both values are equal, Spine-11 rewrites the original AS value with its own AS.

Step-4:
Spine-11 imports the NLRI into the BGP Loc-RIB table where it is sent through the Adj-RIB-Out to Leaf-102 with RT 65099:10000.

Step 5-7:
Leaf-102 does the same verification process that what Spine-11 did in phases 1-4 and import the NLRI into Loc-RIB.

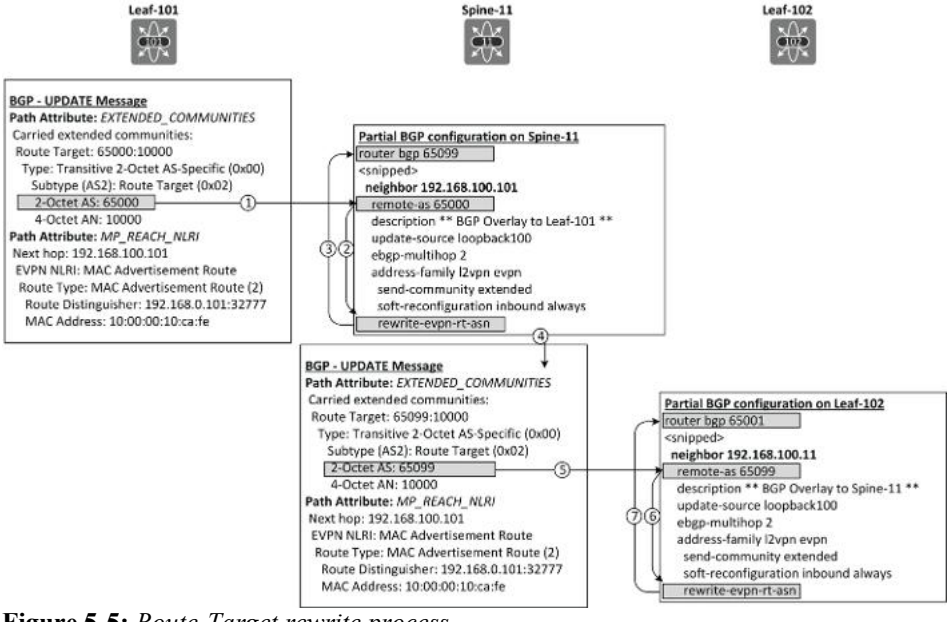


Figure 5-5: *Route-Target rewrite process.*

Example 5-12 shows that now the NLRI originated by Leaf-101 is installed from the BGP Adj-RIB-In into Loc-RIB.

```
Leaf-102# sh bgp l2vpn evpn 1000.0010.cafe
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.0.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216, version 71
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, received and used, is best path
             Imported to 1 destination(s)
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65001:10000 ENCAP:8

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

```
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 70
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, received and used, is best path
             Imported to 3 destination(s)
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65001:10000 RT:65001:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.0.102:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216, version 74
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path, in rib
             Imported from
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65001:10000 ENCAP:8

  Path-id 1 not advertised to any peer
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 72
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path, in rib
             Imported from
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
  AS-Path: 65099 65000 , path sourced external to AS
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65001:10000 RT:65001:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.0.102:4    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272, version 73
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: external, path is valid, is best path
             Imported from
192.168.0.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
  AS-Path: 65099 65000 , path sourced external to AS
```

```
    192.168.50.101 (metric 0) from 192.168.100.11 (192.168.0.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65001:10000 RT:65001:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 not advertised to any peer

Leaf-102#
```

Example 5-12: *BGP table on Leaf-102.*

Example 5-13 shows that host Cafe (172.16.10.101/1000.0010.cafe) connected to Leaf-101 is now able to ping host Abba (172.16.10.102/1000.0010.abba) connected to Leaf-102.

```
Cafe#ping 172.16.10.102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.102, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/25/34 ms
Cafe#
```

Example 5-13: *Ping from host Café to host Abba.*

References:

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Cisco Programmable Fabric with VXLAN BGP EVPN Configuration Guide
https://www.cisco.com/c/en/us/td/docs/switches/datacenter/pf/configuration/guide/b-pf-configuration/IP-Fabric-Underlay-Options.html

Chapter 6: Layer 2 Multi-Destination Traffic - Anycast-RP with PIM.

The L2VNI specific Layer 2 BUM traffic can be forwarded in two ways in VXLAN fabric. The first method relies on Underlay Network Multicast routing. The second option uses Ingress- Replication (IR) solution, where the copy of each ingress L2BUM frame is sent as a Unicast to every VTEP participating in the same L2VNI. IR can be used if Multicast Routing is not enabled on Underlay Network. This chapter explains the Multicast mode focusing Anycast-RP with PIM solution. For the Layer 3 Multicast traffic between hosts, there should be Tenant Routed Multicast (TRM) Overlay Multicast design.

Figure 6-1 illustrates the example network and IP addressing schemes used throughout this chapter. Spine-11 and Spine-12 shares the same Anycast-RP (Rendezvous Point) IP address and they belong to the same "Anycast-RP set" group that uses IP address of interface Loopback 238. Also, there is an additional Loopback interface, which must be unique in each spine (Loopback 511 and 512). These addresses are used as an Anycast-RP group member identifier. Both addresses, shared and unique, need to be reachable for all switches.

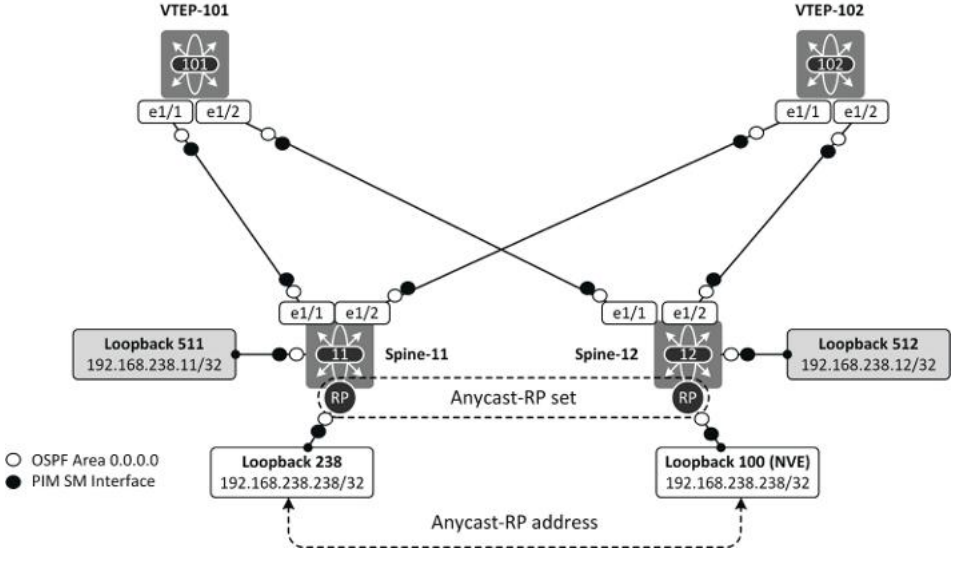


Figure 6-1: *Example topology with Anycast-RP - IP addresses.*

This section explains the configuration and theory of Anycast-RP Multicast routing solution in an Underlay Network.

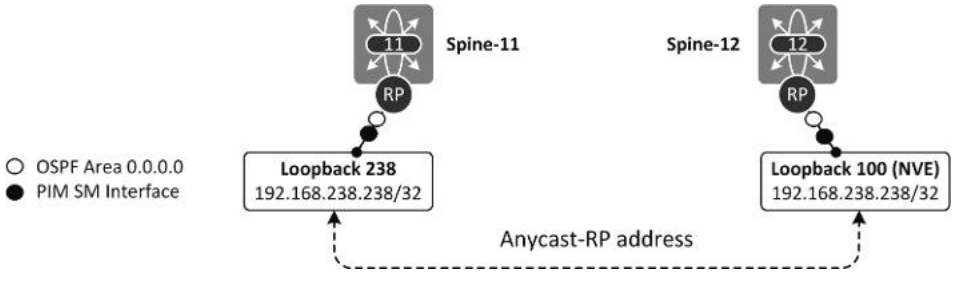


Figure 6-2: *Anycast-RP Cluster Members.*

Step 1: Configuring Anycast-RP cluster

Enable PIM feature in both Spine switches. Create interface Loopback 238 for Anycast-RP and assign the IP address 192.168.238.238/32 for both switches. In addition, enable PIM and OSPF in Interface. The IP address of Loopback Interface 238 has to be reachable for all switches.

```
feature pim
!
interface loopback238
  description ** Anycast-RP address **
  ip address 192.168.238.238/32
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
```

Example 6-1: *Anycast-RP Loopback Interface Configuration.*

Step 2: Assign unique Cluster Member IP and define members

Configure the unique IP addresses for each Anycast-RP cluster member and enable PIM-SM and OSPF on it. The unique address is used as a cluster member identifier. Define the other Anycast-RP cluster members. An example configuration is taken from Spine-11.

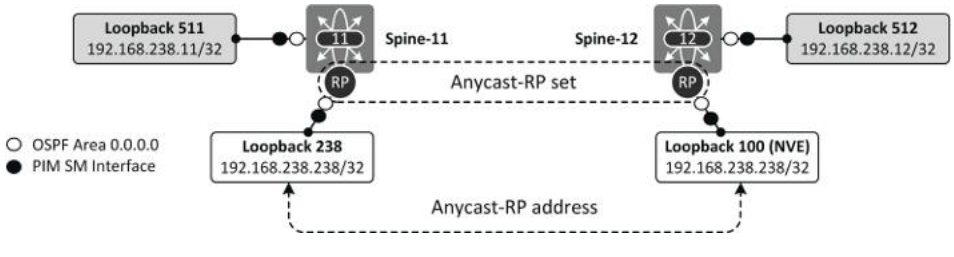


Figure 6-3: *Anycast-RP Cluster Members.*

```
interface loopback511
  description ** Unique Address for Anycast-RP **
  ip address 192.168.238.11/32
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
!
ip pim anycast-rp 192.168.238.238 192.168.238.11
ip pim anycast-rp 192.168.238.238 192.168.238.12
```

Example 6-2: *Anycast-RP Unique Loopback Interface Configuration on Spine-11.*

Step 3: Assign unique Cluster Member IP and define members

Configure the group-specific Rendezvous Point IP address to all switches.

```
ip pim rp-address 192.168.238.238 group-list 224.0.0.0/4
```

Example 6-3: *Defining the Anycast-RP IP address.*

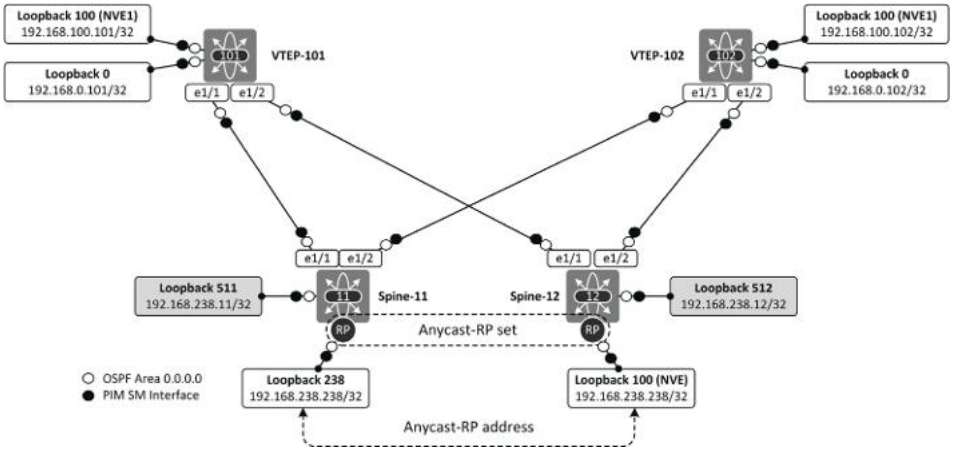


Figure 6-4: *Anycast-RP Cluster Members.*

Examples 6-4 and 6-5 verifies that both Spine switches are now a member of the same Anycast-RP Cluster.

```
Spine-11# sh ip pim rp vrf default
PIM RP Status Information for VRF "default"
<snipped>

Anycast-RP 192.168.238.238 members:
  192.168.238.11*  192.168.238.12

RP: 192.168.238.238*, (0),
 uptime: 02:28:30   priority: 255,
 RP-source: (local),
 group ranges:
 224.0.0.0/4
```

Example 6-4: *sh ip pim rp vrf default on Spine-11.*

```
Spine-12# sh ip pim rp vrf default
PIM RP Status Information for VRF "default"
<snipped>

Anycast-RP 192.168.238.238 members:
  192.168.238.11  192.168.238.12*

RP: 192.168.238.238*, (0),
 uptime: 02:09:53   priority: 255,
 RP-source: (local),
 group ranges:
 224.0.0.0/4
```

Example 6-5: *sh ip pim rp vrf default on Spine-12.*

The Anycast-RP configuration is now ready and the Underlay Network is capable of forwarding L2BUM.

The next section discusses the functionality. VLAN 10 is attached to VNI 10000 on both VTEP (Virtual Tunnel End Point). The Multicast Group 238.0.0.10 is attached to VNI 10000 under NVE (Network Virtualization Edge) Interface. This means that L2 BUM traffic is forwarded towards the group RP.

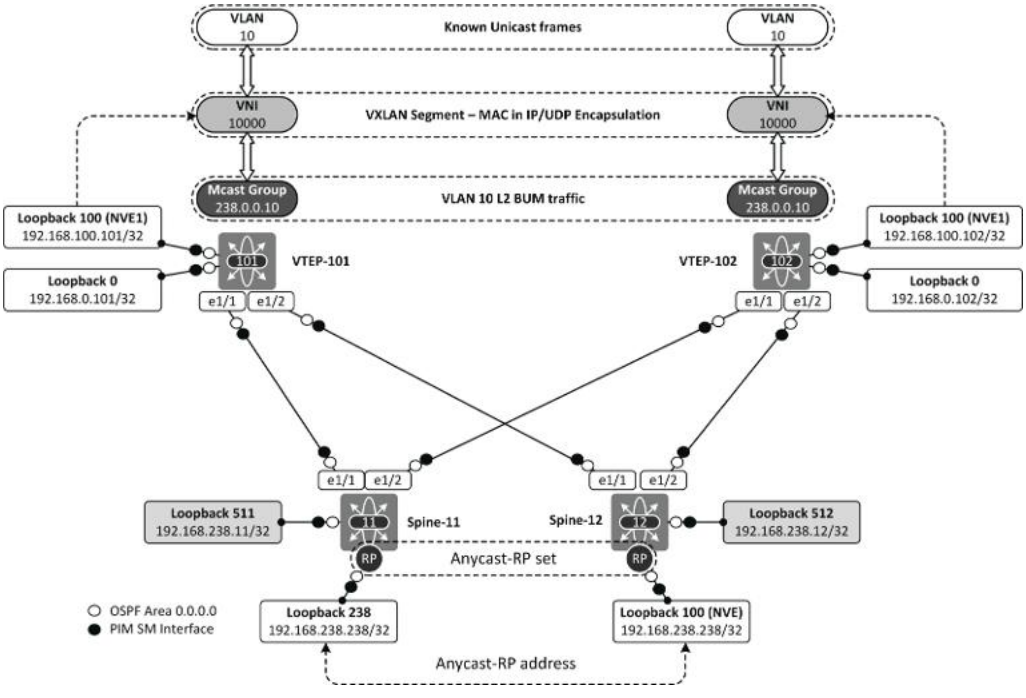


Figure 6-5: *VLAN-to-VNI-to-Multicast Group*

Configuring NVE interface

Create VLAN and assign it to VNI 10000. Add VNI 10000 under the interface NVE1 as a member VNI and use Multicast Group 238.0.0.10 for VNI specific L2 BUM traffic (ARP, DHCP, ND and so on).

```
vlan 10
  vn-segment 10000
!
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
!
interface nve1
  no shutdown
  source-interface loopback100
  member vni 10000
    mcast-group 238.0.0.10
```

Example 6-6: *sh ip pim rp vrf default on Spine-12.*

Anycast-PIM Control Plane Operation

The Control Plane operation of Anycast-RP with PIM solution is discussed next. The NVE1 is first brought down in both VTEP switches.

The Multicast Group 238.0.0.10 (MG) specific Incoming Interface List (IIL) and Outgoing Interface List (OIL) are empty on the Multicast RIB (MRIB) of both spine switches.

```
Spine-11# sh ip mroute
IP Multicast Routing Table for VRF "default"

(*, 232.0.0.0/8), uptime: 00:00:01, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)
```

Example 6-7: *sh ip pim rp vrf default on Spine-12.*

```
Spine-12# sh ip mroute
IP Multicast Routing Table for VRF "default"

(*, 232.0.0.0/8), uptime: 00:00:01, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)
```

Example 6-8: *sh ip pim rp vrf default on Spine-12.*

Phase 1: PIM Join

When the state of the interface NVE1 on VTEP1 changes from down to up, it triggers a PIM Join process. VTEP-101 uses PIM Join messages to inform its willingness to receive Multicast traffic sent to Multicast Group 238.0.0.10, which is the group where VTEP switches forwards L2BUM traffic (e.g. ARP requests) originated from VLAN 10 (mapped to VNI 10000). The PIM Join message is sent towards the Multicast group-specific Rendezvous Point (RP), which is statically defined in each switch. In Anycast-RP solution used in this chapter, both spines switches share the role of RP and in Unicast RIB there are two equal-cost path to RP address. However, the forwarding decision of PIM Join messages is based on Multicast Routing Information Base (MRIB) and by default, only one of the upstream links is selected to Reverse Path Forwarding (RPF) list. In this example, VTEP-101 has chosen and installed Spine-11 into RPF list with outgoing interface E1/1 as can be seen form example 6-9. This is why the PIM Join message is sent to only Spine-11. The source address is VTEP-101s Underlay Network IP address 192.168.0.101 (Unnumbered Loopback 0).

```
(*, 238.0.0.10/32), bidir, uptime: 00:25:52, nve ip pim
  Incoming interface: Ethernet1/1, RPF nbr: 192.168.0.11
  Outgoing interface list: (count: 2)
    Ethernet1/1, uptime: 00:07:51, pim, (RPF)
    nve1, uptime: 00:25:52, nve
```

Example 6-9: *M-RIB on VTEP-101.*

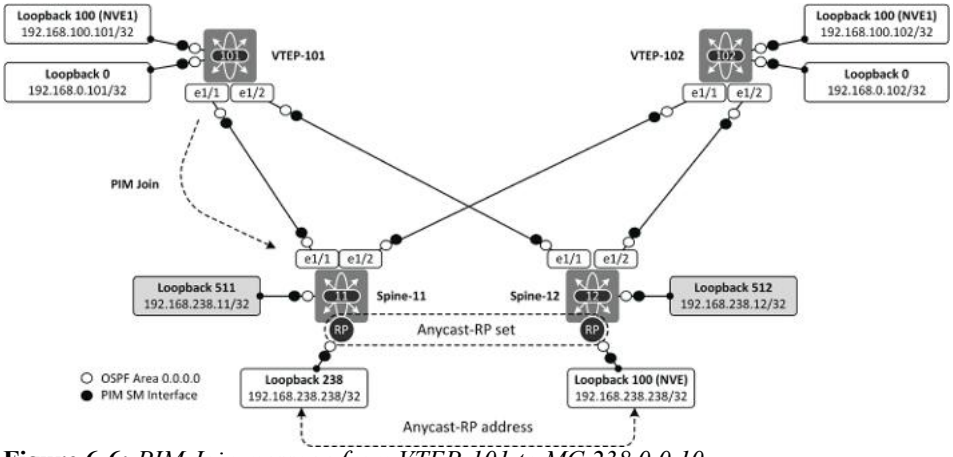


Figure 6-6: *PIM Join message from VTEP-101 to MC 238.0.0.10.*

The capture below illustrates the PIM Join message captured from interface E1/1 on VTEP-101.

```
Frame 9: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: IPv4mcast_0d
(01:00:5e:00:00:0d)
Internet Protocol Version 4, Src: 192.168.0.101, Dst: 224.0.0.13
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0011 = Type: Join/Prune (3)
    Reserved byte(s): 00
    Checksum: 0x4866 [correct]
    [Checksum Status: Good]
    PIM Options
        Upstream-neighbor: 192.168.0.11
        Reserved byte(s): 00
        Num Groups: 1
        Holdtime: 210
        Group 0: 238.0.0.10/32
            Num Joins: 1
                IP address: 192.168.238.238/32 (SWR)
            Num Prunes: 1
                IP address: 192.168.100.101/32 (SR)
```

Capture 6-1: *PIM Join message from VTEP-101 to MG 238.0.0.10.*

Phase 2: PIM Registration

In the L2BUM solution that relies on Multicast routing, all VTEP switches are also Multicast sources. VTEP-101 informs RP that in addition to role of receiver it is source for Multicast group 238.0.0.10. Leaf-101 sends a PIM Register message to MG specific Rendezvous Point. Unlike PIM join process where the PIM Join message is sent based on Multicast RIB, the PIM Register message is routed based on Unicast RIB. Since both spine switches introduce themself as Anycast-RP, VTEP-101 has two equal-cost paths to 192.168.238.238. ECMP hash on

VTEP-101 selects link towards Spine-12. The PIM Register message use encapsulation method where the Multicast traffic to specific group is encapsulated by outer IP header where the destination IP is set to RP address (capture 6-2). After registration process, all L2BUM traffic is sent without encapsulation using Multicast Group address as a destination IP.

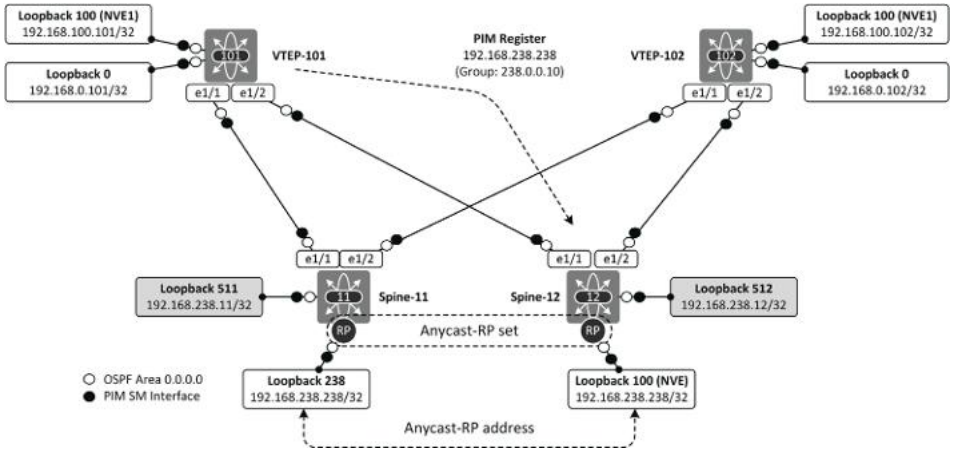


Figure 6-7: *PIM Register message from VTEP-101 to MG 238.0.0.10.*

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: 5e:00:00:03:00:07
(5e:00:00:03:00:07)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 192.168.238.238
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0001 = Type: Register (1)
    Reserved byte(s): 00
    Checksum: 0x9eff [correct]
    [Checksum Status: Good]
    PIM Options
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 20
    Identification: 0x0000 (0)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: PIM (103)
    Header checksum: 0xa86a [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.100.101
    Destination: 238.0.0.10
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
```

Capture 6-2: *PIM Register message from VTEP-101 to Anycast-RP.*

Phase 3: PIM Registration-Stop

Next, Spine-12 instructs VTEP-101 to stop to encapsulate the Multicast traffic within PIM Register message. This is done by sending a PIM Register-Stop message. Spine-12 uses the Anycast-RP IP address as a source IP and the IP address of interface NVE1 on VTEP-101 as a destination IP address.

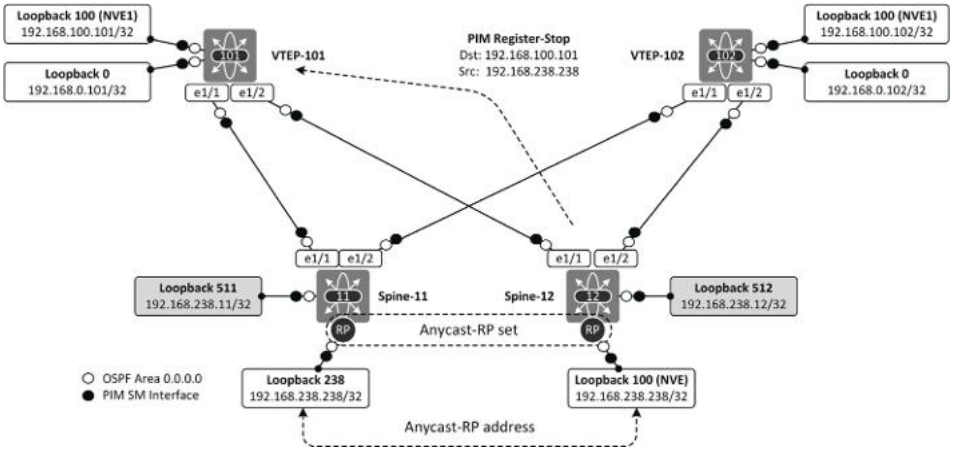


Figure 6-8: *PIM Register-Stop message from Spine-12 to VTEP-101.*

```
Ethernet II, Src: 5e:00:00:03:00:07, Dst: 5e:00:00:00:00:07
Internet Protocol Version 4, Src: 192.168.238.238, Dst: 192.168.100.101
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0010 = Type: Register-stop (2)
    Reserved byte(s): 00
    Checksum: 0xc8c6 [correct]
    [Checksum Status: Good]
    PIM Options
        Group: 238.0.0.10/32
        Source: 192.168.100.101
```

Capture 6-3: *PIM Register-Stop message from Spine-12 to VTEP-101.*

Phase 4: Anycast-RP peer notification

At this stage, Spine-11 has received PIM join message from VTEP-101 while Spine-12 has received the PIM Register message. To complete the registration process, Spine-11 will forward the PIM Register message received from VTEP-101 to its Anycast-RP Cluster member peer Spine-11. Spine-11 encapsulates the original PIM Register message with an outer IP header where it uses its' own unique Anycast-RP Cluster IP address 192.168.238.12

(Loopback 512) as a source and the unique Anycast-RP Cluster IP address 192.168.238.11 of Spine-11 as a destination.

When Spine-11 receives the forwarded PIM, it verifies that the sender is part of the same Anycast-RP Cluster by checking that the source IP address in the outer IP header is also defined as an Anycast-RP Cluster member.

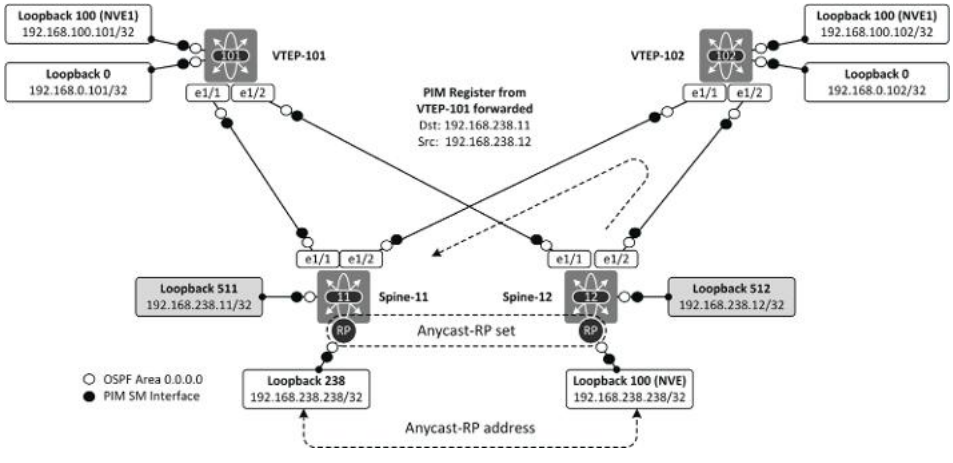


Figure 6-9: *PIM Register-Stop message from Spine-12 to Spine-11.*

Capture 6-4 shows the PIM Register message forwarded by Spine-12.

```
Ethernet II, Src: 5e:00:00:03:00:07 (5e:00:00:03:00:07), Dst: 5e:00:00:01:00:07
(5e:00:00:01:00:07)
Internet Protocol Version 4, Src: 192.168.238.12, Dst: 192.168.238.11
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0001 = Type: Register (1)
    Reserved byte(s): 00
    Checksum: 0x9eff [correct]
    [Checksum Status: Good]
    PIM Options
        Flags: 0x40000000
            0... .... .... .... .... .... .... .... = Border: No
            .1.. .... .... .... .... .... .... .... = Null-Register: Yes
        0100 .... = IP Version: IPv4 (4)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
```

Capture 6-4: *PIM Register-Stop message from Spine-12 to VTEP-101.*

Example 6-10 verifies that VTEP-101 has been joined to Multicast Group 238.0.0.10 as a receiver (*, 238.0.0.10/32) and inter-switch link E1/1 is listed in an Outgoing Interface List (OIL). In addition, an example verifies that VTEP-101 is registered as a source for the same Multicast Group (192.168.100.101, 238.0.0.10) and the inter-switch link E1/1 is listed in an Incoming Interface List (IIL).

```
Spine-11# sh ip mroute
IP Multicast Routing Table for VRF "default"

(*, 232.0.0.0/8), uptime: 00:03:57, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)


(*, 238.0.0.10/32), uptime: 00:00:06, pim ip
  Incoming interface: loopback238, RPF nbr: 192.168.238.238
  Outgoing interface list: (count: 1)
    Ethernet1/1, uptime: 00:00:06, pim


(192.168.100.101/32, 238.0.0.10/32), uptime: 00:03:07, pim ip
  Incoming interface: Ethernet1/1, RPF nbr: 192.168.0.101, internal
  Outgoing interface list: (count: 0)
```

Example 6-10: *M-RIB table on Spine-12.*

Example 6-11 shows that VTEP-101 is registered only as a source for the same Multicast Group (192.168.100.101, 238.0.0.10) and the inter-switch link E1/1 is listed in an Incoming Interface List (IIL) of Spine-12. Because Spine-12 hasn't received a PIM Join form VTEP-101, it is not installed into group-specific receiver entry.

```
Spine-12# sh ip mroute
IP Multicast Routing Table for VRF "default"

(*, 232.0.0.0/8), uptime: 00:04:24, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)


(192.168.100.101/32, 238.0.0.10/32), uptime: 00:03:46, pim ip
  Incoming interface: Ethernet1/1, RPF nbr: 192.168.0.101, internal
  Outgoing interface list: (count: 0)
```

Example 6-11: *M-RIB table on Spine-12.*

Examples 6-12 and 6-13 show the MRIB of both spine switches after the interface NVE1 in VTEP-102 has also brought up. Both VTEP switches are registered as a source for the Multicast group 238.0.0.10 while only VTEP-101 is registered as a receiver in Spine-11 and only VTEP-102 is registered as receiver for the same group in Spine-12. Though, both VTEP switches are listed in Group 238.0.0.10 OIL list (S,G).

```
Spine-11# sh ip mroute
<snipped>
(*, 232.0.0.0/8), uptime: 00:11:36, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)


(*, 238.0.0.10/32), uptime: 00:07:45, pim ip
  Incoming interface: loopback238, RPF nbr: 192.168.238.238
  Outgoing interface list: (count: 1)
    Ethernet1/1, uptime: 00:07:45, pim


(192.168.100.101/32, 238.0.0.10/32), uptime: 00:10:46, pim ip
  Incoming interface: Ethernet1/1, RPF nbr: 192.168.0.101, internal
  Outgoing interface list: (count: 0)


(192.168.100.102/32, 238.0.0.10/32), uptime: 00:00:45, pim mrib ip
  Incoming interface: Ethernet1/2, RPF nbr: 192.168.0.102, internal
  Outgoing interface list: (count: 1)
    Ethernet1/1, uptime: 00:00:45, pim
```

Example 6-12: *M-RIB table on Spine-11.*

```
Spine-12# sh ip mroute
<snipped>
 (*, 232.0.0.0/8), uptime: 00:11:59, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)


(*, 238.0.0.10/32), uptime: 00:02:17, pim ip
  Incoming interface: loopback238, RPF nbr: 192.168.238.238
  Outgoing interface list: (count: 1)
    Ethernet1/2, uptime: 00:02:17, pim


(192.168.100.101/32, 238.0.0.10/32), uptime: 00:11:20, pim ip
  Incoming interface: Ethernet1/1, RPF nbr: 192.168.0.101, internal
  Outgoing interface list: (count: 1)
    Ethernet1/2, uptime: 00:02:17, pim


(192.168.100.102/32, 238.0.0.10/32), uptime: 00:01:19, pim mrib ip
  Incoming interface: Ethernet1/2, RPF nbr: 192.168.0.102, internal
  Outgoing interface list: (count: 0)
```

Example 6-13: *M-RIB on Spine-12.*

Data Plane operation

Data Plane verification is done by pinging from host Cafe to host Beef. Both hosts are in the same L2 broadcast domain (VLAN10-VNI10000). The first three ICMP request is not answered due to unfinished ARP-process.

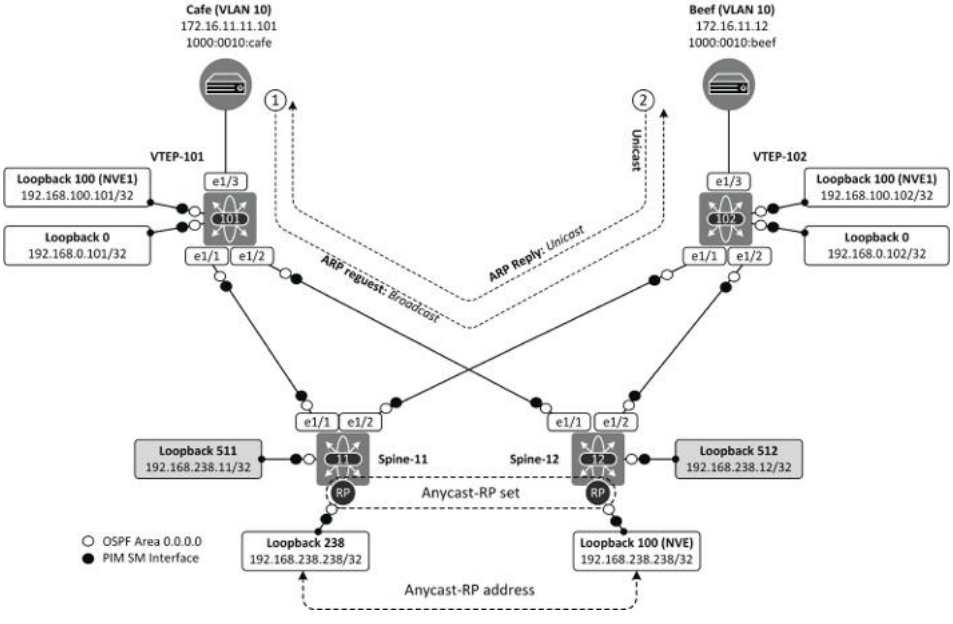


Figure 6-10: *ARP Process.*

```
Host-1#ping 192.168.11.12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.11.12, timeout is 2 seconds:
...!!
Success rate is 40 percent (2/5), round-trip min/avg/max = 18/22/26 ms
Host-1#
```

Example 6-14: Ping from host Cafe to host Beef.

ARP Request

Capture 6-5 taken from the interface E1/1 of VTEP-101 shows the encapsulated ARP Request message originated by host Cafe. The destination IP address in the outer IP header is Multicast Group address 238.0.0.10. VTEP-101 makes a forwarding decision based on ECMP hash and the result is E1/1. When Spine-11 receives the frame, it checks OIL list of Multicast Group 238.0.0.10 and based on that it forwards ARP Request message to VTEP-102. VTEP-102 receives the encapsulated ARP Request. It checks the VNI Id from the VXLAN header and switches the original L2 broadcast frame out of all interfaces that are participating in VLAN 10. Note that VTEP-101 learns the MAC address of host Cafe from the ingress frame. It advertises this MAC address to VTEP-102 by using BGP EVPN Route-Type 2 Update message.

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: IPv4mcast 0a
(01:00:5e:00:00:0a)
    Destination: IPv4mcast_0a (01:00:5e:00:00:0a)
    Source: 5e:00:00:00:00:07 (5e:00:00:00:00:07)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
User Datagram Protocol, Src Port: 63240, Dst Port: 4789
    Source Port: 63240
    Destination Port: 4789
    Length: 76
    [Checksum: [missing]]
    [Checksum Status: Not present]
    [Stream index: 0]
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: fa:16:3e:21:05:cd (fa:16:3e:21:05:cd), Dst: Broadcast
(ff:ff:ff:ff:ff:ff)
    Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Source: fa:16:3e:21:05:cd (fa:16:3e:21:05:cd)
    Type: ARP (0x0806)
    Trailer: 00000000000000000000000000000000000000
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: fa:16:3e:21:05:cd (fa:16:3e:21:05:cd)
    Sender IP address: 192.168.11.11
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.11.12
```

Capture 6-5: *ARP request sent by host Cafe captured from interface e1/1 on VTEP-101.*

ARP Reply

Host Beef receives the ARP Request. Since the IP address of the Target IP field in ARP request belongs to host Beef, it replies with ARP reply message. Host Beef learns the MAC address of requesting host Cafe from the sender MAC address field message and it sends an ARP Reply as unicast frame using the MAC address of host Cafe as a destination MAC. VTEP-102 has learned the MAC address of host Cafe from BGP Update sent by VTEP-101 and it forwards the ARP Reply message as a Unicast packet where outer IP address is the address of VTEP-101 and the destination MAC address belongs to host Cafe. After ARP process, there is Layer 2 connection between host Cafe and Abba.

```
Ethernet II, Src: 5e:00:00:02:00:07 (5e:00:00:02:00:07), Dst: 5e:00:00:00:00:07
(5e:00:00:00:00:07)
    Destination: 5e:00:00:00:00:07 (5e:00:00:00:00:07)
    Source: 5e:00:00:02:00:07 (5e:00:00:02:00:07)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.102, Dst: 192.168.100.101
User Datagram Protocol, Src Port: 56465, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: fa:16:3e:3a:05:f1 (fa:16:3e:3a:05:f1), Dst: fa:16:3e:21:05:cd
(fa:16:3e:21:05:cd)
    Destination: fa:16:3e:21:05:cd (fa:16:3e:21:05:cd)
    Source: fa:16:3e:3a:05:f1 (fa:16:3e:3a:05:f1)
    Type: ARP (0x0806)
    Trailer: 000000000000000000000000000000000000
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: fa:16:3e:3a:05:f1 (fa:16:3e:3a:05:f1)
    Sender IP address: 192.168.11.12
    Target MAC address: fa:16:3e:21:05:cd (fa:16:3e:21:05:cd)
    Target IP address: 192.168.11.11
```

Capture 6-6: *ARP Reply sent by host Beef captured from interface e1/2 on VTEP-102.*

References:

[RFC 4610] D. Farinacci and Y. Cai, “Anycast-RP Using Protocol Independent Multicast (PIM)”, RFC 4620, August 2006.

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Chapter 7: Layer 2 Multi-destination traffic - PIM BiDir.

This chapter explains the PIM BiDir (RFC5015) for L2 BUM traffic in VXLAN fabric. Figure 7-1 illustrates the example network and IP addressing schemes used throughout this chapter.

Spine-11 and Spine-12 both have the same IP address 192.168.238.6 assigned to Loopback 238. However, Spine-11 use subnet mask /29 while Spine-12 use subnet mask /28. Both switches redistribute the subnet into OSPF process. VTEP-101 and VTEP-102 install the subnet advertised by Spine-11 and Spine-12 into OSPF Data Base and into RIB. VTEP switches use IP address 192.168.238.1 as a Multicast RP. However, the IP address is not configured in any of the devices and this is where the name Phantom-RP comes from. All traffic towards destination 192.168.238.1 is routed based on longest match 192.168.238.0/29 advertised by Spine-11 as long as it is alive. If Spine-11 stops advertising the subnet, then the route advertised by Spine-11 will be used by VTEP switches.

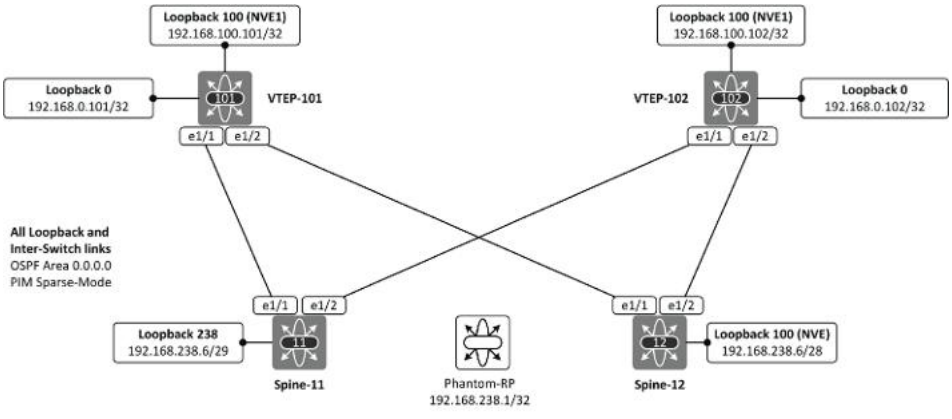


Figure 7-1: *Example topology with Anycast-RP - IP addresses.*

Configuration

Configure the loopback address 192.168.238.6 on both Spine switches. Use subnet mask /29 in Spine-11 and mask /28 in Spine-12. Enable OSPF (Area Id 0.0.0.0) and PIM-Sparse-Mode. Note that the OSPF network type has to be point-to-point, otherwise the loopback IP will be advertised as a host route with mask /32.

```
interface loopback238
  description ** random IP in Phantom-RP network **
  ip address 192.168.238.6/29
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
```

Example 7-1: *Loopback interface 238 configuration on Spine-11.*

```
interface loopback238
  description ** random IP in Phantom-RP network **
  ip address 192.168.238.6/28
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
```

Example 7-2: *Loopback interface 238 configuration on Spine-12.*

Define the IP address 192.168.238.1 as the RP of multicast groups 238.0.0.0/24 in all switches.

```
ip pim rp-address 192.168.238.1 group-list 238.0.0.0/24 bidir
```

Example 7-3: *Mcast RP definition in all switches.*

Example 7-4 shows that VTEP-101 has installed both subnets 192.168.238.0/23 and 192.168.28.0./24 into the RIB.

```
Leaf-101# sh ip route | b 192.168.238
192.168.238.0/28, ubest/mbest: 1/0
    *via 192.168.0.12, Eth1/2, [110/41], 00:40:14, ospf-UNDERLAY-NET, intra
192.168.238.0/29, ubest/mbest: 1/0
    *via 192.168.0.11, Eth1/1, [110/41], 00:40:03, ospf-UNDERLAY-NET, intra
```

Example 7-4: *RIB of VTEP-101 concerning subnets 192.168.238.0/29 and /28.*

However, the only route advertised by Spine-11 is used for routing for destination IP address 192.168.238.1.

```
Leaf-101# sh ip route 192.168.238.1
<snipped>
192.168.238.0/29, ubest/mbest: 1/0
    *via 192.168.0.11, Eth1/1, [110/41], 00:38:37, ospf-UNDERLAY-NET, intra
```

Example 7-5: *Route to Phantom-RP on VTEP-101.*

Control Plane Operation

The next section discusses the Control Plane operation of PIM BiDir. The NVE1 is first brought down in both VTEP switches.When the state of the interface NVE1 is changing from down to up, it triggers a PIM Join process. VTEP-101 joins Multicast Group 238.0.0.10 by sending a PIM join message to Rendezvous Point out of its' E1/1. The same process happens in VTEP-102. The packet capture 7-1 is captured from the VTEP-101 interface E1/1.

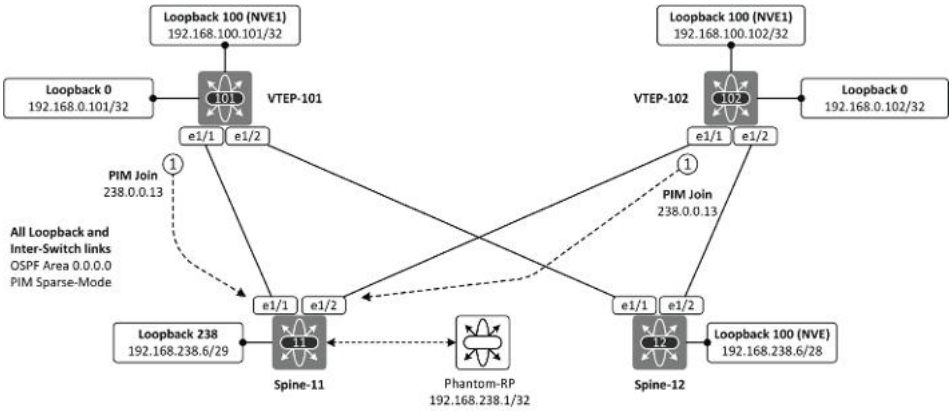


Figure 7-2: *Example topology with Anycast-RP - IP addresses.*

Capture 7-1 shows the PIM Join message sent by VTEP-101.

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: IPv4mcast_0d
(01:00:5e:00:00:0d)
Internet Protocol Version 4, Src: 192.168.0.101, Dst: 224.0.0.13
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0011 = Type: Join/Prune (3)
    Reserved byte(s): 00
    Checksum: 0x7482 [correct]
    [Checksum Status: Good]
    PIM Options
        Upstream-neighbor: 192.168.0.11
        Reserved byte(s): 00
        Num Groups: 1
        Holdtime: 210
        Group 0: 238.0.0.10/32
            Num Joins: 1
                IP address: 192.168.238.1/32 (SWR)
            Num Prunes: 0
```

Capture 7-1: *PIM Join message sent by VTEP-101.*

Based on these received PIM join messages, Spine-11 adds the Interfaces E1/1 and E1/2 into Outgoing Interface List (OIL) for group 238.0.0.10.

```
Spine-11# sh ip mroute
IP Multicast Routing Table for VRF "default"

(*, 232.0.0.0/8), uptime: 01:00:27, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)
```

```
(*, 238.0.0.0/24), bidir, uptime: 00:58:01, pim ip
  Incoming interface: loopback238, RPF nbr: 192.168.238.1
  Outgoing interface list: (count: 1)
    loopback238, uptime: 00:08:11, pim, (RPF)


(*, 238.0.0.10/32), bidir, uptime: 00:54:51, pim ip
  Incoming interface: loopback238, RPF nbr: 192.168.238.1
  Outgoing interface list: (count: 3)
    Ethernet1/2, uptime: 00:08:10, pim
    Ethernet1/1, uptime: 00:08:10, pim
    loopback238, uptime: 00:08:11, pim, (RPF
```

Example 7-6: *M-RIB on VTEP-101.*

The bidirectional Multicast Tree is now ready. Spine-12 is not participating in the Shared Tree at this moment. In case of Spine-11 failure, Spine-12 still advertises RPA network with mask /28 (remember that Spine-11 uses mask /29).

References

[RFC 5015] M. Handley et al., “Bidirectional Protocol Independent Multicast (BIDIR-PIM)”, RFC 5015, October 2007

[RFC 7761] B. Fenner et al., “Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)”, RFC 7761, March 2016.

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Chapter 8: BGP EVPN VXLAN Configuration and building blocks.

This chapter introduces the basic building block and its configuration. It starts by explaining the Intra-VNI switching and its configuration. Next, it discusses tenant-specific Inter-VN routing. Figure 8-1 shows the example topology and addressing scheme used in this chapter.

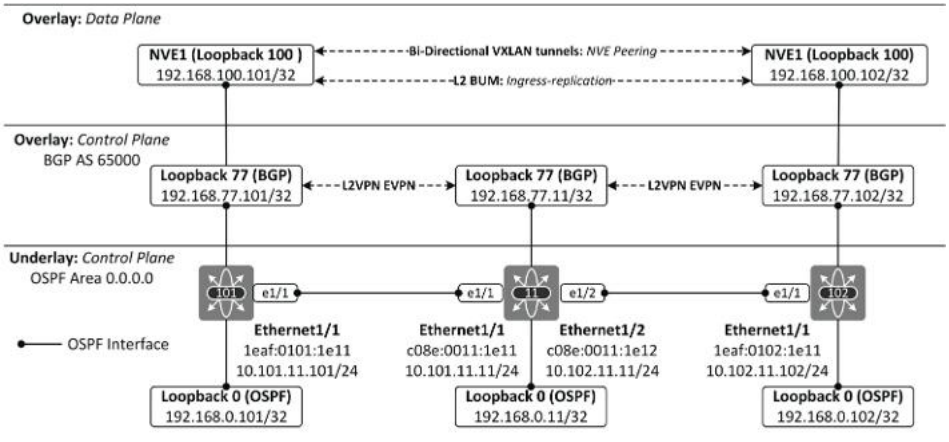


Figure 8-1: *Example topology and addressing scheme.*

BGP EVPN VXLAN Building Blocks for Intra-VNI switching

This section introduces the basic building blocks of BGP EVPN VXLAN fabric from the Intra-VNI switch perspective. The Underlay Network solution used in this section is Unicast-only (no Multicast support) using OSPF for routing exchange. The Underlay Network routing protocol has four main tasks from the VXLAN perspective:

Offer an IP connectivity between Loopback interfaces used for Overlay Network BGP peering:

Virtual Tunnel End Point (VTEP) switches Leaf-101 and Leaf-102 used in this example establish iBGP L2VPN EVPN peering with Spine-11 by using their logical interface Loopback77. Leaf-switches exchanges L2VPN EVPN NLRIs (Network Layer Reachability Information) over these peering. This section introduces the L2VPN EVPN NLRI route-type 3 *"Inclusive Multicast Route"*, which is used by Leaf-switches to introduce their willingness to participate in Ingress-Replication L2 multi-destination tree. This multi-destination tree is used for Intra-VNI L2BUM traffic (mainly ARP and DHCP). In the case of a tenant-based L3 Multicast routing, the "*Tenant Routed Multicast"* (TRM) solution is required.

Offer an IP connectivity between Loopback interfaces that are used for Overlay Network NVE peering:

Network Virtualization Edge (NVE) interface on Leaf-101 and Leaf-102 will establish an NVE peering between each other by using the IP address of interface Loopback100. NVE interfaces are used for VXLAN encapsulation/decapsulation for user data. The first requirement for NVE peering between leaf-switches is an NVE-to-NVE IP reachability. The second requirement that is needed to bringing up the NVE peering is two-way BGP L2VPN EVPN BGP Update messages exchange between leaf-switches. It could be the *MAC Advertisement Route,* which carries information about the VM MAC address. In addition, if Ingress-Replication is used for L2BUM traffic with BGP, the trigger for NVE tunnel could be *Inclusive Multicast Route* BGP Update. Both of these route-types includes information about advertising switch and VNI for which the specific Update belongs to and based on that information, switches will build an NVE peering. The IP address of NVE interface is used as a next-hop value in MP-REACH_NLRI PA in BGP update messages. It is also used in the outer IP header in VXLAN encapsulated frames.

Advertise all Inter-Switch links subnets:

Inter-switch links IP address has to be reachable for all switches for next-hop resolving.

Advertise the Multicast Rendezvous Point (RP) IP address:

In a solution where Multicast is used for L2BUM traffic such as ARP-request forwarding, Multicast routing has to be enabled in an Underlay Network. The RP of given Multicast Group has to reachable for all switches in local VXLAN fabric.

Underlay Network: OSPF configuration

OSPF configuration is straightforward, enables OSPF feature, starts the OSPF process and defines the OSPF Router-Id. In addition, enable OSPF in Loopback interfaces and in all Inter-Switch link.

```
Leaf-101# sh run ospf
<snipped>
feature ospf

router ospf UNDERLAY-NET
  router-id 192.168.0.101

interface loopback0
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface loopback77
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface loopback100
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface Ethernet1/1
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
```

Example 8-1: *OSPF configuration on Leaf-101.*

Overlay Network: BGP L2VPN EVPN configuration

Example 8-2 illustrates the BGP configuration on Leaf-101. Spine-11 is an iBGP neighbor and Leaf-101 exchange BGP L2VPN EVPN Updates. The "***nv overlay evpn***" –command is required before BGP neighbor can be specified as an L2VPN EVPN peer. The configuration is identical in both leaf –switches, excluding the BGP Router-Id value.

```
Leaf-101# sh run bgp
<snipped>
feature bgp
nv overlay evpn
!
router bgp 65000
  router-id 192.168.77.101
  address-family ipv4 unicast
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
```

Example 8-2: *BGP configuration on Leaf-101.*

The BGP configuration of Spine-11 follows the same construction with an exception that leaf-switches are defined as route-reflector clients. Output taken from Spine-11 shows that Leaf-101 and Leaf-102 have both established iBGP L2VPN EVPN peering with Spine-11 but neither switch has sent any BGP Updates.

```
Spine-11# sh bgp l2vpn evpn summary
<>
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.77.101  4 65000      21      21        4    0    0 00:15:13 0
192.168.77.102  4 65000      21      21        4    0    0 00:15:15 0
```

Example 8-3: *BGP verification.*

Overlay Network: NVE Peering

Feature ***"nv overlay"*** is required in order to configure NVE-interface. Host reachability information is exchange by using BGP with interface Loopback100 as a source IP address. NVE interface is only needed in Leaf-101 and Leaf-102. The command "***source-interface hold-down-time***" can be used for delaying the advertisement of the NVE loopback interface IP address until the Overlay Network Control Plane is converged. The default value is 180 seconds and the range is 0-1000 seconds.

```
feature nv overlay
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
```

Example 8-4: *Interface NVE1 on Leaf-101.*

Example 8-5 shows that Leaf-101 has no NVE peers at the moment, which is ok at this phase.

```
Leaf-101# sh nve peers detail
Leaf-101#
```

Example 8-5: *Leaf-101 NVE peers.*

Example 8-6 verifies that the NVE1 interface is up and it is using IP address 192.168.100.101 as a source address. The packet sends over the NVE interface will be encapsulated with a VXLAN header. VXLAN is MAC in IP/UDP encapsulation where the original frames are encapsulated with an outer MAC/IP/UDP/VXLAN header. These headers are explored later when the actual encapsulated frame is sent over NVE interface.

```
Leaf-101# sh nve interface nve1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 5000.0001.0007
 Host Learning Mode: Control-Plane
 Source-Interface: loopback100 (primary: 192.168.100.101, secondary: 0.0.0.0)
 Source Interface State: Up
 Virtual RMAC Advertisement: No
 NVE Flags:
 Interface Handle: 0x49000001
 Source Interface hold-down-time: 180
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: N/A
 Interface state: nve-intf-add-complete
```

Example 8-6: *Interface NVE1 on Lwaf-101.*

Overlay Network: Host Mobility Manager

The VXLAN service in NX-OS has a *Host Mobility Manager (HMM)* component. The HMM keeps track of MAC address moving inside VXLAN fabric switches. The operation of HMM is explained later in the context of the Control Plane learning process. Example 8-7 shows the command that enables the HMM feature.

```
feature fabric forwarding
```

Example 8-7: *Enabling the "Host Mobility Manager" function on Leaf-101.*

Overlay Network: Anycast Gateway (AGW)

Virtual Machines (VMs) may move from one leaf switch to another in VXLAN fabric. Each leaf switch operates as an Anycast Gateway for VLAN SVIs (Switched Virtual Interface). In practice, this means that the VLAN specific SVI is configured in every leaf switch where the VLAN exists. In addition, each SVI uses the same Anycast MAC-address regardless of L3 SVI or tenant. Features needed for SVI and Anycast Gateway are shown in Example 8-8

```
feature interface-vlan
!
fabric forwarding anycast-gateway-mac 0001.0001.0001
```

Example 8-8: *Enabling the "Anycast Gateway" function on Leaf-101.*

Overlay Network: VLAN based service

The example VXLAN fabric uses VLAN based Virtual Network Segments (VN-Segment). The example 8-9 shows the configuration.

```
feature vn-segment-vlan-based
```

Example 8-9: *Enabling "VLAN based VNI-segment" function on Leaf-101.*

Overlay Network: TCAM modification

TCAM (Ternary Content Addressable Memory) is a part of the specialized memory block, which supports fast parallel lookups. The memory slices on TCAM consists of 256 bytes or 512 bytes blocks/banks. There are one ingress TCAM and one egress TCAM whose size varies based on the switching platform. The TCAM modification used in this example (Router-ACL, vPC-Convergence, and arp-ether) is shown in example 8-10.

```
hardware access-list tcam region racl 512
hardware access-list tcam region vpc-convergence 256
hardware access-list tcam region arp-ether 256 double-wide
```

Example 8-10: *Enabling "VLAN based VNI-segment" function on Leaf-101.*

Example 8-11 summarizes the VXLAN configuration required for switches before Layer 2 Intra-VNI or Layer 3 Inter-VNI services can be implemented into VXLAN fabric.

```
nv overlay evpn
feature ospf
feature bgp
feature fabric forwarding
feature interface-vlan
feature vn-segment-vlan-based
feature nv overlay
!
fabric forwarding anycast-gateway-mac 0001.0001.0001
!
hardware access-list tcam region racl 512
hardware access-list tcam region vpc-convergence 256
hardware access-list tcam region arp-ether 256 double-wide
!
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
!
interface Ethernet1/1
  no switchport
  mac-address 1eaf.0101.1e11
  medium p2p
  ip address 10.101.11.101/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  no shutdown
!
```

```
interface loopback0
  description ** RID/Underlay **
  ip address 192.168.0.101/32
  ip router ospf UNDERLAY-NET area 0.0.0.0
!
interface loopback77
  description ** BGP peering **
  ip address 192.168.77.101/32
  ip router ospf UNDERLAY-NET area 0.0.0.0
!
interface loopback100
  description ** VTEP/Overlay **
  ip address 192.168.100.101/32
  ip router ospf UNDERLAY-NET area 0.0.0.0
!
router ospf UNDERLAY-NET
  router-id 192.168.0.101
router bgp 65000
  router-id 192.168.77.101
  address-family ipv4 unicast
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
```

Example 8-11: *Complete VXLAN service configuration on Leaf-101.*

Intra-VNI service (L2VNI) in VXLAN Fabric

This section explains the Intra-VNI solutions used in VXLAN Fabric. One way to define the VNI is a broadcast domain (VLAN) stretched over the VXLAN fabric. Intra-VNI implementation starts by creating the L2 VLAN and attaching it to VN-Segment. Example 8-12 shows how to create vlan and attached it to VNI 10000.

```
vlan 10
  name VLAN10-mapped-to-VNI10000
  vn-segment 10000
```

Example 8-12: *Adding VLAN and assign it to VNI.*

The next step is to create an EVPN instance for VNI 10000. Route-Distinguisher (RD) is used to differentiating possible overlapping MAC addresses in different VNIs. RD is formed based on BGP RID and base value 32767 plus VLAN Id. Leaf-101 BGP RID is 192.168.77.101 and the VLAN Id used in this example is 10. This gives the RD 192.168.77.101:32777 for all MAC addresses participating in VLAN10/VNI 10000.

Each MAC Advertisement route exported to the BGP process carries VNI specific Route-Target value that is taken from EVPN instance for given VNI. The automatic creation of Route-Target is formed based on AS number: VNI-Id. This gives the VNI value 65000:10000 for EVPN instance VNI 10000. The RT is the same in all leaf-switches where VLAN 10 is used for VNI 10000. Note that VLAN Id does not have to be the same in all VTEP switches. VLAN Id has only local (switch) significance while VNI-Id has Fabric wide significance.

```
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
```

Example 8-13: *Creating EVPN instance for VNI 10000.*

The last two steps are to add the VNI10000 to the VXLAN process and define the L2BUM traffic replication mode. There are two modes for L2BUM replication, first is the *Multicast Mode* explained in chapters 6 and 7. The second one, *Ingress-Replication* (IR) is explained in this section. In IR mode peers leaf-switches can be defined either manually or by resolving automatically by using BGP. This section introduces the BGP solution. Ingress-Replication differs from Multicast L2BUM replication in two ways. In Multicast Mode, switch sends only one copy of L2BUM traffic to given Multicast group Rendezvous-Point without VXLAN encapsulation. In Ingress-Replication mode, the leaf-switch will send copies of L2BUM traffic to each leaf-switches participating in L2BUM delivery tree over VXLAN tunnel.

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  member vni 10000
    ingress-replication protocol bgp
```

Example 8-14: *Adding VNI to VXLAN service and defining the L2BUM handling mode.*

Figure 8-2 illustrates the hardware resources reserved for VLAN10/VNI1000. First, the VLAN 10 is activated and the MAC address can be installed into MAC address-table. Next, there is an L2 Routing and Forwarding Instance a.k.a L2RIB where the locally learned MAC address information is installed by L2 Forwarding component (L2FWDR). From the L2RIB the MAC address information is exported into BGP table with VNI specific Path Attributes (PA) values such as a Route-Target (Extended Community PA) and a Route-Distinguisher (MP-REACH-NLRI PA). Because there are no connected hosts in VLAN 10 at this moment, the MAC address-table and L2RIB are empty. Also, there are no route-type 2 routes (MAC Advertisement Route) in BGP table concerning L2VNI 10000 for the same reason. Though, there is a route-type 3 (Inclusive Multicast Route). Leaf-101 uses this route-type to inform its willingness to join the L2BUM delivery tree for VNI10000. Switches hosting VNI10000 will import this route int BGP table (BRIB) based on the auto-generated RT value.

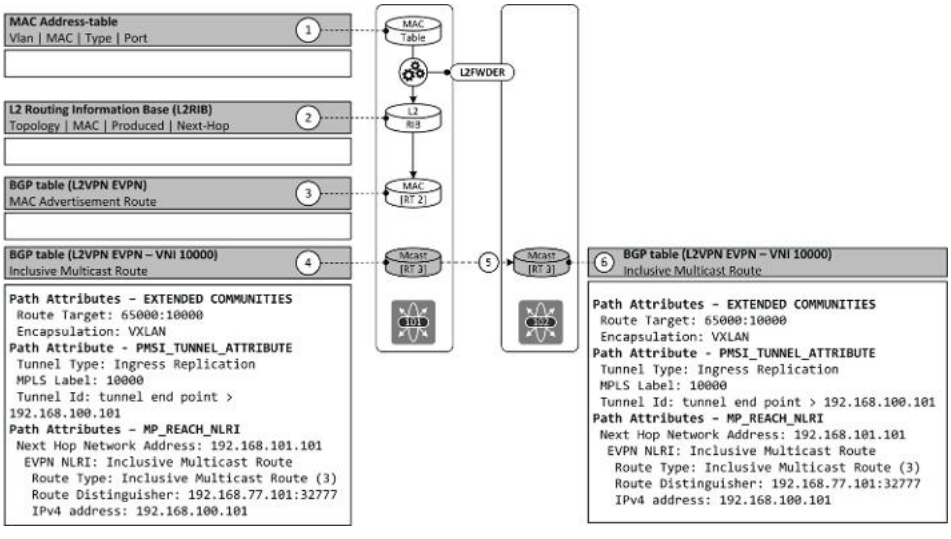


Figure 8-2: *Hardware resources and BGP route-type 3.*

P-Multicast Service Instance (PMSI) Path Attribute shown in figure 8-2 describes the PSMI tunnel end-point for the Multi-Destination tree for VNI 10000. The MPLS label describes the Virtual Network Identifier (VNI) for this Multi-destination tree. Capture 8-1 shows the BGP L2VPN EVPN Route-Type 3 BGP Update message send by Leaf-101.

```
Ethernet II, Src: 1e:af:01:01:1e:11 (1e:af:01:01:1e:11), Dst: c0:8e:00:11:1e:11
(c0:8e:00:11:1e:11)
Internet Protocol Version 4, Src: 192.168.77.101, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 27001, Dst Port: 179, Seq: 1, Ack: 1,
Len: 100
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 100
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 77
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 16
            Carried extended communities: (2 communities)
                Route Target: 65000:10000
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
        Path Attribute - PMSI_TUNNEL_ATTRIBUTE
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: PMSI_TUNNEL_ATTRIBUTE (22)
            Length: 9
            Flags: 0
```

```
            Tunnel Type: Ingress Replication (6)
            0000 0000 0010 0111 0001 .... = MPLS Label: 625
            Tunnel ID: tunnel end point -> 192.168.100.101
        Path Attribute - MP_REACH_NLRI
            Type Code: MP_REACH_NLRI (14)
            Length: 28
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (19 bytes)
                EVPN NLRI: Inclusive Multicast Route
                    Route Type: Inclusive Multicast Route (3)
                    Length: 17
                    Route Distinguisher: (192.168.77.101:32777)
                    Ethernet Tag ID: 0
                    IP Address Length: 32
                    IPv4 address: 192.168.100.101
```

Capture 8-1: *Inclusive Multicast Route (route-type 3).*

Figure 8-3 illustrates the VXLAN tunnel that has now been set up. Both Unicast and L2BUM frames originated by the local host will be encapsulated with a tunnel header that includes outer MAC and IP headers, where source MAC address is taken from the egress interface E1/1 and the source IP is taken from the interface NVE1. The destination IP address will be set to an interface NVE1 of destination leaf-switch. After the MAC and IP headers, there is a UDP header where the destination port is always set to 4789 for VXLAN. The source UDP port is the result of a 5-tuple hash. In addition, there is a VXLAN header that describes the VNI where carried frame belongs to. The Control Plane and Data Plane operation are discussed in detail in chapter 9. Note, in the case where the Ingress-Replication mode is static, the NVE peering does not come up until one BGP Update message is sent by both leaf-switches.

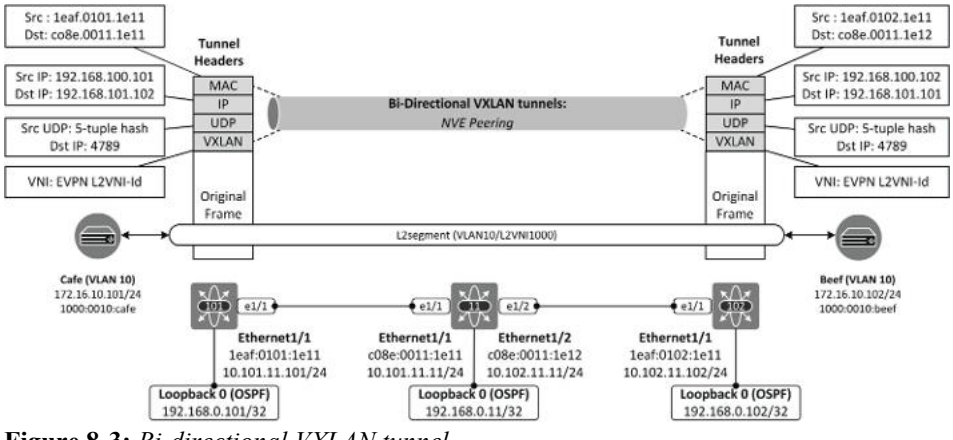


Figure 8-3: *Bi-directional VXLAN tunnel.*

Layer 2 Intra VNI service can be used for example in the case where the VXLAN fabric offers only an L2 switching path between. One example is the solution where the Firewall is used as a gateway between Inter-VNI segments.

Tenant based Inter-VNI Routing (L3VNI) in VXLAN Fabric

The previous section introduces the basic building blocks and configuration for Intra-VNI switching. This section discusses tenant-based Inter-VNI routing.

The first step is to set up a tenant-based routing domain by configuring VRF context (Virtual Routing and Forwarding instance) for specific customer/application (example 8-15). The VNI used for inter-VLAN routing in this example is 10077. This VNI is used in VXLAN header as tenant identifier. Just like in previous L2VNI examples, auto-generated values for both RD and RT are used. RD is formed based on switch BGP RID:<VRF Id>. Example 8-16 shows that VRF-Id of TENANT77 is three. This gives RD 192.168.77.101:3 for Inter-VNI routes. RT, in turn, is formed based on BGP AS:VNI-Id. This gives RT 65000:10077 for Inter-VNI routes.

```
vrf context TENANT77
  vni 10077
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
```

Example 8-15: *configuring VRF context named TEANANT77on Leaf-101.*

```
Leaf-101# sh vrf TENANT77
VRF-Name                           VRF-ID State   Reason
TENANT77                                3 Up      --
```

Example 8-16: *VRF context TEANANT77 VRF-Id on Leaf-101.*

The next step is to define the actual routing interface for Inter-VNI routing. It is done by specifying the routing interface and L2 VLAN for it in every switch that hosts the VRF. There is no IP address attached to L3 interface because it is used only in Data Plane (it is not advertised in any routing protocol). When a VTEP switch receives an IP packet to another subnet (inside tenant) from its directly connected host, switch routes the packet over the L3 interface. The VNI-Id added to VXLAN header is set to value that is defined under the VRF Context configuration. L2VLAN for routing interface is used for reserving hardware resources. VXLAN is a MAC-in-IP/UDP encapsulation. So there is a need for inner MAC address also for routed Inter-VNI packets. The address of the sending host is naturally out of the question so the inner source MAC address is taken from the L2 VLAN associated with L3 routing interface. MAC address depends on the Underlay Network Inter-switch IP addressing. If Inter-Switch links use Unnumbered interface addressing scheme, the MAC address attaches to L2 VLAN is the switch system MAC, otherwise, the MAC address of the uplink interface is used. The destination MAC address on inner frame is set to destination VTEP switch MAC (in case of Unicast packet). Capture 8-2 shows the ICMP request sent by host Cafe in VLAN 10 (172.16.10.101) connected to Leaf-101 to host Beef in VLAN 20 (172.16.20.102) connected to remote Leaf-102.

```
vlan 77
  name TENANT77
  vn-segment 10077
!
interface Vlan77
  no shutdown
  mtu 9216
  vrf member TENANT77
  ip forward
```

Example 8-17: *L2 VLAN and L3 SVI for Inter-VNI routing inside TEANANT77.*

```
Ethernet II,
 Src: 1e:af:01:01:1e:11, Dst: c0:8e:00:11:1e:11
Internet Protocol Version 4,
 Src: 192.168.100.101, Dst: 192.168.100.102
User Datagram Protocol,
 Src Port: 63500, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10077
    Reserved: 0
Ethernet II,
 Src: 1e:af:01:01:1e:11, Dst: 50:00:00:03:00:07
Internet Protocol Version 4,
 Src: 172.16.10.101, Dst: 172.16.20.102
Internet Control Message Protocol
```

Capture 8-2: *Routed IP packet from VLAN10 to VLAN20 taken from Leaf-101 uplink.*

As the last step, the VNI used for routing is attached to NVE1 interface.

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  member vni 10000
    ingress-replication protocol bgp
  member vni 10077 associate-vrf
```

Example 8-18: *Associating VNI 10077 into interface NVE1.*

The previous section describes the tenant-based routing. This short section adds an SVI for VLAN 10. Interface VLAN is attached to VRF TENANT77 and ip address 172.16.10.1/24 is attached to it. The example 8-8 defines the tenant wide Anycast Gateway (AGW) MAC address 0001.0001.0001 used in every VLAN. The last command "*fabric forwarding mode anycast-gateway*" enables AGW for VLAN 10.

```
interface Vlan10
  no shutdown
  vrf member TENANT77
  ip address 172.16.10.1/24
  fabric forwarding mode anycast-gateway
```

Example 8-19: *SVI for VLAN 10.*

When host Cafe in VLAN 10 sends data packets to host Beef in VLAN 20, the packet is first sent to Leaf-101 which is AGW for network 172.16.10.0/24. Leaf-101 does routing lookup and forwards packet to Leaf-102 by using VNI 10077 in VXLAN header. When Leaf-102 receives the packet it knows that the packet belongs to specific tenant and does routing lookup of tenant RIB. Then packet is sent to host Beef. Chapter 9 discusses routing model in detail.

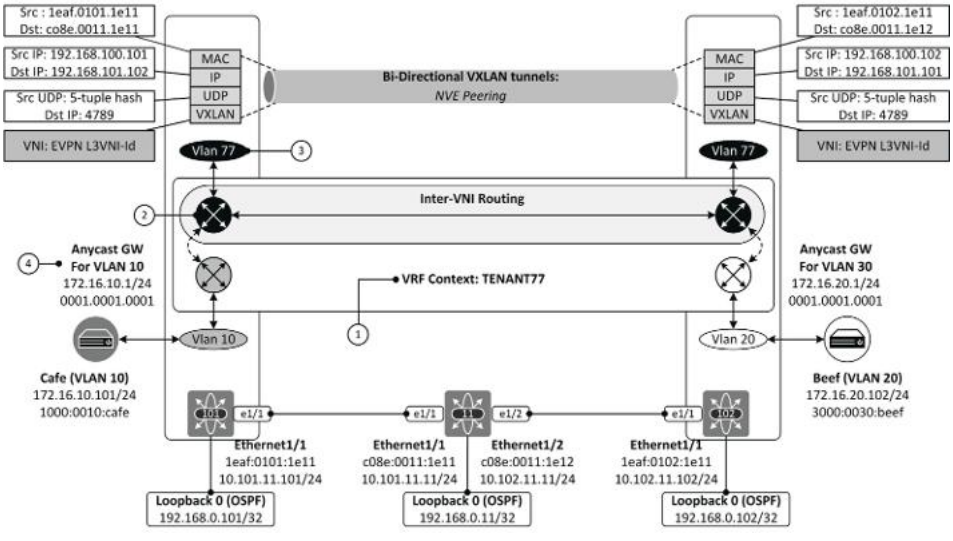


Figure 8-4: *Inter-VNI Routing model.*

This chapter introduces the basic VXLAN building blocks and their configuration. The next chapter introduces the BGP L2VPN EVPN Control Plane operation.

References

Nexus 9000 TCAM Carving: https://www.cisco.com/c/en/us/support/docs/switches/nexus-9000-series-switches/119032-nexus9k-tcam-00.html

Cisco Nexus 9000 Series NX-OS Security Configuration Guide, Release 7.x: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/7-x/security/configuration/guide/b_Cisco_Nexus_9000_Series_NX-OS_Security_Configuration_Guide_7x/b_Cisco_Nexus_9000_Series_NX-OS_Security_Configuration_Guide_7x_chapter_01001.html#concept_846AE66E9B2C4E0EAA3E54FBE51C4A87

Configuring VXLAN BGP EVPN: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/7-x/vxlan/configuration/guide/b_Cisco_Nexus_9000_Series_NX-OS_VXLAN_Configuration_Guide_7x/b_Cisco_Nexus_9000_Series_NX-OS_VXLAN_Configuration_Guide_7x_chapter_0100.pdf

A TCAM-Based Distributed Parallel IP Lookup Scheme and Performance Analysis: https://www.researchgate.net/publication/3335215_A_TCAM-based_distributed_parallel_IP_lookup_scheme_and_performance_analysis

Chapter 9: BGP EVPN VXLAN Control and Data Plane Operation.

The first section introduces how the *Intra-VNI* communication (L2VNI) within subnet over VXLAN fabric is achieved. It starts by explaining the process how switches learn the local VMs MAC address information and how this information is propagated through the switch into BGP process and advertised to remote switch. It also explains the difference between the usage of MAC and MAC-IP information.

The second section discusses how *Inter-VNI* communication (L3VNI) between subnets belonging to same tenant can be achieved. It explains the process how switches learn the localhost IP address information and how this information is sent to remote switch.

The third section explains the process of how tenant-specific subnets are advertised between leaf switches. It also explains why and when prefix advertisements are needed.

All three sections discuss both Control Plane and Data Plane operation. Figure 9-1 illustrates the building blocks explained in this chapter.

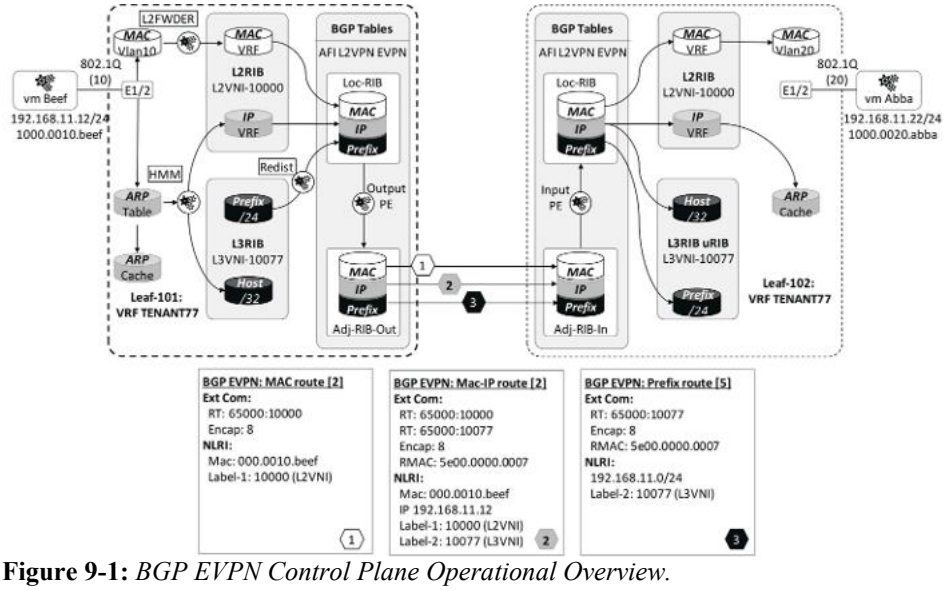


Figure 9-1: *BGP EVPN Control Plane Operational Overview.*

MAC address learning process

This section describes how the local VTEP switch learns the MAC addresses of its directly connected hosts from the ingress Ethernet frame and installs the information into MAC address-table as well as into VNI specific Layer 2 Routing Information Base (L2RIB) also called MAC VRF. In addition, this section explains how MAC information is exchanged between leaf switches by using BGP EVPN Route-Type 2 (Mac Advertisement Route) updates. Figure 9-1 illustrates the overall operation.

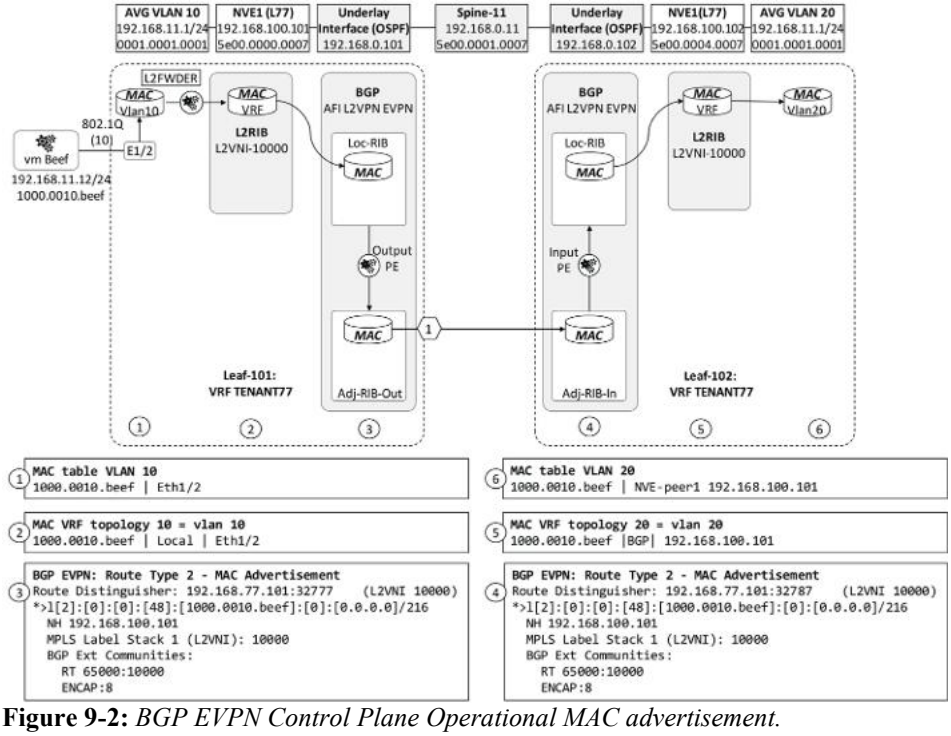


Figure 9-2: *BGP EVPN Control Plane Operational MAC advertisement.*

Phase 1: MAC Address-Table update

Virtual Machine Beef comes up. It expresses its existence to a network and validates the uniqueness of its IP-address by sending a Gratuitous ARP (GARP). VTEP switch Leaf-101 receives the GARP message from interface E1/2 and it stores the MAC address information into the MAC address table.

Example 9-1 shows the MAC address-table of local VTEP switch Leaf-10. The MAC address 1000.0010.beef is located behind port E1/2 and it belongs to VLAN 10. Note that the default MAC entry aging time is 1800 seconds.

```
Leaf-101# show system internal l2fwder mac
<snipped>
   VLAN     MAC Address       Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------
*    10    1000.0010.beef   dynamic   00:03:27   F     F     Eth1/2
```

Example 9-1: *show system internal l2fwder mac*

Phase 2: L2RIB Update

The new MAC address information is also installed into VNI10000 (aka EVPN Instance) specific Layer 2 Routing Information Base (L2RIB) by Layer 2 Forwarder (L2FWDER) component. The VNI specific L2RIB database is shown as MAC VRF in figure 9-2.

Example 9-3 illustrates the L2RIB of L2VNI 10000 on Leaf-101. The L2RIB contains the MAC address and next-hop information as well as information about learning method. The Topology-Id describes the VLAN where MAC address 1000.0010.beef belongs to.

```
Leaf-101#  show l2route evpn mac evi 10
<snipped>
Topology    Mac Address    Prod   Flags          Seq No     Next-Hops
----------- -------------- ------ -------------- ---------- --------------
10          1000.0010.beef Local  L,             0          Eth1/2
```

Example 9-2: *show l2route evpn mac evi 10*

Example 9-3 shows the Topology-Id (VLAN) to the L2VNI mapping information.

```
Leaf-101# show vlan id 10 vn-segment
VLAN Segment-id
---- -----------
10   10000
```

Example 9-3: *show vlan id 10 vn-segment*

Example 9-4 illustrates the process of how the L2FWDER component notices the new MAC address entering from the interface E1/2 (interface index 1a00200). The received frame has an 802.1Q tag, with the VLAN-Id 10. Based on VLAN-Id 10, the L2FWDER component knows that the MAC address information belongs to VNI10000.

```
Leaf-101# show system internal l2fwder event-history events | i beef
l2fwder_dbg_ev, 690 l2fwder_vxlan_mac_update, 886MAC move 1000.0010.beef (10)
0x0 -> 0x1a000200
l2fwder_dbg_ev, 690 l2fwder_l2rib_add_delete_local_mac_routes, 154Adding route
topo-id: 10, macaddr: 1000.0010.beef, nhifindx: 0x1a000200
l2fwder_dbg_ev, 690 l2fwder_l2rib_mac_update, 736MAC move 1000.0010.beef (10)
0x0 -> 0x1a000200
l2fwder_construct_and_send_macmv_ntf_per_cookie, 5261 mac 1000.0010.beef vlan 1
new if_index = 1a000200, old if_index = 0, is_del=0
```

Example 9-4: *show system internal l2fwder event-history events | i beef*

Example 9-5 verifies the if-index to physical interface mapping information.

```
Leaf-101# show interface snmp-ifindex | i 0x1a000200
Eth1/2          436208128  (0x1a000200)
```

Example 9-5: *show interface snmp-ifindex | i 0x1a000200*

Example 9-6 shows from top to down the update process of L2RIB.

```
Leaf-101# sh system internal l2rib event-history mac | i beef

Rcvd MAC ROUTE msg: (10, 1000.0010.beef), vni 0, admin_dist 0, seq 0, soo 0,
(10,1000.0010.beef):Mobility check for new rte from prod: 3
(10,1000.0010.beef):Current non-del-pending route local:no, remote:no, linked
mac-ip count:1
(10,1000.0010.beef):Clearing routelist flags: Del_Pend,
(10,1000.0010.beef,3):Is local route. is_mac_remote_at_the_delete: 0
(10,1000.0010.beef,3):MAC route created with seq 0, flags L, (),
(10,1000.0010.beef,3): soo 0, peerid 0, pc-ifindex 0
(10,1000.0010.beef,3):Encoding MAC best route (ADD, client id 5)
(10,1000.0010.beef,3):vni:10000 rt_flags:L, admin_dist:6, seq_num:0
ecmp_label:0 soo:0(--)
(10,1000.0010.beef,3):res:Regular esi:(F) peerid:0 nve_ifhdl:1224736769
mh_pc_ifidx:0 nh_count:1
(10,1000.0010.beef,3):NH[0]:Eth1/2
```

Example 9-6: *show system internal l2rib event-history mac | i beef*

Why there are two almost similar L2 Databases per VLAN/VN in VTEP switches (MAC address-table and L2RIB)? Routes can be sent to BGP only if the route is first installed into RIB and vice versa.

Phase 3: BGP MAC Route Export on Local VTEP

VTEP switch Leaf-101 exports the MAC route from the L2RIB into BGP Loc-RIB, from where it is installed through the Output Policy Engine into Adj-RIB-Out. The BGP-process of Leaf-101 attaches the Path Attributes based on the BGP peer type (iBGP/eBGP/RR-Client) and sends the BGP EVPN Route-Type 2 Update to Spine-11 (Route-Reflector). Spine-11 forwards the message to its RR-Client Leaf-102. The BGP Path Attribute "*MP_REACH_NLRI*" carried in the BGP Update contains NLRI information, where the address information includes MAC address but also *Route-Distinguisher (RD)* which is the prefix used for all MAC routes in VNI10000. RD can be thought as a kind of MAC VRF identifier. Spine-11, which has no knowledge about VNIs/VLANs, use RD for differentiating possible overlapping MAC addresses which could be used in different VLANs/VNIs. From the Spine-11 perspective, the Layer2 address of host Beef is 192.168.101:32777:1000.0010.beef.

RD for MAC route is formed from the sender "BGP Router-Id: base value 32767+VLAN-Id". The RD for VLAN 10 MAC addresses in Leaf-101 is 192.168.77.101:32777.
There is also MPLS Label Stack 1 field in NLRI, which includes the L2VNI Identifier. VLAN 10 is mapped to VNI 10000 (= MPLS Label Stack 1: 10000). VNI Id is used in Data Plane in the VXLAN header.

The update message includes two BGP Extended Community Attributes. The first one, the Route-Target attribute is used for route export/import policy by VTEP switches. The second one, Encapsulation type defines the encapsulation used in Data Plane (Type 8 = VXLAN).

Example 9-7 shows how the BGP process of Leaf-101 receives the MAC route exported from the L2RIB. Leaf-101 installs the MAC route information into BGP Loc-RIB with required information related to BGP EVPN Route-Type 2 advertisement (L2VNI Identifier, Route-Target and Encapsulation type). The bit count /112 at the end of the address is the sum of bits for RD (8 octets) + MAC address (6 octets) = 14 octets = 112 bits.

```
Leaf-101# sh bgp internal event-history events | i beef

BRIB: [L2VPN EVPN] Installing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112
(local) via 192.168.100.101 label 10000 (0x0/0x0) into BRIB with extcomm
Extcommunity: RT:65000:10000 ENCAP:8

EVT: Received from L2RIB MAC route: Add ESI 0000.0000.0000.0000.0000 topo 10000
mac 1000.0010.beef flags 0x000002 soo 0 seq 0 reorig: 0
```

Example 9-7: *show bgp l2vpn evpn 1000.0010.beef*

Example 9-8 shows the BGP Adj-RIB-Out entry concerning the NLRI of vmBeef. The address information in BGP entry are explained below:

- Route Distinguisher 192.168.77.101:32777
- [2] - BGP EVPN Route-Type 2, MAC/IP Advertisement Route
- [0] - Ethernet Segment Identifier (ESI), all zeroed out = single homed site
- [0] - Ethernet Tag Id, EVPN routes must use value 0
- [48] - Length of MAC address
- [1000.0010.beef] - MAC address
- [0] - Length of IP address
- [0.0.0.0] - Carried IP address
- /216 - Length of the MAC VRF NLRI in bits: RD (8 octets) + MAC address (6 octets) + L2VNI Id (3 octets) + ESI (10 octets) = 27 octets = 216 bits.

The L2VNI information is shown in the Received Label field. There are also two BGP Extended Community Path Attributes:

Route-Target: 65000:10000 - Used for export/Import policy (Control Plane)
Encapsulation 8: Defines the encapsulation type VXLAN (Data Plane).

```
Leaf-101# show bgp l2vpn evpn 1000.0010.beef
BGP routing table information for VRF default, address family L2VPN
EVPN
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216, version 28
Paths: (1 available, best #1)
Flags: (0x000102) on xmit-list, is not in l2rib/evpn

  Advertised path-id 1
  Path type: local, path is valid, is best path
```

```
  AS-Path: NONE, path locally originated
    192.168.100.101 (metric 0) from 0.0.0.0 (192.168.77.101)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8

  Path-id 1 advertised to peers:
    192.168.77.11
<- Comment: For the simplicity, the MAC-IP entry removed from this
output->
```

Example 9-8: *show bgp l2vpn evpn 1000.0010.beef*

Capture 9-1 shows the BGP EVPN Update message sent by Leaf-101. Note that the Next Hop address and the MPLS Label Stack (L2VNI ID) are only visible in HEX portion of the capture:
Next Hop: HEX c0 a8 64 65 = BIN 192.168.100.101
MPLS Label Stack 1: HEX 00 27 10 = 10000 (L2VNI id)

```
Border Gateway Protocol - UPDATE Message
    Type: UPDATE Message (2)
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Type Code: EXTENDED_COMMUNITIES (16)
            Carried extended communities: (2 communities)
                Route Target: 65000:10000
                    Type: Transitive 2-Octet AS-Specific (0x00)
                    Subtype (AS2): Route Target (0x02)
                    2-Octet AS: 65000
                    4-Octet AN: 10000
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
                    Type: Transitive Opaque (0x03)
                    Subtype (Opaque): Encapsulation (0x0c)
                    Tunnel type: VXLAN Encapsulation (8)
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive, Comp.
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
                EVPN NLRI: MAC Advertisement Route
                  Route Type: MAC Advertisement Route (2)
                  Length: 33
                  Route Distinguisher: (192.168.77.101:32777)
                  ESI: 00 00 00 00 00 00 00 00 00
                        ESI Type: ESI 9 bytes value (0)
                        ESI 9 bytes value: 00 00 00 00 00 00 00 00 00
                  Ethernet Tag ID: 0
                  MAC Address Length: 48
                  MAC Address: Private_10:be:ef (10:00:00:10:be:ef)
                  IP Address Length: 0
                  IP Address: NOT INCLUDED
                  MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)

0000  5e 00 00 01 00 07 5e 00 00 00 00 07 08 00 45 c0   ^.....^.......E.
<snipped>
```

```
0060  40 05 04 00 00 00 64 c0 10 10 00 02 fd e8 00 00   @.....d.........
0070  27 10 03 0c 00 00 00 00 00 08 90 0e 00 2c 00 19   '............,..
0080  46 04 c0 a8 64 65 00 02 21 00 01 c0 a8 4d 65 80   F...de..!....Me.
0090  09 00 00 00 00 00 00 00 00 00 00 00 00 00 00 30   ...............0
00a0  10 00 00 10 be ef 00 00 27 10                     ........'.
```

Capture 9-1: *BGP EVPN Update concerning the MAC address of vmBeef*

Phase 4: BGP AFI L2EVPN MAC Route Import on Remote VTEP

VTEP switch Leaf-102 receives the *MAC Route Advertisement* and installs it into the Adj-RIB-In database without modification. Routes are imported into Loc-RIB based on EVPN import policies (import RT 65000:10000) and BGP Best Path Selection process result. During this import process, the Route-Distinguisher changed from the received RD to local RD if necessary. In this example subnet 192.168.11.0/24 in Leaf-102 has VLAN-Id 10 while in Leaf-102 VLAN-Id for the same subnet is 20. This means that the original RD 192.168.77.32777 for L2VNI10000 generated by Leaf-101 will change by Leaf-102 to 192.168.77.102:32787 when it moves route from Adj-RIB-In into Loc-RIB as L2VNI specific route.

Example 9-9 shows the BGP table of Leaf-102. The upper BRIB entry is the original NLRI (BGP Adj-RIB-In) while the lower BRIB entry shows the NLRI installed into Loc-RIB with local RD.

```
Leaf-102# show bgp l2vpn evpn 1000.0010.beef

BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216, version 277
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 1 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

<MAC-IP part Snipped>
Route Distinguisher: 192.168.77.102:32787    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216, version 278
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path, in rib
             Imported from
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8
```

```
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

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

Example 9-9: *show bgp l2vpn evpn 1000.0010.beef*

Example 9-10 shows the BGP Import process. Leaf-102 receives the BGP EVPN Update. It imports the route into BGP Adj-RIB-In. The route is installed into Loc-RIB with respective Attributes (RD is changed from received to local). From the BGP Loc-RIB route is sent the L2RIB.

```
Leaf-102#  sh bgp internal event-history events | i beef

RIB: [L2VPN EVPN]: Send to L2RIB
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112
RIB: [L2VPN EVPN] For
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112, added
1 next hops, suppress 0
RIB: [L2VPN EVPN] Adding
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112 via
192.168.100.101 to NH list (flags2: 0x0)
RIB: [L2VPN EVPN] Add/delete
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112,
flags=0x200, in_rib: no
IMP: [L2VPN EVPN] Created import destination entry for
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112
IMP: [L2VPN EVPN] Importing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112 to
<default> RD 192.168.77.102:32787
BRIB: [L2VPN EVPN]
(192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112
(192.168.77.11)): returning from bgp_brib_add, reeval=0new_path: 1, change: 1,
undelete: 0, history: 0, force: 0, (pfl
ags=0x40002010) rnh_flag_change 0
BRIB: [L2VPN EVPN]
(192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112
(192.168.77.11)): bgp_brib_add: handling nexthop, path->flags2: 0x80000
BRIB: [L2VPN EVPN] Created new path to
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112 via
192.168.77.111 (pflags=0x40000000, pflags2=0x0)
BRIB: [L2VPN EVPN] Installing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/112
(192.168.77.11) via 192.168.100.101 label 10000 (0x0/0x0) into BRIB with
extcomm Extcommunity: RT:65000:10000 ENCAP:8
```

Example 9-10: *sh bgp internal event-history events | i beef*

Phase 5: MAC VRF on Remote VTEP

From the Loc-RIB, route information is installed into L2RIB of VNI10000 (based on the L2VNI Id carried in MPLS Label Stack 1 field). The source of the information is BGP. Port information points to the remote NVE1 interface IP address of VTEP switch Leaf-101. As shown in the previous example 9-10, the MAC route information is sent from BGP Loc-RIB to L2RIB. Example 9-11 shows the operation of L2FWDER and example 9-12 shows the installation process. Example 9-13 verifies the VLAN to VNI topology mapping. Example 9-14 illustrates the actual content of MAC VRF in L2RIB.

```
Leaf-102# show system internal l2fwder event-history events | i beef
l2fwder_dbg_ev, 690 l2fwder_l2rib_add_remote_entry, 299Add remote mac entry
mac: 1000.0010.beef vni: 20 sw_bd 20 vtep ip: 192.168.100.101

l2fwder_dbg_ev, 690 l2fwder_l2rib_msg_cb, 453MAC address: 1000.0010.beef
```

Example 9-11: *show system internal l2fwder event-history events | i beef*

Example 9-12 shows from top to down how the update process of L2RIB.

```
Leaf-102# sh system internal l2rib event-history mac | i beef
Rcvd MAC ROUTE msg: (20, 1000.0010.beef), vni 0, admin_dist 0, seq 0, soo 0,

(20,1000.0010.beef):Mobility check for new rte from prod: 5
(20,1000.0010.beef):Current non-del-pending route local:no, remote:yes, linked
mac-ip count:1
(20,1000.0010.beef):Mobility type: remote-to-remote:
(20,1000.0010.beef): New route ESI: (F), SOO: 0, Seq num: 0Existing route ESI:
(F), SOO: 0, Seq num: 0 , rt_type: 1
(20,1000.0010.beef,5):Using seq number from Recv-based BGP route
(20,1000.0010.beef,5):Setting Recv flag
(20,1000.0010.beef,5):MAC route modified (rc=0) with seq num:0, flags:
(SplRcv), soo:0, peerid:1, MH<truncated>
(20,1000.0010.beef,5):Encoding MAC route (ADD, client id 0)
(20,1000.0010.beef,5):vni:10000 rt_flags: admin_dist:20, seq_num:0 ecmp_label:0
soo:0(--)
(20,1000.0010.beef,5):res:Regular esi:(F) peerid:1 nve_ifhdl:1224736769
mh_pc_ifidx:0 nh_count:1
(20,1000.0010.beef,5):NH[0]:192.168.100.101
```

Example 9-12: *sh system internal l2rib event-history mac | i beef*

Example 9-13 shows the L2RIB of L2VNI 10000. The next-hop is now the IP address of interface NVE1 of Leaf-101. The source is BGP and the Topology-Id is 20 which refers the VNI10000 (example 9-14).

```
Leaf-102# show l2route evpn mac evi 20

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

Topology    Mac Address    Prod   Flags          Seq No     Next-Hops
----------- -------------- ------ -------------- ---------- --------------
20          1000.0010.beef BGP    SplRcv         0          192.168.100.101
```

Example 9-13: *show l2route evpn mac evi 20*

Example 9-14 shows that the VLAN 20 is attached to L2VNI 10000.

```
Leaf-102# sh vlan id 20 vn-segment

VLAN Segment-id
---- -----------
20   10000
```

Example 9-14: *sh vlan id 20 vn-segment*

Phase 6: MAC Address Table on Remote VTEP

As the last step, the remote VTEP Leaf-102 L2FWDER component installs the MAC reachability information from the L2RIB into the MAC address table. The Next-Hop points to Leaf-101 NVE1 interface.

Now both Leaf-101 and Leaf-102 has up to date information on their databases concerning the reachability information of host vmBeef MAC address and they are able to send frames to vmBeef.

Example 9-15 shows the updated MAC address table of VTEP switch Leaf-102 (nve-peer1).

```
Leaf-102# show system internal l2fwder mac | i beef
<snipped>
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+--------------
*    20     1000.0010.beef     static    -            F      F   (0x47000001) nve-
peer1
```

Example 9-15: *show system internal l2fwder mac | i beef*

L2VNI: Intra-VNI Data Plane

This section explains the L2VNI Data Plane operation. Figure 9-3 shows the example network where hosts Cafe (192.168.11.12/24) and Beef (192.168.11.22/24) are connected to the same L2 broadcast domain, however VLAN 10 is used in Leaf-101 while VLAN 20 is used in Leaf-102. This is not a preferred model but it is used here to illustrates that VLAN-Id is only locally significant. Both VLANs are mapped to L2VNI 10000 in leaf switches. Data Plane operation is explained by using simple ping test from Café to Beef.

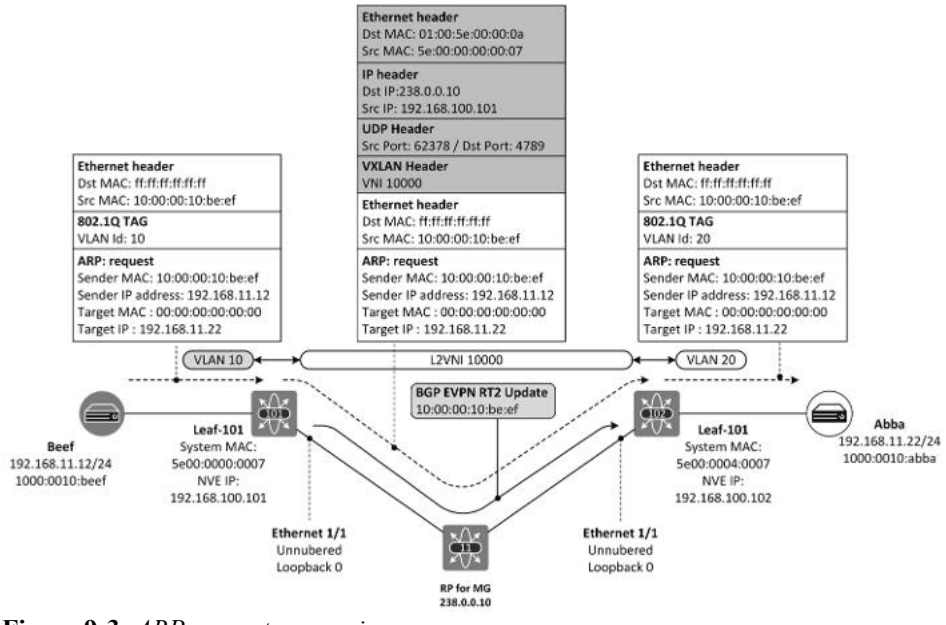


Figure 9-3: *ARP request processing*

ARP Request

Beef starts pinging to Abba. Beef does not yet have the MAC address information of host Abba in its ARP table. Host Beef tries to resolve the MAC address used with IP address 192.168.11.22 by generating an ARP-Request which it sends as L2 Broadcast (figure 9-3).

Leaf-101 receives the frame from its port E1/2. The frame has 802.1Q tag on it with VLAN-Id 10. Based VLAN-to-VNI mapping information, Leaf-101 encapsulates the broadcast frame with VXLAN header where it uses L2VNI10000. In addition, Leaf-101 adds UDP header with fixed destination port 4789 and source port hashed from the payload. Note that the UDP source port is the only changing variable when doing ECMP load balancing between the equal-cost links based on 5-tuple input (destination IP/source IP, Layer 4 Protocol and source port/destination port). The destination IP address is Multicast group address 238.0.0.10 that is used for L2BUM traffic with L2VNI10000. The destination MAC address is derived from the Multicast group address and the source MAC address is the system MAC of Leaf-101 in outer MAC address. Leaf-101 forwards the encapsulated packet out of interface E1/1 towards Rendezvous Point Spine-11 of Multicast Group 238.0.0.10. In addition, Leaf-101 generates BGP EVPN Update for NLRI of host Beef and send it to Spine-11

Spine-11 makes a routing decision based on the IP address of outer IP header. Because the destination IP address is Multicast group address, Spine-11 routes packet based on Multicast-RIB where interface 1/2 is listed in Outgoing Interface List (OIL). Spine-11 also forward the BGP Update message received from Leaf-101 to Leaf-102.

VTEP switch Leaf-102 receives the ARP-request sent by Beef. Leaf-102 removes the headers used for VXLAN tunneling (outer Ethernet header, IP header, UDP header, and VXLAN header). Based on the VNI-to-VLAN mapping database, Leaf-102 knows that it has to switch received Broadcast Ethernet frame out of its interfaces participating in VLAN 20. Leaf-102 adds the 801.Q tag with VLAN-Id 20 into the frame and forwards it out of the Interface E1/2 to Abba. Leaf-102 does not learn the MAC address from the encapsulated frame sent by Leaf-101. Instead of it Leaf-102 learns the MAC address from the BGP Update message.
The Capture 9-2 to 9-4 shows the ARP-request messages.

```
Ethernet II, Src: 10:00:00:10:be:ef, Dst: Broadcast (ff:ff:ff:ff:ff:ff)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 10:00:00:10:be:ef
    Sender IP address: 192.168.11.12
    Target MAC address: 00:00:00_00:00:00
    Target IP address: 192.168.11.22
```

Capture 9-2: *ARP request from vmBeef to vmAbba: Beef to Leaf-101.*

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: IPv4mcast_0a
(01:00:5e:00:00:0a)
    Destination: IPv4mcast_0a (01:00:5e:00:00:0a)
    Source: 5e:00:00:00:00:07 (5e:00:00:00:00:07)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
User Datagram Protocol, Src Port: 62378, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: Broadcast
(ff:ff:ff:ff:ff:ff)
    Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Source: Private_10:be:ef (10:00:00:10:be:ef)
    Type: ARP (0x0806)
    Trailer: 000000000000000000000000000000000000
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: Private_10:be:ef (10:00:00:10:be:ef)
    Sender IP address: 192.168.11.12
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.11.22
```

Capture 9-3: *ARP request from vmBeef to vmAbba: Leaf-101 to Spine-11.*

```
Ethernet II, Src: 10:00:00:10:be:ef, Dst: ff:ff:ff:ff:ff:ff
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 0100 = ID: 20
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: Private_10:be:ef (10:00:00:10:be:ef)
    Sender IP address: 192.168.11.12
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.11.22
```

Capture 9-4: *ARP request from vmBeef to vmAbba: Captured from the link Leaf-102 to vmAbba.*

ARP Reply

Host Abba receives the ARP-request. Based on the “Target IP” information found from the message, it knows that the requested MAC belongs to it. It generates an ARP-reply message that it sends as an L2 Unicast using MAC address of host Beef, which was learned from the received frame (figure 9-4).

Leaf -102 receives the frame. It encapsulates the frame with VXLAN/UDP/outer IP/outer MAC headers. Because Leaf-102 has learned the MAC address of host Beef from BGP Update, it sends the packet as Unicast to Leaf-101.

Leaf-101 removes the VXLAN-encapsulation and forwards the ARP-message to host Beef.

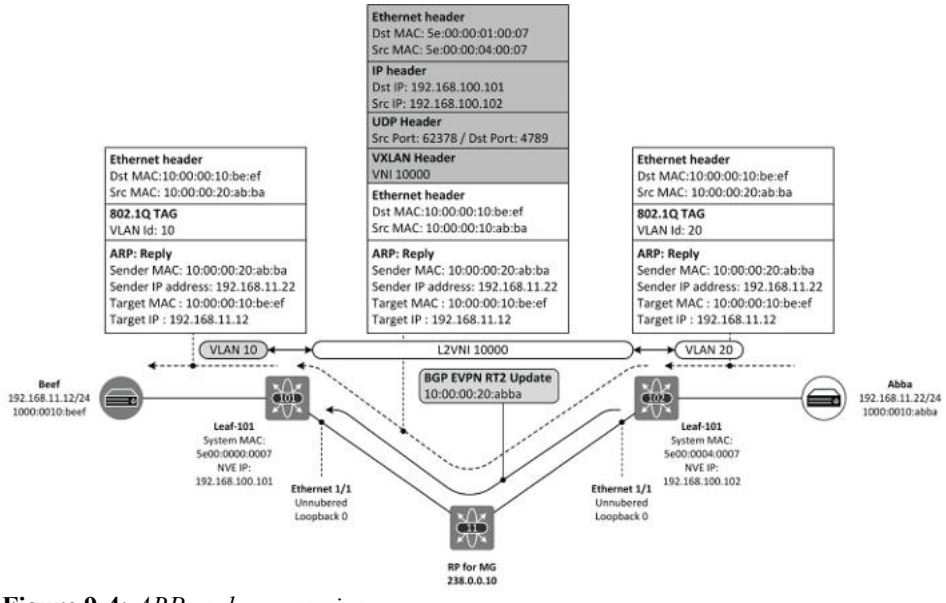


Figure 9-4: *ARP reply processing*

The Capture 9-5 to 9-7 shows the ARP-reply messages.

```
Ethernet II, Src: 10:00:00:20:ab:ba, Dst: 10:00:00:10:be:ef
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 0100 = ID: 20
    Type: ARP (0x0806)
    Padding: 000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 10:00:00:20:ab:ba
    Sender IP address: 192.168.11.22
    Target MAC address: 10:00:00:10:be:ef
    Target IP address: 192.168.11.12
```

Capture 9-5: *ARP reply from vmAbba to vmBeef: Captured from the link between Abba and Leaf-102.*

```
Ethernet II, Src: 5e:00:00:01:00:07, Dst: 5e:00:00:00:00:07
Internet Protocol Version 4, Src: 192.168.100.102, Dst:
192.168.100.101
User Datagram Protocol, Src Port: 59206, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
```

```
Ethernet II, Src: 10:00:00:20:ab:ba, Dst: 10:00:00:10:be:ef
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 10:00:00:20:ab:ba
    Sender IP address: 192.168.11.22
    Target MAC address: 10:00:00:10:be:ef
    Target IP address: 192.168.11.12
```

Capture 9-6: *ARP reply from vmAbba to vmBeef: Leaf-101 to Spine-11.*

```
Ethernet II, Src: 10:00:00:20:ab:ba, Dst: 10:00:00:10:be:ef
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: ARP (0x0806)
    Padding: 000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: Private_20:ab:ba (10:00:00:20:ab:ba)
    Sender IP address: 192.168.11.22
    Target MAC address: Private_10:be:ef (10:00:00:10:be:ef)
    Target IP address: 192.168.11.12
```

Capture 9-7: *ARP reply from vmAbba to vmBeef: Captured from the link Leaf-101 to vmBeef.*

ICMP Request

After resolving the MAC address of Abba, Beef sends an ICMP-request to Beef. It sends the ICMP-request message with the destination IP address 192.168.11.22. The destination MAC address in the Ethernet frame is previously resolved MAC address 1000.0020.abba.

VTEP switch Leaf-101 receives the frame and base on VLAN Id 10 in VLAN tag in 802.1Q header, Leaf-101 notices that the frame belongs to L2VNI 10000. Leaf-101 forwards frame based on the information found from the MAC address table of VLAN 10. MAC address entry information concerning to MAC address of Abba is taken from L2RIB which in turn has received from BGP. Leaf-101 encapsulates the frame inside a new Ethernet header, IP header, UDP header, and VXLAN header and forwards it towards Leaf-102 via Spine-11.

Leaf-102 receives the packet, it removes the outer Ethernet header, outer IP header, UDP header, and VXLAN header and forwards the original frame tagged with 802.1Q tag with VLAN Id 20 to Abba.

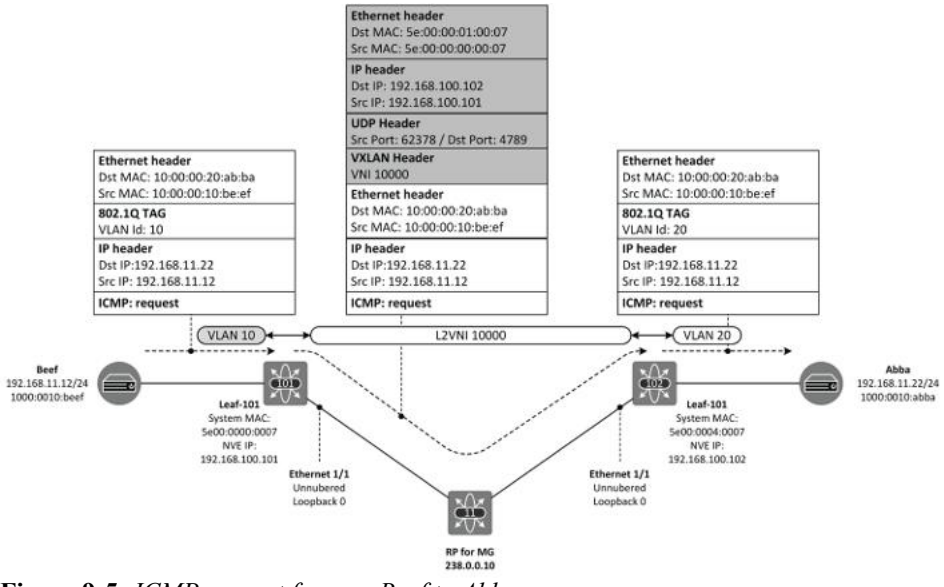


Figure 9-5: *ICMP request from vmBeef to Abba.*

The Capture 9-8 to 9-10 shows the ICMP-Request messages.

```
Ethernet II, Src: 10:00:00:10:be:ef, Dst: 10:00:00:20:ab:ba
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.11.22
Internet Control Message Protocol
```

Capture 9-8: *ICMP request from Beef to Abba: Capture from Leaf-101 to Beef.*

```
Ethernet II, Src: 5e:00:00:00:00:07, Dst: 5e:00:00:01:00:07
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 192.168.100.102
User Datagram Protocol, Src Port: 57986, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: Private_20:ab:ba
(10:00:00:20:ab:ba)
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.11.22
Internet Control Message Protocol
```

Capture 9-9: *ICMP request from Beef to Abba: Capture from Leaf-101 to Spine-11.*

```
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: Private_20:ab:ba
(10:00:00:20:ab:ba)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 0100 = ID: 20
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.11.22
Internet Control Message Protocol
```

Capture 9-10: *ICMP request from Beef to vmAbba: Capture from Leaf-102 to Abba.*

ICMP Reply

When vmAbba receives the ICMP Request, it replies by sending an ICMP-Reply message to vmBeef. The frame processing is the same as what was shown in ARP-Request process.

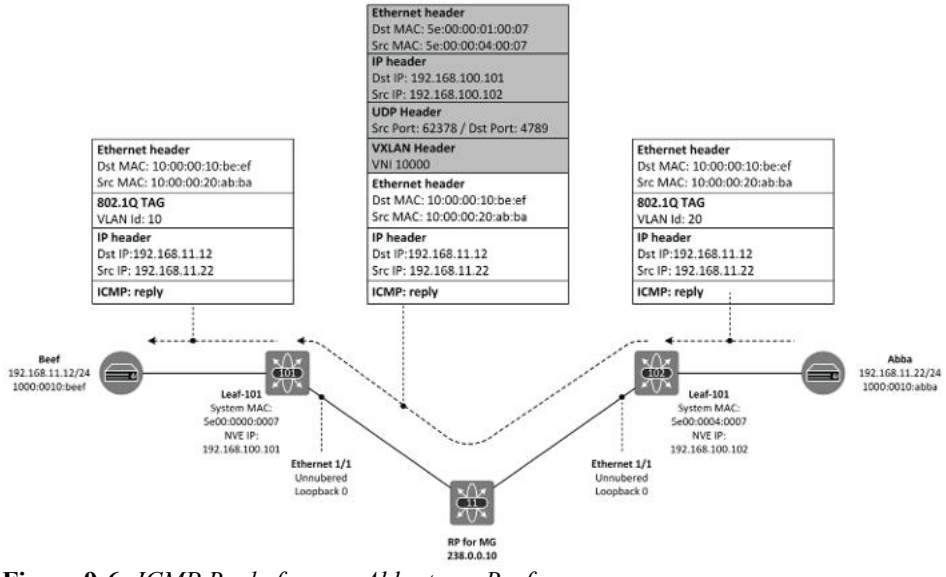


Figure 9-6: *ICMP Reply from vmAbba to vmBeef.*

The Capture 9-11 to 9-13 shows the ARP reply messages.

```
Ethernet II, Src: Private_20:ab:ba (10:00:00:20:ab:ba), Dst: Private_10:be:ef
(10:00:00:10:be:ef)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 0100 = ID: 20
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.11.22, Dst: 192.168.11.12
Internet Control Message Protocol
```

Capture 9-11: *ICMP reply from Abba to Beef: Capture from the link Leaf-102 to Abba.*

```
Ethernet II, Src: 5e:00:00:01:00:07 (5e:00:00:01:00:07), Dst: 5e:00:00:00:00:07
(5e:00:00:00:00:07)
Internet Protocol Version 4, Src: 192.168.100.102, Dst: 192.168.100.101
User Datagram Protocol, Src Port: 57648, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: Private_20:ab:ba (10:00:00:20:ab:ba), Dst: Private_10:be:ef
(10:00:00:10:be:ef)
Internet Protocol Version 4, Src: 192.168.11.22, Dst: 192.168.11.12
Internet Control Message Protocol
```

Capture 9-12: *ICMP reply from Abba to Beef: Capture from the link Leaf-101 to Spine-11.*

```
Ethernet II, Src: Private_20:ab:ba (10:00:00:20:ab:ba), Dst: Private_10:be:ef
(10:00:00:10:be:ef)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.11.22, Dst: 192.168.11.12
Internet Control Message Protocol
```

Capture 9-13: *ICMP request from Abba to Beef: Captured from the link Leaf-101 to Beef.*

Summary

This section shows how the local VTEP switch learns MAC addresses of its connected hosts and how this information is advertised to remote VTEP switches. This chapter also shows the Data Plane operation between the hosts connected to different VTEP switches in the same L2VNI (Layer 2 domain).

MAC-IP address learning process

This section gives a detailed description of how a local VTEP switch learns the IP addresses of its locally connected hosts from the ARP messages generated by the host and how the *Host Mobility Manager (HMM)* component installs the information into the L2RIB of specific VNI. The L2RIB Database holding MAC-IP addresses information is named as IP VRF in figure 9-7. This section also shows how routes are exported from L2RIB into BGP Loc-RIB and advertised via Adj-RIB-Out to the remote VTEP switch by using BGP EVPN Route Type 2 (*MAC Advertisement Route*) advertisement. This section also introduces how the information ends up in L2RIB of remote VTEP. In addition, this section explains how the ARP Suppression mechanism uses MAC-IP binding information to reduce Layer 2 BUM (Broadcast, Unknown Unicast, and Multicast) traffic in VXLAN Fabric. Figure 9-7 illustrates the overall process.

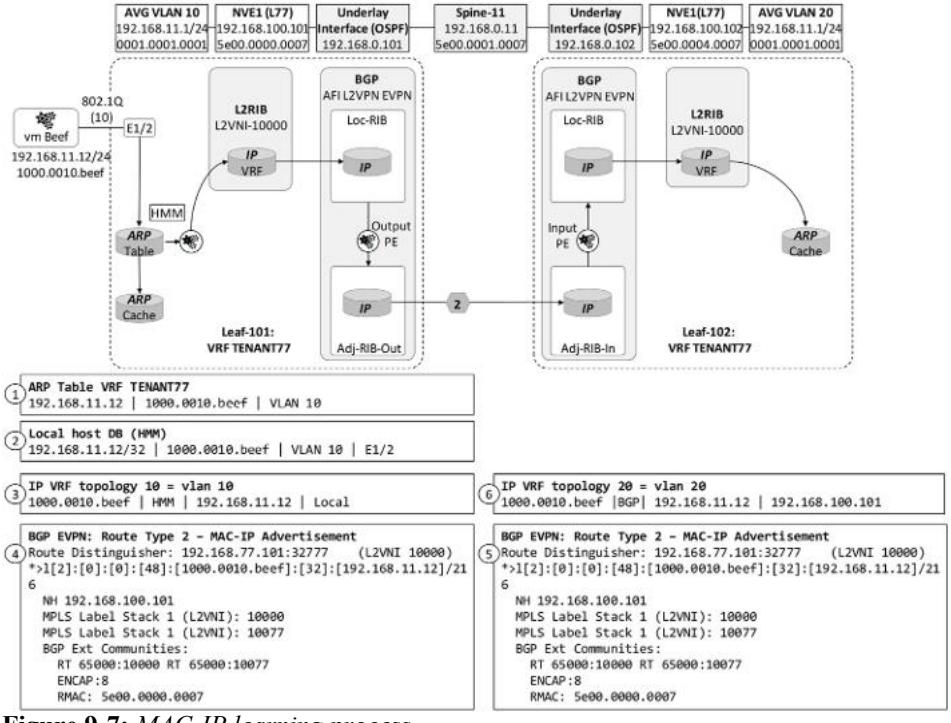


Figure 9-7: *MAC-IP learning process.*

Phase 1: ARP Table on Local VTEP

Virtual Machine Beef comes up. It expresses its' existence to a network and validates the uniqueness of its IP-address by sending a GARP (Gratuitous ARP). VTEP switch Leaf-101 receives the GARP message from interface e1/2 and stores the MAC-IP address binding information from the Sender MAC and the Sender IP fields from the GARP payload into ARP table.

Example 9-16 shows the ARP table of VRF TENANT77. The default aging time for locally learned ARP-entries is in NX-OS is 1500 seconds, which is 300 seconds shorter than MAC-address aging timer. When the ARP aging timers exceed, the switch checks the presence of the host by sending an ARP-request to host. If the host response to ARP-request, the switch will reset the aging timer. If the host does not reply, the entry is removed from the ARP-table but kept in the BGP EVPN table for an additional 1800 seconds (MAC aging timer) before the withdrawn message is sent. The MAC address aging timer should be bigger than the ARP aging timer. This is because the ARP refresh process will also update the MAC table and unnecessary flooding can be avoided.

```
Leaf-101# sh ip arp vrf TENANT77
<snipped>
Address         Age       MAC Address     Interface       Flags
192.168.11.12   00:03:34  1000.0010.beef  Vlan10
```

Example 9-16: *sh ip arp vrf TENANT77*

Phase 2-3: MAC-IP on Local VTEP

The Host Mobility Manager component (HMM) learns the MAC-IP information as a local route. HMM installs the information into the *Local Host Database* and forwards the MAC-IP information into L2RIB. The Local Host Database includes information about the IP address (/32), MAC address, SVI, and local interface. L2RIB has the same information without SVI information.

Example 9-17 shows the partial MAC-IP learning process on Leaf-101.

```
Leaf-101# show system internal l2rib event-history mac-ip

L2RIB MAC-IP Object Event Logs:

Rcvd MAC-IP ROUTE BASE msg: obj_type: 13 oper_type: 1 oper_sbtype: 0 producer:
12

Rcvd MAC-IP ROUTE msg: (10, 1000.0010.beef, 192.168.11.12), l2 vni 0, l3 vni
10077,

(10,1000.0010.beef,192.168.11.12):MAC-IP entry created

(10,1000.0010.beef,192.168.11.12,12):MAC-IP route created with flags 0, l3 vni
10077, seq 0

(10,1000.0010.beef,192.168.11.12,12): admin dist 7, soo 0, peerid 0, peer
ifindex 0

(10,1000.0010.beef,192.168.11.12,12): esi (F), pc-ifindex 0

(10,1000.0010.beef,192.168.11.12,12):Encoding MAC-IP best route (ADD, client id
5), esi: (F)
```

Example 9-17: *show system internal l2rib event-history mac-ip*

Example 9-18 shows the information related to Beef MAC-IP binding in Local Host Database (HMM RIB) of VRF TENANT77.

```
Leaf-101# show fabric forwarding ip local-host-db vrf TENANT77

HMM host IPv4 routing table information for VRF TENANT77
<snipped>
    Host              MAC Address     SVI     Flags     Physical Interface
*   192.168.11.12/32  1000.0010.beef  Vlan10  0x420201   Ethernet1/2
```

Example 9-18: *show fabric forwarding ip local-host-db vrf TENANT77*

Example 9-19 shows that the information concerning the MAC-IP of Beef in IP VRF in L2RIB is produced by the HMM component.

```
Leaf-101# show l2route mac-ip topology 10 detail

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv(D):Del Pending (S):Stale (C):Clear
(Ps):Peer Sync (Ro):Re-Originated
Topology    Mac Address    Prod   Flags  Seq No  Host IP        Next-Hops
----------- -------------- ------ ------ ------- -------------  -----------
10          1000.0010.beef HMM    --     0       192.168.11.12  Local
            L3-Info: 10077
```

Example 9-19: *show fabric forwarding ip local-host-db vrf TENANT77*

Phase 4: BGP Route Export on Local VTEP

VTEP switch Leaf-101 installs the MAC-IP route from the L2RIB into the BGP Loc-RIB. The MAC-IP information is advertised as a separate BGP EVPN Route Type 2 advertisement (dedicated updates for both MAC-only and MAC-IP NLRIs might be used). The difference in carried NLRI information between MAC-Only and MAC-IP route advertisement is that later one has also host IP address and mask information as well as an additional MPLS Label Stack 2 information, that defines the L3VNI used in VRF TENANT77. There are also two additional Extended Communities; RT 65000:10077 and Router MAC 5e00.0000.0007 carried within the update.

Example 9-20 shows the internal process of how VTEP switch Leaf-101 receives the MAC-IP route information and installs it into RIB and BGP Loc-RIB. Note that BGP Extended Community Router MAC information is not shown in the output. The mask length is includes RD (8 octet) + MAC address (6 octet) + IP address (4 octet) = 18 octets = 144 bits. The octet count of the prefix can be seen from the RIB event "Adding Prefix".

```
Leaf-101# sh bgp internal event-history events | i beef
BRIB:
[L2VPN EVPN] Installing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
(local) via 192.168.100.101 label 10000 (0x0/0x0) into BRIB with extcomm
Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Rou

RIB:
[L2VPN EVPN] Adding prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]
Route Length 16 Prefix Length 18:

EVT:
Received from L2RIB MAC-IP route: Add ESI 0000.0000.0000.0000.0000 topo 10000
mac 1000.0010.beef ip 192.168.11.12 L3 VNI 10077 flags 00000000 soo 0 seq 0,
reorig :0
```

Example 9-20: *Leaf-101# sh bgp internal event-history events | i beef*

Example 9-21 shows the BGP Loc-RIB concerning the MAC-IP NLRI of Beef. Prefix information is explained below:

- Route Distinguisher
- [2] - BGP EVPN Route-Type 2, MAC/IP Advertisement Route
- [0] - Ethernet Segment Identifier (ESI), all zeroed out = single homed site
- [0] - Ethernet Tag Id, EVPN routes must use value 0
- [48] - Length of MAC address
- [1000.0010.beef] - MAC address
- [32] - Length of IP address
- [192.168.11.12] - Carried IP address
- /272 - Length of the MAC-IP VRF NLRI in bits: RD (8 octets) + MAC address (6 octets) + L2VNI Id (3 octets) + L3VNI Id (3 octets) + IP address (4 octets) ESI (10 octets) = 34 octets = 272 bits.

The L2VNI information is shown in Received Label field. There are also three BGP Extended Community Path Attributes:

- Route-Target: 65000:10000 - Used for export/Import policy (L2VNI)
- Route-Target: 65000:10077 - Used for export/Import policy (L3VNI)
- Encapsulation 8: Defines the encapsulation type VXLAN (Data Plane)
- Router MAC: 5e00.0000.0007 - Used for Inner MAC Header source address for routed packets. This is needed because VXLAN is MAC in IP/UDP encapsulation tunneling mechanism and data payload over L3 border does not carry source host MAC address information. This is where the RMAC is used.

```
Leaf-101# sh bgp l2vpn evpn 192.168.11.12

BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272, version 5
Paths: (1 available, best #1)
Flags: (0x000102) on xmit-list, is not in l2rib/evpn

  Advertised path-id 1
  Path type: local, path is valid, is best path
  AS-Path: NONE, path locally originated
    192.168.100.101 (metric 0) from 0.0.0.0 (192.168.77.101)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 advertised to peers:
    192.168.77.11
```

Example 9-21: *sh bgp l2vpn evpn 192.168.11.12*

Phase 5: BGP Route Import on Remote VTEP

VTEP switch Leaf-102 receives the BGP EVPN MAC route Advertisement and installs it to the BGP Adj-RIB-In database without any modification. From there, Leaf-102 imports the route to its Loc-RIB where it is installed through the best path selection process to L2RIB. When remote VTEP switch Leaf-102 installs the route from the BGP Adj-RIB into BGP Loc-RIB, it changes the RD to 192.168.77.102:32787 based on its BGP RID and VLAN Id. This process is the same as MAC-Only route import and is based on the same RT 65000:10000.

Example 9-22 shows the internal process, where received MAC-IP route is installed into BGP Adj-RIB-In with RD 192.168.77.101:32777. This route is then imported into BGP Adj-RIB-In Post with RD 192.168.77.102:32787 and installed into Loc-RIB where it is imported into L2RIB. Note that the example includes the installation process of L3RIB too.

```
Leaf-102# sh bgp internal event-history events | i beef

RIB: [L2VPN EVPN]: Send to L2RIB
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144

RIB: [L2VPN EVPN] For
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
, added 1 next hops, suppress 0

RIB: [L2VPN EVPN] Adding
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
via 192.168.100.101 to NH list (flags2: 0x0)

RIB: [L2VPN EVPN] Add/delete
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
, flags=0x200, in_rib: no

IMP: [L2VPN EVPN] Created import destination entry for
192.168.77.102:3:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144

IMP: [L2VPN EVPN] Importing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
to <default> RD 192.168.77.102:3

IMP: [L2VPN EVPN] Created import destination entry for
192.168.77.102:32787:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144

IMP: [L2VPN EVPN] Importing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
to <default> RD 192.168.77.102:32787

IMP: [IPv4 Unicast] Importing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
to <TENANT77> RD 192.168.77.102:3

RIB: [L2VPN EVPN] Add/delete
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
, flags=0x200, evi_ctx invalid, in_rib: no

BRIB: [L2VPN EVPN]
(192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/14
4 (192.168.77.11)): returning from bgp_brib_add, reeval=0new_path: 1, change:
1, undelete: 0, history: 0, force: 0, (pflags=0x40002010) rnh_flag_ch

BRIB: [L2VPN EVPN]
(192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/14
4 (192.168.77.11)): bgp_brib_add: handling nexthop, path->flags2: 0x80000

BRIB: [L2VPN EVPN] Created new path to
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
via 192.168.77.111 (pflags=0x40000000, pflags2=0x0)
BRIB: [L2VPN EVPN] Installing prefix
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/144
(192.168.77.11) via 192.168.100.101 label 10000 (0x0/0x0) into BRIB with
extcomm Extcommunity: RT:65000:10000 RT:65000:10077 ENC
```

Example 9-22: *sh bgp internal event-history events | i beef*

Example 9-23 shows the partial output of BGP-RIB (BRIB) on Leaf-102 (Adj-RIB-In and Loc-RIB). The first highlighted part describes the original, unmodified NLRI received from Spine-11 installed into Adj-RIB-In. The second highlighted part shows the same NLRI installed into Loc-RIB with modified RD value. The import is based on highlighted RT65000:10000. The third highlighted part describes the same NLRI. Though it is installed with RD 192.168.77.102:3 which is used for Inter-VNI traffic (L3VNI). Importing into L3VNI specific Loc-RIB is based on RT65000:10077 auto-generated under VRF Context settings.

```
Leaf-102#sh bgp l2vpn evpn 192.168.11.11 vrf TENANT77
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272, version 4
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 3 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.102:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272, version 5
Paths: (1 available, best #1)
Flags: (0x000212) on xmit-list, is in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path, in rib
             Imported from
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe
]:[32]:[192.168.11.11]/272
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0
007
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.102:3    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272, version 6
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
```

```
            Imported from
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe
]:[32]:[192.168.11.11]/272
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

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

Example 9-23: *sh bgp l2 vpn 192.168.11.11 vrf TENANT77*

Phase 6: IP VRF on Remote VTEP

Remote VTEP Leaf-102 verifies the reachability of Next-Hop IP address found from NLRI and since it is a hit, the L2FWDER component installs the MAC-IP route into L2RIB. Local topology ID is now 20 (based on VLAN 20) and the source of the information is BGP. Next-Hop Port information points to the NVE1 interface IP address of VTEP switch Leaf-101.

At this phase both VTEP switches have information of MAC-IP of vmBeef in their L2RIB as well as in BGP tables but only local VTEP switch Leaf-101 has installed the MAC-IP binding information into ARP table.

Example 9-24 shows the partial MAC-IP learning process.

```
Leaf-102# sh system internal l2rib event-history mac-ip

L2RIB MAC-IP Object Event Logs:

Rcvd MAC-IP ROUTE BASE msg: obj_type:13 oper_type:1 oper_sbtype: 0 producer: 5
Rcvd MAC-IP ROUTE msg:(20, 1000.0010.beef, 192.168.11.12), l2 vni 0, l3 vni 0,
Rcvd MAC-IP ROUTE msg: flags , admin_dist 0, seq 0, soo 0, peerid 0,
Rcvd MAC-IP ROUTE msg: res 0, esi (F), ifindex 0, nh_count 1, pc-ifindex 0
NH: 192.168.100.101
(20,1000.0010.beef,192.168.11.12):MAC-IP entry created
(20,1000.0010.beef,192.168.11.12,5):MAC-IP route created with flags 0, l3 vni
0, seq 0
(20,1000.0010.beef,192.168.11.12,5): admin dist 20, soo 0, peerid 0, peer
ifindex 0
(20,1000.0010.beef,192.168.11.12,5): esi (F), pc-ifindex 0
```

Example 9-24: *sh system internal l2rib event-history mac-ip*

Example 9-25 shows the MAC-IP information in L2RIB is produced by BGP.

```
Leaf-102# show l2route mac-ip topology 20 detail

<snipped>
Topology  Mac Address    Prod   Flags Seq No Host IP        Next-Hops
--------- -------------- ------ ----- -----  ------        ------------
20        1000.0010.beef BGP    --      0   192.168.11.12  192.168.100.101
```

Example 9-25: *sh system internal l2rib event-history mac-ip*

At this phase, both VTEP switches have the MAC-IP address information of vmBeef.

ARP-Suppression

The previous section explains how the MAC-IP address information is propagated in BGP EVPN VXLAN fabric. This section describes how the VTEP switches use MAC-IP binding information to reduce the unnecessary Broadcast traffic in VXLAN fabric. This section starts from the phase where vmBeef comes up and sends GARP/ARP message to the network. Leaf-101 installs the MAC-IP binding information into the ARP table of VRF TENANT77. Example 9-26 shows the ARP table and figure 9-8 illustrates the overall process.

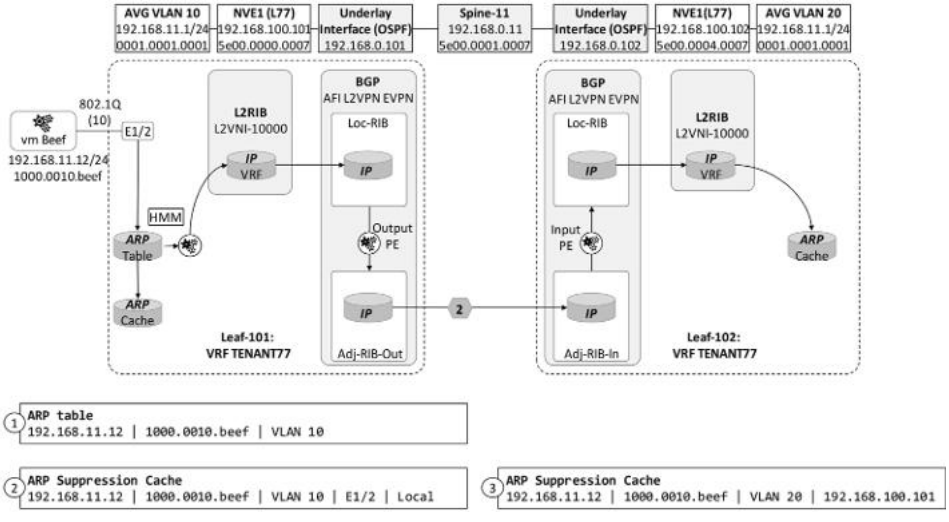


Figure 9-8: *MAC-IP information in ARP table and ARP Suppress Cache.*

```
Leaf-101# sh ip arp vrf TENANT77 | b Address

Address         Age       MAC Address     Interface       Flags
192.168.11.12   00:02:01  1000.0010.beef  Vlan10
```

Example 9-26: *sh system internal l2rib event-history mac-ip*

When VNI based ARP-Suppression is enabled on local VTEP switches, the MAC-IP address binding information is also installed into local ARP Suppression Cache from the ARP table. (Example 9-27).

```
Leaf-101# sh ip arp suppression-cache detail
<snipped>
Ip Address    Age  Mac Address  Vlan Physical-ifindex  Flags Remote Vtep Addrs

192.168.11.12   00:03:06 1000.0010.beef   10 Ethernet1/2    L
```

Example 9-27: *sh ip arp suppression-cache detail*

When ARP-suppression is enabled on remote VTEP switches, the ARP Suppression Cache information is taken from the L2RIB. Example 9-28 illustrates this from Leaf-102 perspective.

```
Leaf-102# show ip arp suppression-cache detail
<snipped>
Ip Address    Age    Mac Address Vlan Physical-ifindex Flags Remote Vtep Addrs
192.168.11.12    00:03:33 1000.0010.beef    20 (null)  R     192.168.100.101
```

Example 9-28: *show ip arp suppression-cache detail*

Figure 9-9 illustrates the ARP operation with and without ARP suppression as well as with Unknown Unicast Suppression.

No Suppression: All ARP-Requests are flooded towards the Mcast group defined for specific VNI and all VTEP switches joined to that group receives the ARP Request message and forwards it out of the ports participating in Broadcast domain defined by VNI Id in VXLAN header.

ARP Suppression: The Local VTEP switch checks if the requested MAC-IP binding information is stored into local ARP Suppression Cache. If the check is hit, switch sends an ARP reply back to the requester without flooding the actual ARP request to the network. If the ARP Suppression Cache check is a miss, then the ARP request is flooded to the network. ARP suppression should be enabled only after initial Intra-VNI reachability testing.

ARP and Unknown Unicast Suppression: Works the same way as ARP-Suppression in case that ARP Suppression check is hit but in case of a miss, the ARP Request is dropped. This option requires that there is no silent host in the VXLAN Fabric.

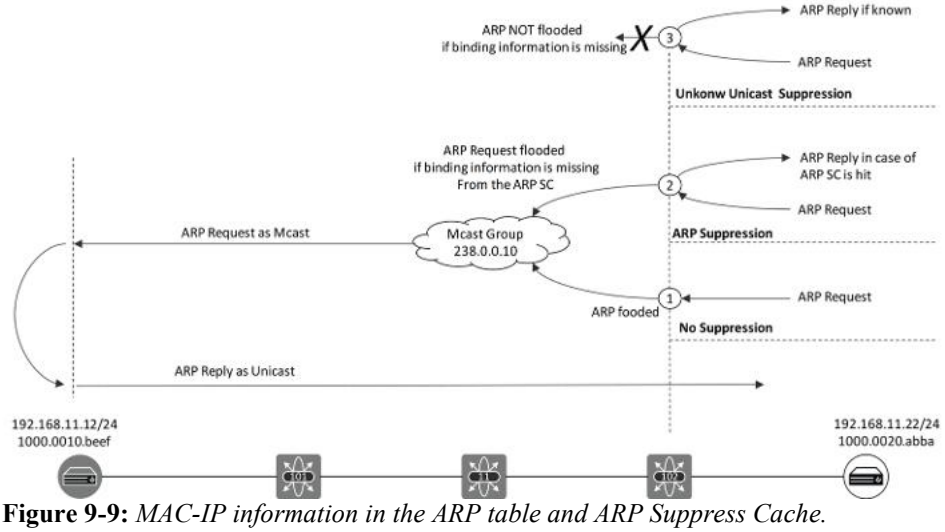


Figure 9-9: *MAC-IP information in the ARP table and ARP Suppress Cache.*

At this phase, the network is able to work as a transparent Layer 2 switch for hosts participating in L2VNI 10000 and switch frames between the hosts connected to it.

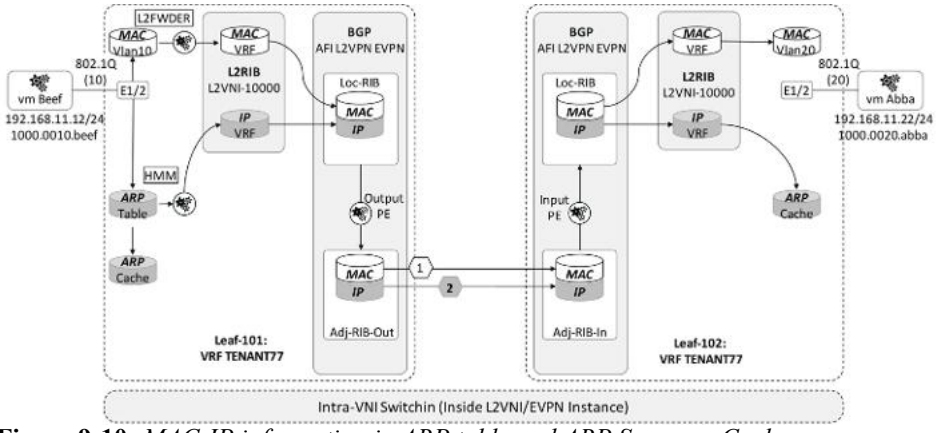


Figure 9-10: *MAC-IP information in ARP table and ARP Suppress Cache.*

Host route Advertisement: Inter-VNI routing (L3VNI)

First two sections explain how the MAC and MAC-IP information of hosts are propagated over the VXLAN Fabric and how the information is used for Intra-VNI switching and MAC address resolution as well as reducing BUM traffic. This section explains how host routes are imported into L3RIB and how this information is used for Inter-VNI routing. In addition, this section explains the mechanism of how MAC address information of silent hosts is resolved by using prefix route advertisement. Figure 9-11 describes the overall process.

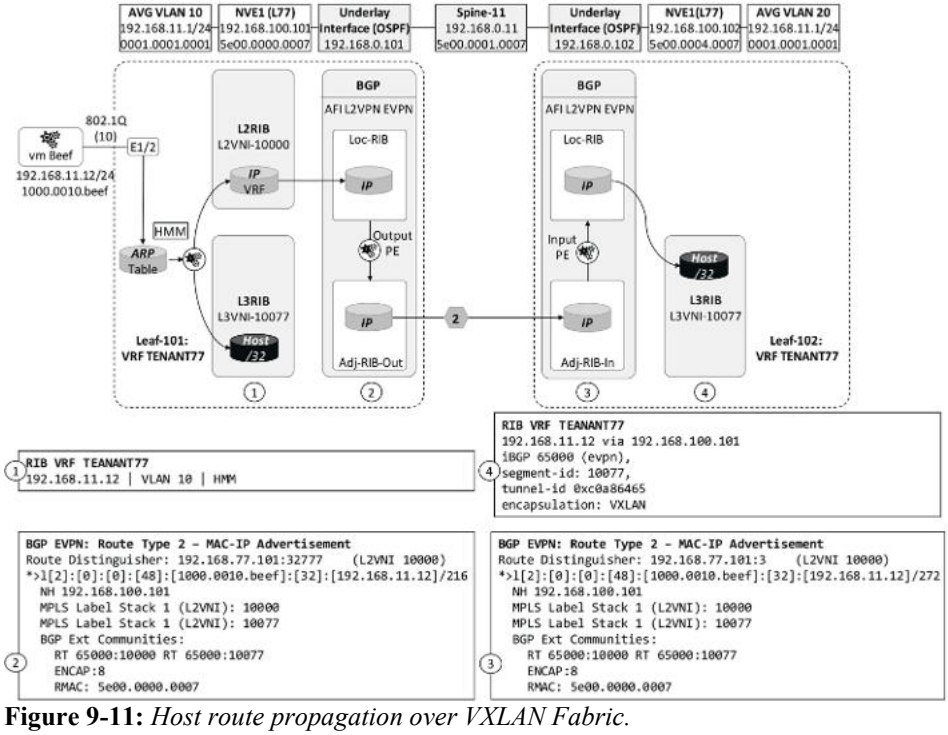


Figure 9-11: *Host route propagation over VXLAN Fabric.*

Phase 1. Host Route in Local Routing Information Base (RIB)

Section “MAC-IP Learning Process” describes how the local VTEP switch installs the MAC-IP address binding information into the ARP table and how the HMM component installs the information into L2RIB. In addition to this process, the HMM component installs the MAC-IP information from the ARP-Table into L3RIB.

Example 9-29 shows the RIB of VRF TENANT77 in local VTEP switch Leaf-101. The route is learned from VLAN 10 and it is installed into an RIB by HMM.

```
Leaf-101# show ip route 192.168.11.12 vrf TENANT77

IP Route Table for VRF "TENANT77"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.11.12/32, ubest/mbest: 1/0, attached
    *via 192.168.11.12, Vlan10, [190/0], 03:34:14, hmm
```

Example 9-29: *show ip route 192.168.11.12 vrf TENANT77*

Phase 2. Host Route BGP Process on Local VTEP

Section “MAC-IP Learning Process” also covers the process of how the MAC-IP information is sent from the L2RIB to Loc-RIB and from there send to Adj-RIB-Out where it is advertised as a BGP EVPN Route type 2 Update to remote VTEP switches.

Example 9-30 shows the BGP Loc-RIB concerning the IP address of vmBeef. This same output has been earlier explained in detail in example 9-21.

```
Leaf-101# sh bgp l2vpn evpn 192.168.11.12

BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272, version 16
Paths: (1 available, best #1)
Flags: (0x000102) on xmit-list, is not in l2rib/evpn

  Advertised path-id 1
  Path type: local, path is valid, is best path
  AS-Path: NONE, path locally originated
    192.168.100.101 (metric 0) from 0.0.0.0 (192.168.77.101)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007

  Path-id 1 advertised to peers:
    192.168.77.11
```

Example 9-30: *sh bgp l2vpn evpn 192.168.11.12*

Phase 3. Host Route BGP Process on Remote VTEP

The section "MAC-IP Learning Process" did not explain how the MAC-IP routing information ends up into L3RIB of Remote VTEP switch. BGP EVPN Route type 2 Update concerning the MAC-IP NLRI of Beef includes also Route Target 65000:10077 (L3VNI). The received NLRI information is sent through the Import Policy Engine (import is based on RT 65000:10077) and the Decision process into Loc-RIB as an L3VNI entry. During the Input Policy processing, the original RD 192.168.77.101:32777 is changes to VRF TENANT77 specific RD 192.168.77.102:3 (3 = VRF Id of VRF TENANT77). RD is used for the differentiated overlapping IP address in different VRFs. Example 9-31 shows the BGP table of Leaf-102. The first highlighted portion is the original, received NLRI in Adj-RIB-In. The second highlighted portion is the same update imported into Loc-RIB. The import is based on the RT 65000:10077 defined under VRF Context. The RD is changed from 192.168.77.101:32777 to 192.168.77.102:3. Example 9-34 shows the VRF Id of VRF TENANT77.

```
Leaf-102# show bgp l2vpn evpn 192.168.11.12
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272, version 22
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 3 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

<L2VNI snipped for simplicity>

Route Distinguisher: 192.168.77.102:3    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272, version 24
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported from
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

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

Example 9-31: *show bgp l2vpn evpn 192.168.11.12*

```
Leaf-102# show vrf TENANT77

VRF-Name                                VRF-ID State   Reason
TENANT77                                     3 Up      --
```

Example 9-32: *show vrf TENANT77*

Phase 4. Installing Host Route into RIB of Remote VTEP

The route is installed into L3 RIB from the BGP Loc-RIB. The RIB entry includes information about Next-Hop address and tunnel-id, encapsulation type (VXLAN), segment Id and route source (BGP). At this phase, both local VTEP switches Leaf-101 and remote VTEP switch Leaf-102 are capable to route Inter-VNI traffic to Beef (belonging to L2VNI 10000) from the hosts participating in different L2VNI.

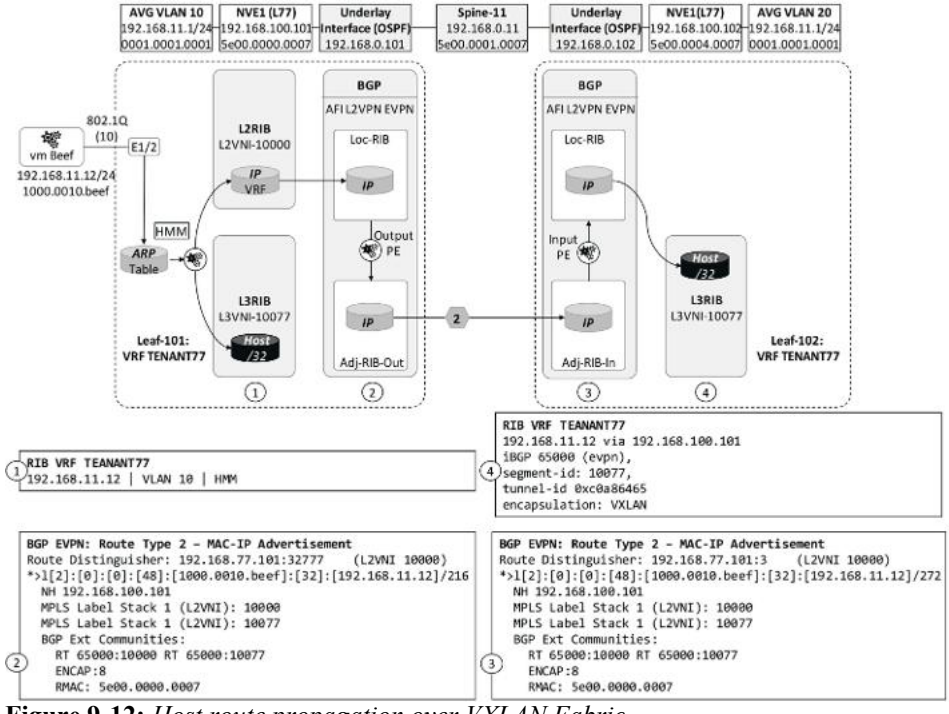


Figure 9-12: *Host route propagation over VXLAN Fabric.*

Example 9-33 shows the VRF TENANT77 RIB entry concerning the host route 192.168.11.12/32

```
Leaf-102# show ip route 192.168.11.12 vrf TENANT77

IP Route Table for VRF "TENANT77"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
```

```
192.168.11.12/32, ubest/mbest: 1/0
    *via 192.168.100.101%default, [200/0], 04:20:01, bgp-65000, internal, tag
65000 (evpn) segid: 10077 tunnelid: 0xc0a86465 encap: VXLAN
```

Example 9-33: *show vrf TENANT77*

Example 9-34 shows the BGP Recursive Next Hop Database and that the 192.168.100.101 is used as a next-hop for destination 192.168.11.12.

```
Leaf-102# show nve internal bgp rnh database vni 10077

------------------------------------------
Total peer-vni msgs recvd from bgp: 10
Peer add requests: 6
Peer update requests: 0
Peer delete requests: 4
Peer add/update requests: 6
Peer add ignored (peer exists): 0
Peer update ignored (invalid opc): 0
Peer delete ignored (invalid opc): 0
Peer add/update ignored (malloc error): 0
Peer add/update ignored (vni not cp): 0
Peer delete ignored (vni not cp): 0
------------------------------------------
Showing BGP RNH Database, size : 2 vni 10077

Flag codes: 0 - ISSU Done/ISSU N/A        1 - ADD_ISSU_PENDING
            2 - DEL_ISSU_PENDING          3 - UPD_ISSU_PENDING


VNI     Peer-IP            Peer-MAC           Tunnel-ID  Encap     (A/S)  Flags
10077   192.168.100.101    5e00.0000.0007     0xc0a86465 vxlan     (1/0)    0
```

Example 9-34: *show nve internal bgp rnh database vni 10077*

Data Plane operation

Figure 9-13 shows the Data Plane operation when Bebe in VLAN 30 attached to L2VNI 30000 sends an ICMP-Request to Beef in VLAN 10 attached L2VNI 10000. Figure 9-13 illustrates the overall process.

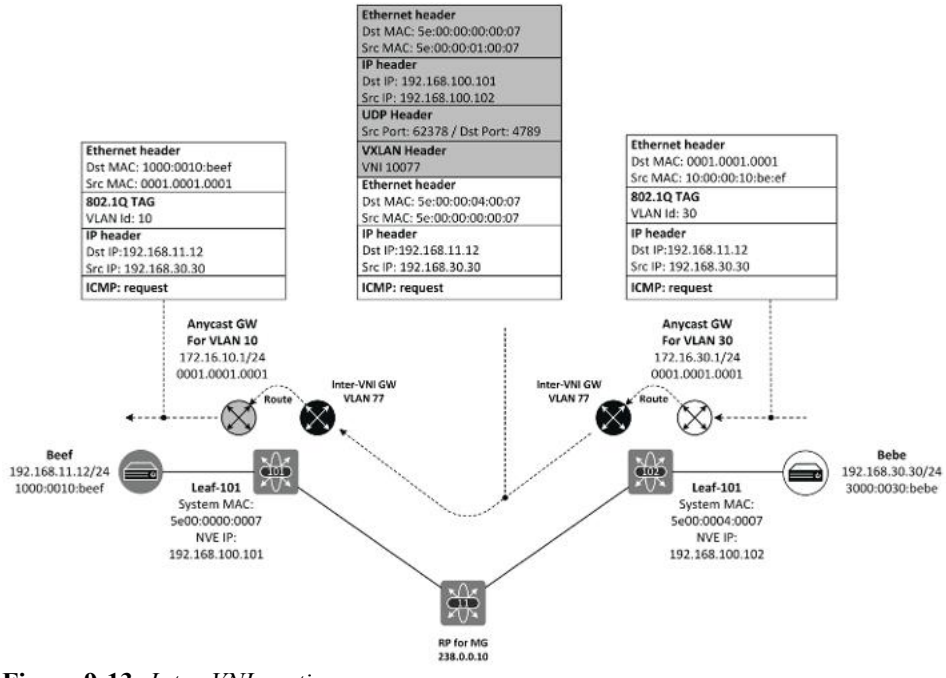


Figure 9-13: *Inter-VNI routing process.*

Phase 1. Switching in VNI30000 on VTEP-102

Because the destination IP address is in a different subnet, Bebe sends an ICMP request message to its default gateway Leaf-102 using Anycast Gateway MAC (AGM) 0001.0001.0001 as a destination MAC address.

Phase 2. Routing from VNI30000 to VNI 10077 on VTEP-102

Local VTEP switch Leaf-102 receives the frame. The destination IP address 192.168.11.12 (host Beef) is learned via BGP and installed into RIB with Next Hop IP address 192.168.100.101 (Leaf-101) and additional information used in Data Plane, such as L3VNI and Encapsulation type. Leaf-102 makes the recursive routing lookup for next-hop address, encapsulates the original packet with VXLAN header with VN Id 10077 (L3VNI), and routes packet towards Leaf-101 via Spine-11 (outer destination MAC belongs to Spine-11). Because VXLAN is a MAC in IP/UDP tunneling mechanism, there has to be the inner source and destination MAC address. The inner source MAC address is taken from the SVI used in Inter-VNI routing, in our case SVI VLAN 77. The inner destination address is RMAC received via BGP Update as BGP Extended Community.

Phase 3. Routing from VNI10077 to VNI 10000 on VTEP-101

When the VTEP switch Leaf-101 receives the VXLAN encapsulated packet, it removes the outer headers used in VXLAN tunneling. Since the VNI 10077 is attached to VRF TENANT77, the routing decision is based on RIB of VRF TENANT77. Leaf-101 routes the original ICMP request to VLAN 10 and switched out of the interface E1/2 with an additional 802.1Q Tag with VLAN Id 10.

This process describes the Symmetric Integrated Route and Bridge (IRB) model where the packet is first switched by the local VTEP, which then routed over VXLAN fabric by using common L3VNI in VXLAN header. The receiving VTEP switch removes VXLAN encapsulation and makes the routing decision based on the target IP address of the original IP packet. After the routing decision, the packet is switched to the destination (bridge-route-route-bridge). The return traffic follows the same model.

Using symmetric IBR gives design flexibility since unlike in Asymmetric IRB, there is no need for adding all VNIs to all VTEP switches. Asymmetric IRB is based on a bridge-route-bridge model where there is no dedicated VNI for Inter-VNI routing. As an example: If we are using Asymmetric IRB in our VXLAN fabric, the host Bebe sends the packet to its default gateway (switched), just like in the case of symmetric IRB. Local VTEP switch Leaf-102 makes routing decision but instead of using common L3VNI, it uses the VNI10000 in VXLAN header, which is attached to VLAN 20 (Local VLAN for VNI 10000). This is the “routed” part. Receiving VTEP switch Leaf-101 removes the VXLAN header and based on the VLAN 10000 it switches the packet out of VLAN 10 (locally attached to VLAN 10).

Capture 9-14 is taken from the link between Spine-11 and Leaf-101 while pinging from Bebe to Beef.

```
Ethernet II, Src: 5e:00:00:04:00:07 (5e:00:00:04:00:07), Dst: 5e:00:00:01:00:07
(5e:00:00:01:00:07)
Internet Protocol Version 4, Src: 192.168.100.102, Dst: 192.168.100.101
User Datagram Protocol, Src Port: 63384, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10077
    Reserved: 0
Ethernet II, Src: 5e:00:00:04:00:07 (5e:00:00:04:00:07), Dst: 5e:00:00:00:00:07
(5e:00:00:00:00:07)
Internet Protocol Version 4, Src: 192.168.30.30, Dst: 192.168.11.12
Internet Control Message Protocol
```

Capture 9-14 *ICMP request captured from the link between the Leaf-101 and Spine-11.*

Summary

This section explains how the IP address of hosts are propagated across the VXLAN fabric and how those are installed into L3RIB.

Prefix Advertisement

Prefix advertisement is a simple process but why it is needed if all VTEP switches know MAC addresses and IP addresses of all connected hosts? One reason is the connectivity with VXLAN Fabric external networks. The other reason is related to the connectivity inside VXLAN Fabric where there might be silent hosts, which does not generate any traffic without request. In some cases, this might lead to a situation where hosts in one L2VNI does not have connectivity to silent host in other L2VNI.

The first example shows the processes when Beef in VNI 10000 connected to Leaf-101 pings the silent host Bebe in VNI 30000 connected to Leaf-102. In this example, both VTEP switches have VNI 30000. IP prefix redistribution in this example is not needed. Figure 9-14 describes the phases 1-3.

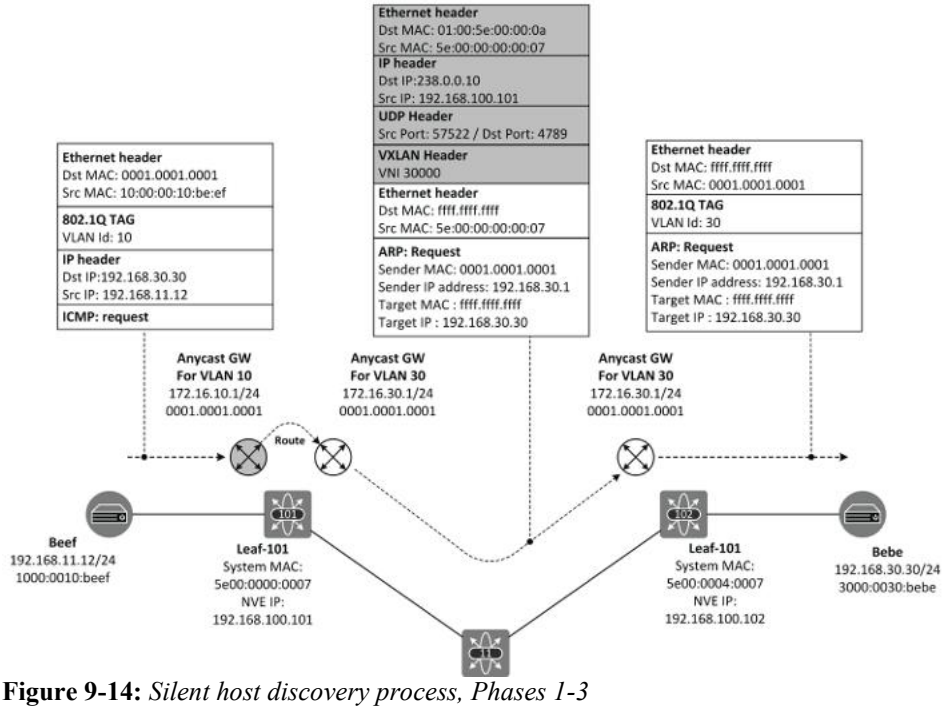


Figure 9-14: *Silent host discovery process, Phases 1-3*

Phase 1: vmBeef start pinging to vmBebe

At this stage, Beef has resolved the MAC address of its default gateway. It sends an ICMP request towards 192.168.30.30. Since the destination Bebe is in a different subnet than host Beef, it sends the ICMP request to the default gateway. There is no response to the first ICMP request.

```
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: EquipTra_01:00:01
(00:01:00:01:00:01)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.30.30
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x574b [correct]
    [Checksum Status: Good]
    Identifier (BE): 0 (0x0000)
    Identifier (LE): 0 (0x0000)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
    [No response seen]
    Data (72 bytes
```

Capture 9-15: *ICMP request captured from the link between the Leaf-101 vmBeef.*

Phase 2: Local VTEP Leaf-101: ARP process

VTEP switch Leaf-101 has both VNI 10000 and 30000 configured locally. Even though there is no host route to vmBebe in the RIB, there is a routing entry for the local subnet 192.168.30.0/24 (VLAN 30 attached to VNI 30000) and the packet is routed from VNI 10000 to VNI 30000. After routing, Leaf-101 tries to figure out the MAC-IP binding information and it sends an ARP-Request to Multicast group used in VNI 30000. Example 9-35 shows the routing table of Leaf-101 and Capture 9-13 shows the ARP request message capture taken from the link between Leaf-101 and Spine-11.

```
Leaf-101# show ip route vrf TENANT77

IP Route Table for VRF "TENANT77"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.11.0/24, ubest/mbest: 1/0, attached
    *via 192.168.11.1, Vlan10, [0/0], 01:09:38, direct, tag 77
192.168.11.1/32, ubest/mbest: 1/0, attached
    *via 192.168.11.1, Vlan10, [0/0], 01:09:38, local, tag 77
192.168.11.22/32, ubest/mbest: 1/0
    *via 192.168.100.102%default, [200/0], 00:45:30, bgp-65000, internal, tag
65
000 (evpn) segid: 10077 tunnelid: 0xc0a86466 encap: VXLAN

192.168.30.0/24, ubest/mbest: 1/0, attached
    *via 192.168.30.1, Vlan30, [0/0], 00:02:36, direct
192.168.30.1/32, ubest/mbest: 1/0, attached
    *via 192.168.30.1, Vlan30, [0/0], 00:02:36, local
```

Example 9-35: *show ip route vrf TENANT77*

Because this is a switched packet inside L2VNI 30000 the source MAC address of the inner Ethernet header is an Anycast Gateway MAC (AGM) address of VLAN 30, which used commonly in every host SVI. By using AGM, hosts do not have to resolve the MAC address of the gateway when moving from one VTEP to another. The destination MAC address is derived from the Multicast Group IP address.

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: IPv4mcast_0a
(01:00:5e:00:00:0a)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
User Datagram Protocol, Src Port: 57522, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 30000
    Reserved: 0
Ethernet II, Src: EquipTra_01:00:01 (00:01:00:01:00:01), Dst: Broadcast
(ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: EquipTra_01:00:01 (00:01:00:01:00:01)
    Sender IP address: 192.168.30.1
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.30.30
```

Capture 9-16: *ICMP request captured from the link between the Leaf-101 and Spine-11.*

Phase 3: Remote VTEP Leaf-102: ARP process - Request

The remote VTEP switch Leaf-102 receives the ARP request. Based on the VNI 30000 in VXLAN header it knows that this packet belongs to VLAN 30. It removes the VXLAN encapsulation and forwards the ARP request out of all interfaces participating in VLAN 30. Leaf-102 insert 802.1Q tag with VLAN id 30 to frame and sent it out of interface E1/2.

```
Ethernet II, Src: 00:01:00:01:00:01, Dst: (ff:ff:ff:ff:ff:ff)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 30
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 1110 = ID: 30
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: EquipTra_01:00:01 (00:01:00:01:00:01)
    Sender IP address: 192.168.30.1
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.30.30
```

Capture 9-17 *ARP request send to vmBebe*

Phase 4: vmBebe: ARP process - Reply

The ARP request reaches the host Bebe and since the ARP-request target IP belongs to it, Bebe response by sending an ARP reply. The source MAC address in received ARP request is AGM, which is also used by Leaf-102. When Bebe send the ARP reply as a Unicast message by using MAC 0001.0001.0001 (AGW) as a destination, the message stops to Leaf-102. This means that Leaf-102 never forwards the ARP response message Leaf-101.

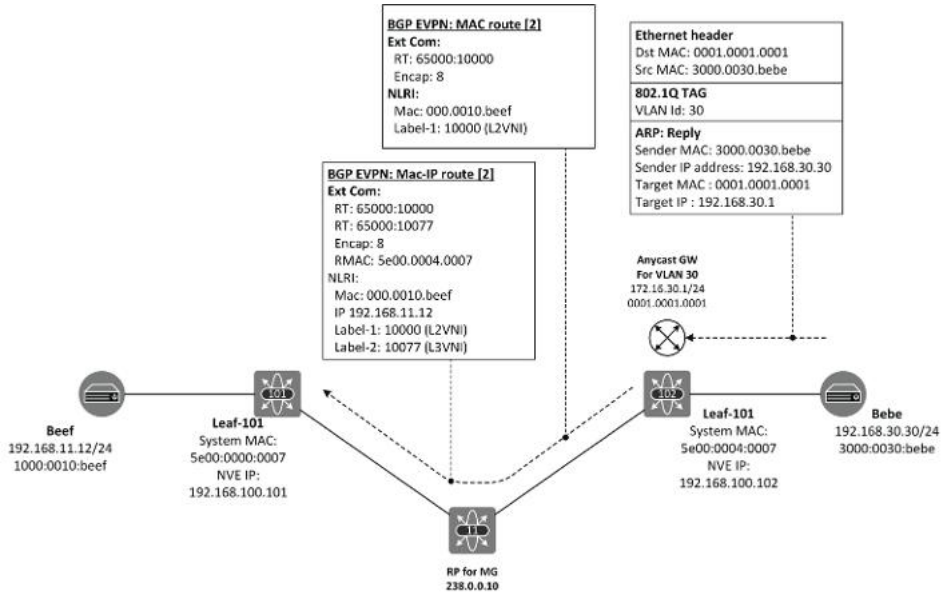


Figure 9-15: *Silent host discovery process, Phases 4-6*

```
Ethernet II, Src: 30:00:00:30:be:be (30:00:00:30:be:be), Dst: EquipTra_01:00:01
(00:01:00:01:00:01)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 30
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 1110 = ID: 30
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 30:00:00:30:be:be (30:00:00:30:be:be)
    Sender IP address: 192.168.30.30
    Target MAC address: EquipTra_01:00:01 (00:01:00:01:00:01)
    Target IP address: 192.168.30.1
```

Capture 9-18 *ARP request send to vmBebe*

Phase 5: remote VTEP switch Leaf-102: BGP Update

When the remote VTEP switch Leaf-102 receives the ARP reply, it learns the MAC-IP information of Bebe from the ARP payload and generates two BGP EVPN route type 2 MAC advertisement route, where the other carries MAC address and the other one MAC-IP address information of Bebe.

```
Ethernet II, Src: 5e:00:00:01:00:07 (5e:00:00:01:00:07), Dst: 5e:00:00:00:00:07
(5e:00:00:00:00:07)
Internet Protocol Version 4, Src: 192.168.77.11, Dst: 192.168.77.101
Transmission Control Protocol, Src Port: 179, Dst Port: 54583, Seq: 1, Ack:
232, Len: 141
Border Gateway Protocol - UPDATE Message
<snipped>
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 32
            Carried extended communities: (4 communities)
                Route Target: 65000:10077
                Route Target: 65000:30000
                Encapsulation: VXLAN Encapsulation
                Unknown subtype 0x03: 0x5e00 0x0004 0x0007
        Path Attribute - ORIGINATOR_ID: 192.168.77.102
        Path Attribute - CLUSTER_LIST: 192.168.77.111
        Path Attribute - MP_REACH_NLRI
            Type Code: MP_REACH_NLRI (14)
            Length: 51
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (42 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 40
                    Route Distinguisher: 192.168.77.102:32797
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: 30:00:00:30:be:be
                    IP Address Length: 32
                    IPv4 address: 192.168.30.30
                    MPLS Label Stack 1: 1875, (BOGUS: Bottom of Stack NOT set!)
                    MPLS Label Stack 2: 629 (bottom)
```

Capture 9-19 *ARP request send to vmBebe*

Phase 6: Local VTEP switch Leaf-102: BGP Update

Local VTEP switch Leaf-101 receives the BGP EVPN Updates and installs the routing information into L2RIB of VNI 30000. This is explained in the section “MAC/IP address learning process”. Right after the L2RIB updates, Leaf-101 is able to route packet sent by Beef to Bebe even though the original ARP-Request was never answered.

```
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: EquipTra_01:00:01
(00:01:00:01:00:01)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.30.30
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x574b [correct]
    [Checksum Status: Good]
    Identifier (BE): 0 (0x0000)
    Identifier (LE): 0 (0x0000)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
    [No response seen]
    Data (72 bytes
```

Capture 9-20: *ICMP request from the link between the Leaf-101 and Spine-11.*

What if all VNIs are not implemented in each VTEP switch? In the scenario where the VTEP switch Leaf-101 has only VNI 10000, it does not have any L2/L3 address information about silent host Bebe, which means that Leaf-101 is not able to switch or route the packet to any hosts in network 192.168.30.0/24. The resolution for this is a prefix advertisement in Leaf-102. At the starting point, Leaf-102 redistributes the local network 192.168.30.0/24 from RIB into BGP via route-map. The update is sent as a BGP EVPN route-type 5 (*Prefix advertisement route*). Example 9-36 shows the BGP RIB of Leaf-101concerning the NLRI for 192.168.30.0/24. BGP EVPN route-type 5 update carries only RT 65000:10077. Received Label field defines the L3VNI. The original RD carried in NLRI is generated based on BGP RID and VRF Id. Figure 9-16 illustrates route-type 5 advertisement process.

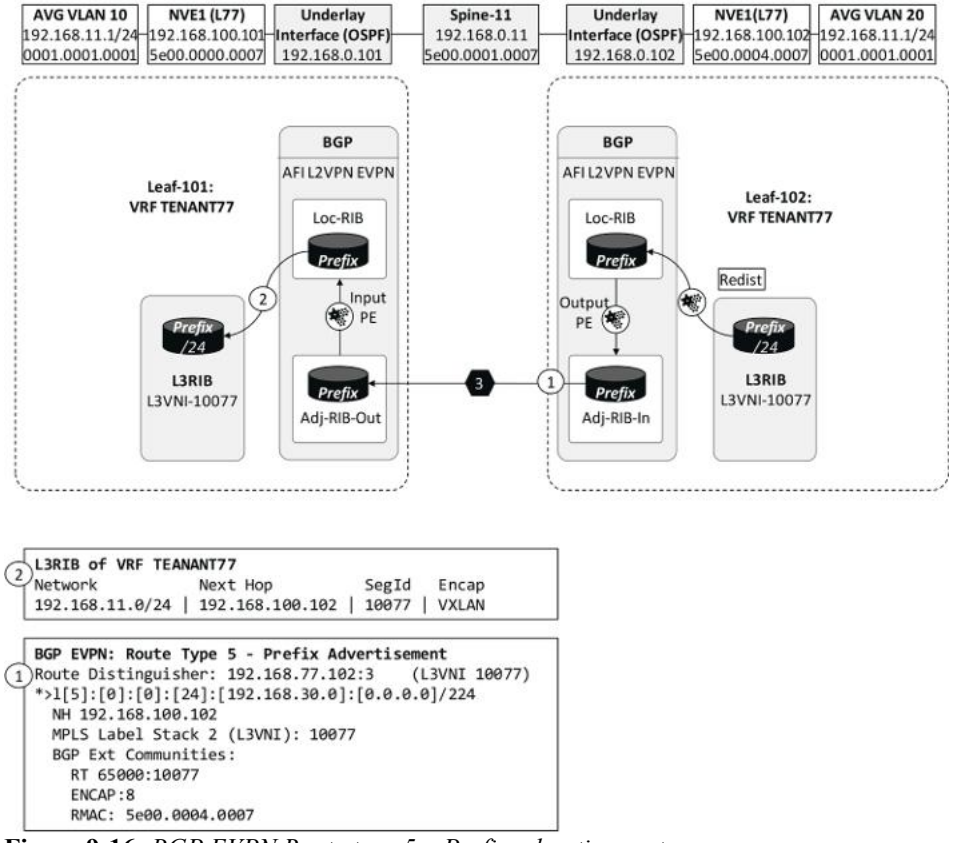


Figure 9-16: *BGP EVPN Route type 5 – Prefix advertisement.*

```
Leaf-101# show bgp l2vpn evpn 192.168.30.0

BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.102:3
BGP routing table entry for [5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224,
version 505
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 2 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin incomplete, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0004.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
```

```
BGP routing table entry for [5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224,
version 506
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported from
192.168.77.102:3:[5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin incomplete, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0004.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

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

Example 9-36: *show bgp l2vpn evpn 192.168.30.0*

Capture 9-21 shows the BGP EVPN Prefix Advertisement (route type 5). Note that Extended Community Unknown Subtype 0x03 defines the RMAC.

```
Ethernet II, Src: 5e:00:00:01:00:07, Dst: 5e:00:00:00:00:07
Internet Protocol Version 4, Src: 192.168.77.11, Dst: 192.168.77.101
Transmission Control Protocol, Src Port: 179, Dst Port: 54583, Seq: 294, Ack:
246, Len: 134
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 134
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 111
    Path attributes
        Path Attribute - ORIGIN: INCOMPLETE
        Path Attribute - AS_PATH: empty
        Path Attribute - MULTI_EXIT_DISC: 0 0
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 24
            Carried extended communities: (3 communities)
                Route Target: 65000:10077
                Encapsulation: VXLAN
                Unknown subtype 0x03: 0x5e00 0x0004 0x0007
        Path Attribute - ORIGINATOR_ID: 192.168.77.102
        Path Attribute - CLUSTER_LIST: 192.168.77.111
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive, Complete
            Type Code: MP_REACH_NLRI (14)
            Length: 45
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (36 bytes)
                EVPN NLRI: IP Prefix route
                    Route Type: IP Prefix route (5)
                    Length: 34
                    Route Distinguisher: 192.168.77.102:3
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
```

```
                    IP prefix length: 24
                    IPv4 address: 192.168.30.0
                    IPv4 Gateway address: 0.0.0.0
                    MPLS Label Stack: 629 (bottom)
```

Capture 9-21: *ICMP request captured from the link between the Leaf-101 and Spine-11.*

Leaf-101 verifies the reachability of Next Hop reported in MP_NLRI_REACH. Leaf-101 has an entry for reported NH in its BGP RNH DB and it installs route into RIB from the BGP Loc-RIB (example 9-37). BGP RNH output was introduced in earlier example 9-34.

```
Leaf-101# show ip route 192.168.30.0 vrf TENANT77

IP Route Table for VRF "TENANT77"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.30.0/24, ubest/mbest: 1/0
    *via 192.168.100.102%default, [200/0], 00:10:27, bgp-65000, internal, tag
65000 (evpn) segid: 10077 tunnelid: 0xc0a86466 encap: VXLAN
```

Example 9-37: *show ip route 192.168.30.0 vrf TENANT77*

Data Plane testing

Data Plane is tested by pinging from host Beef to host Bebe. Figure 9-17 illustrates the first four phases.

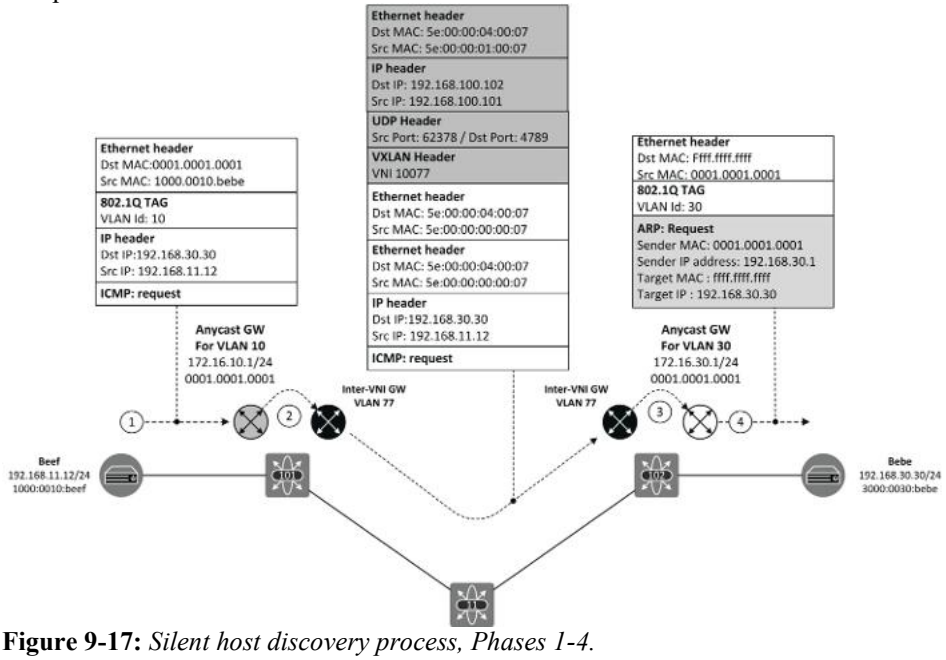


Figure 9-17: *Silent host discovery process, Phases 1-4.*

Phase 1: vmBeef start pinging to vmBebe

At this stage, Beef has resolved the MAC address of its default gateway. It sends an ICMP request to 192.168.30.30. Since the destination is in a different subnet than vmBeef, it sends the packet to its default gateway.

```
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: EquipTra_01:00:01
(00:01:00:01:00:01)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.30.30
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    <snipped>
```

Capture 9-22: *ICMP request sent by vmBeef: capture from the link vmBeef-Leaf-101.*

Phase 2: Local VTEP Leaf-101: Routing

VTEP switch Leaf-101 receives the ICMP packet from Beef with the destination IP address 192.168.30.30. In the previous example, Leaf-101 has both VNI 10000 (subnet 192.168.11.0/24) and VNI 30000 (192.168.30.0/24) implemented in. That is why Leaf-101 started the address resolution process by sending ARP to Mcast Group specific to VNI 30000. In this scenario, there is no VNI 30000 implemented in Leaf-101. Instead of the ARP process, Leaf-101 now routes the packet based on the longest match 192.168.30.0/24 found in its RIB. It routes packet towards the next-hop address 192.168.100.102 (Leaf-102). The real next hop is resolved through the recursive route lookup. Leaf-101 encapsulates the ICMP request with VXLAN header with L3VNI Id 10077. Capture 9-23 shows VXLAN encapsulated packet taken from the link between Leaf-101 and Spine-11.

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: 5e:00:00:01:00:07
(5e:00:00:01:00:07)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 192.168.100.102
User Datagram Protocol, Src Port: 58173, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10077
    Reserved: 0
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: 5e:00:00:04:00:07
(5e:00:00:04:00:07)
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.30.30
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x2861 [correct]
    [Checksum Status: Good]
    Identifier (BE): 5 (0x0005)
    Identifier (LE): 1280 (0x0500)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
    [No response seen]
    Data (72 bytes)
```

Capture 9-23: *ICMP request from the link between the Leaf-101 and Spine-11.*

Phase 3-4: Remote VTEP Leaf-102: ARP request

Remote VTEP switch Leaf-102 receives the ICMP request. Based on VNI 10077 in the VXLAN header, it knows that this packet belongs to VRF TENANT and has to be routed based on its RIB. It removes the VXLAN header and does routing lookup. The packet is routed based on the longest prefix match 192.168.30.0/24 (local VLAN 30). Because Leaf-102 does not have MAC-IP binding information for IP 192.168.30.30, it proceeds with ARP request that is sent out to VLAN 30 (attached to network 192.168.30.0/24). Capture 9-24 is taken from trunk link between Leaf-102 and Bebe.

```
Ethernet II, Src: EquipTra_01:00:01 (00:01:00:01:00:01), Dst: Broadcast
(ff:ff:ff:ff:ff:ff)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 30
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 1110 = ID: 30
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: EquipTra_01:00:01 (00:01:00:01:00:01)
    Sender IP address: 192.168.30.1
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.30.30
```

Capture 9-24: *ARP request captured from the trunk link vmBebe and Leaf-101.*

Phase 5: vmBebe: ARP Reply

Host Bebe receives the ARP request and responds to it by sending ARP reply message as a unicast to VTEP switch Leaf-102. Figure 9-16 shows the data Plane operation phases 5-11.

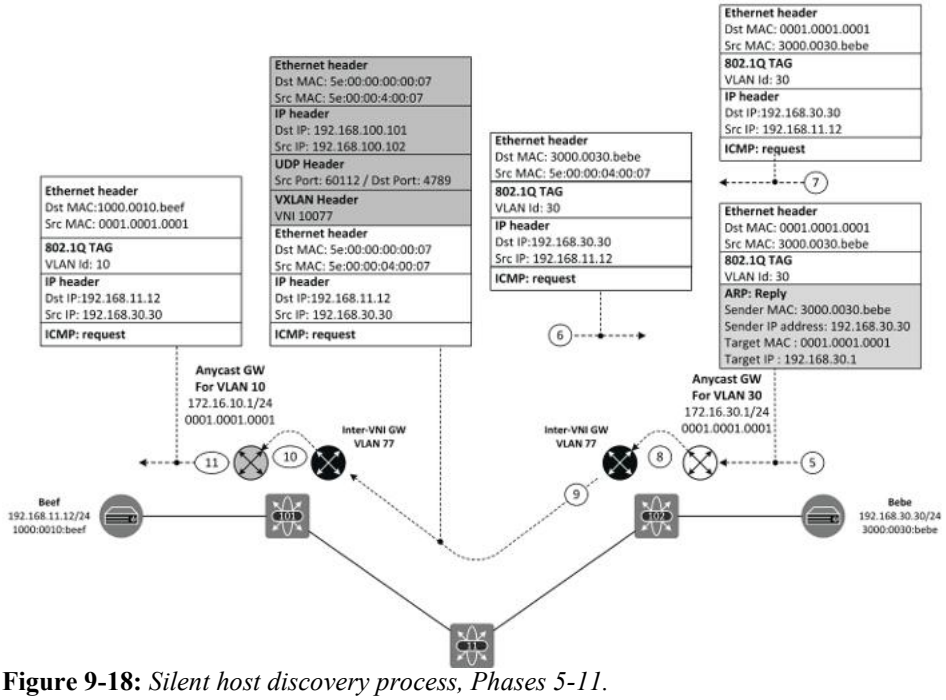


Figure 9-18: *Silent host discovery process, Phases 5-11.*

```
Ethernet II, Src: 30:00:00:30:be:be (30:00:00:30:be:be), Dst: EquipTra_01:00:01
(00:01:00:01:00:01)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 30
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 1110 = ID: 30
    Type: ARP (0x0806)
    Padding: 000000000000000000000000000000
    Trailer: 00000000
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 30:00:00:30:be:be (30:00:00:30:be:be)
    Sender IP address: 192.168.30.30
    Target MAC address: EquipTra_01:00:01 (00:01:00:01:00:01)
    Target IP address: 192.168.30.1
```

Capture 9-25: *ARP reply captured from the link vmBebe and Leaf-101.*

Phase 6: Remote VTEP Leaf-102: ICMP Request forwarding

Now Leaf-102 is able to forward the ICMP request to Bebe

```
Ethernet II, Src: 5e:00:00:04:00:07 (5e:00:00:04:00:07), Dst: 30:00:00:30:be:be
(30:00:00:30:be:be)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 30
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.30.30
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
```

Capture 9-26: *ICMP request from the link between the Leaf-101 and Spine-11.*

Phase 7: vmBebe: ICMP reply

Bebe receives the ICMP Request and sends an ICMP reply back to vmBeef.

```
Ethernet II, Src: 30:00:00:30:be:be (30:00:00:30:be:be), Dst: EquipTra_01:00:01
(00:01:00:01:00:01)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 30
Internet Protocol Version 4, Src: 192.168.30.30, Dst: 192.168.11.12
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
```

Capture 9-27: *ICMP request from the link between the Leaf-101 and Spine-11.*

Phase 8-9: Remote VTEP Leaf-102: Routing decision and ICMP reply

The ICMP reply is sent to Leaf-101 by Leaf-102 over VNI 10077.

```
Ethernet II, Src: 5e:00:00:01:00:07 (5e:00:00:01:00:07), Dst: 5e:00:00:00:00:07
(5e:00:00:00:00:07)
Internet Protocol Version 4, Src: 192.168.100.102, Dst: 192.168.100.101
User Datagram Protocol, Src Port: 60112, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10077
    Reserved: 0
Ethernet II, Src: 5e:00:00:04:00:07 (5e:00:00:04:00:07), Dst: 5e:00:00:00:00:07
(5e:00:00:00:00:07)
Internet Protocol Version 4, Src: 192.168.30.30, Dst: 192.168.11.12
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
```

Capture 9-28: *ICMP Reply captured from the link between the Leaf-101 and Spine-11.*

Phase 10-11: Local VTEP Leaf-101: Routing decision and ICMP reply

VTEP switch Leaf-101 receives the ICMP reply packet. It removes the VXLAN encapsulation. Based on VNI 10077 it knows that the packet belongs to VRF TENANT77 and route lookup has to be dome based on VRF TENANT77 RIB. The destination IP address 192.168.11.12 belongs to VLAN 10. Leaf-101 has the MAC-IP binding information for 192.168.11.12, so it switches the packet out of the interface e1/2.

```
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: Private_10:be:ef
(10:00:00:10:be:ef)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 10
Internet Protocol Version 4, Src: 192.168.30.30, Dst: 192.168.11.12
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
```

Capture 9-29: *ICMP request from the link between the Leaf-101 and Spine-11.*

Just like in the previous example where Leaf-101 has both VNIs 10000 and 30000 implemented locally, we are using the Symmetric IRB model in this scenario. The packet is switched in local VLAN 10, and then it is routed over the VXLAN Fabric with VNI 10077 (L3VNI). In remote VTEP switch Leaf-102, the packet is first routed based on RIB of VRF TENANT77 and then switched in local VLAN 30.

During the process, Leaf-102 learns the MAC-IP information of Bebe. This information is advertised to VTEP switch Leaf-101 which in turn installs the routing information in its BGP RIB.

Example 9-38 shows the BRIB. Entries concerning host route 192.168.30.30/32 and subnet 192.168.30.0/24 with RD 192.168.77.101:3 are routes that are actually imported into BGP Loc-RIB of Leaf-101.

```
Leaf-101# sh bgp l2vpn evpn

BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 56, Local Router ID is 192.168.77.101
<snipped>
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup

   Network            Next Hop            Metric   LocPrf  Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>l[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
                      192.168.100.101                   100   32768 i
*>i[2]:[0]:[0]:[48]:[1000.0020.abba]:[0]:[0.0.0.0]/216
                      192.168.100.102                   100       0 i
*>l[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272
                      192.168.100.101                   100   32768 i

Route Distinguisher: 192.168.77.102:3
*>i[5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224
                      192.168.100.102          0        100       0 ?

Route Distinguisher: 192.168.77.102:32787
*>i[2]:[0]:[0]:[48]:[1000.0020.abba]:[0]:[0.0.0.0]/216
```

```
                     192.168.100.102                    100       0 i

Route Distinguisher: 192.168.77.102:32797
*>i[2]:[0]:[0]:[48]:[3000.0030.bebe]:[0]:[0.0.0.0]/216
                     192.168.100.102                    100       0 i
*>i[2]:[0]:[0]:[48]:[3000.0030.bebe]:[32]:[192.168.30.30]/272
                     192.168.100.102                    100       0 i

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
*>i[2]:[0]:[0]:[48]:[3000.0030.bebe]:[32]:[192.168.30.30]/272
                     192.168.100.102                    100       0 i
*>i[5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224
                     192.168.100.102          0         100       0 ?
```

Example 9-38: *sh bgp l2vpn evpn*

Example 9-39 shows that host route 192.168.30.30 is installed from the BGP Adj-RIB-In to Loc-RIB based on RT 65000:10077. During the process, the Input Policy engine changes the RD 192.168.77.102:32797 (L2VNI) to 192.168.77.101:3 (3 = VRF Id of VRF TENANT77).

```
Leaf-101# sh bgp l2vpn evpn 192.168.30.30
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.102:32797
BGP routing table entry for
[2]:[0]:[0]:[48]:[3000.0030.bebe]:[32]:[192.168.30.30]/272, version 65
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 2 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 30000 10077
      Extcommunity: RT:65000:10077 RT:65000:30000 ENCAP:8 Router
MAC:5e00.0004.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[3000.0030.bebe]:[32]:[192.168.30.30]/272, version 46
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported from
192.168.77.102:32797:[2]:[0]:[0]:[48]:[3000.0030.bebe]:[32]:[192.168.30.30]/272
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 30000 10077
      Extcommunity: RT:65000:10077 RT:65000:30000 ENCAP:8 Router
MAC:5e00.0004.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

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

Example 9-39: *sh bgp l2vpn evpn 192.168.30.30*

Also, the BGP EVPN route type 5 (Prefix Route) is installed from the BGP Adj-RIB-In into Loc-RIB.

```
Leaf-101# sh bgp l2vpn evpn 192.168.30.0
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.102:3
BGP routing table entry for [5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224,
version 63
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 2 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin incomplete, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0004.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
BGP routing table entry for [5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224,
version 5
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported from
192.168.77.102:3:[5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin incomplete, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0004.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

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

Example 9-40: *sh bgp l2vpn evpn 192.168.30.0*

Example 9-41 shows that both host route 192.168.30.30/32 and prefix route 192.168.30.0/24 are installed from the BGP Loc-RIB into VRF TENANT77 specific L3RIB.

```
Leaf-101# show ip route vrf TENANT77

IP Route Table for VRF "TENANT77"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.11.0/24, ubest/mbest: 1/0, attached
    *via 192.168.11.1, Vlan10, [0/0], 01:05:03, direct, tag 77
192.168.11.1/32, ubest/mbest: 1/0, attached
    *via 192.168.11.1, Vlan10, [0/0], 01:05:03, local, tag 77
192.168.11.12/32, ubest/mbest: 1/0, attached
    *via 192.168.11.12, Vlan10, [190/0], 00:17:15, hmm
192.168.30.0/24, ubest/mbest: 1/0
    *via 192.168.100.102%default, [200/0], 01:02:54, bgp-65000, internal, tag
65000 (evpn) segid: 10077 tunnelid: 0xc0a86466 encap: VXLAN

192.168.30.30/32, ubest/mbest: 1/0
    *via 192.168.100.102%default, [200/0], 00:17:10, bgp-65000, internal, tag
65000 (evpn) segid: 10077 tunnelid: 0xc0a86466 encap: VXLAN
```

Example 9-41: *show ip route vrf TENANT77*

Summary

This chapter describes the BGP EVPN Control and Data Plane Layer 2 (switching) and Layer 3 (Routing) operation. It also explains the various components used in BGP EVPN VXLAN Fabric (such as L2RIB, MAC table, MAC VRF, IP VRF, L3RIB, ARP table, ARP Suppression Cache, BGP Adj-RIB-IN, Loc-RIB, Adj-RIB-Out) as well as interoperability between the different components.

References

[RFC 4271] Y. Rekhter et al., “A Border Gateway Protocol 4 (BGP-4)”, RFC 4271, January 2006.

[RFC 4760] T. Bates et al., “Multiprotocol Extensions for BGP-4”, RFC 4760, January 2007,

[RFC 7432] A. Sajassi et al., “BGP MPLS-Based Ethernet VPN” RFC 7432, February 2015.

[EVPN-INT-S] A. Sajassi et al., “draft-ietf-bess-evpn-inter-subnet-forwarding-08 - Integrated Routing and Bridging in EVPN”, March 2019.

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective: ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Chapter 10: VXLAN fabric External Connections

This chapter explains how to connect an external network to VXLAN fabric. The external connection can be terminated either to the spine or leaf switches. The preferred model is border leaf connection because spine switches already host both Multicast Rendezvous Point and BGP Route Reflector services. Border services can be implemented also into Spine switches without any performance issue, but then the spine switches become VTEP switches, which means that those will do a VXLAN encapsulation and decapsulation. In addition, scaling out the spine layer by adding a new switch, also affects external connections.

This chapter uses a full-mesh BGP model instead of a U-shaped model for a couple of reasons. First, it is the most resilient option, there will be no blackholing in event of one link failure. Second, there is no need for iBGP peering between Border Leaf switches.

Figure 10-1 shows the example topology and IP addressing scheme.

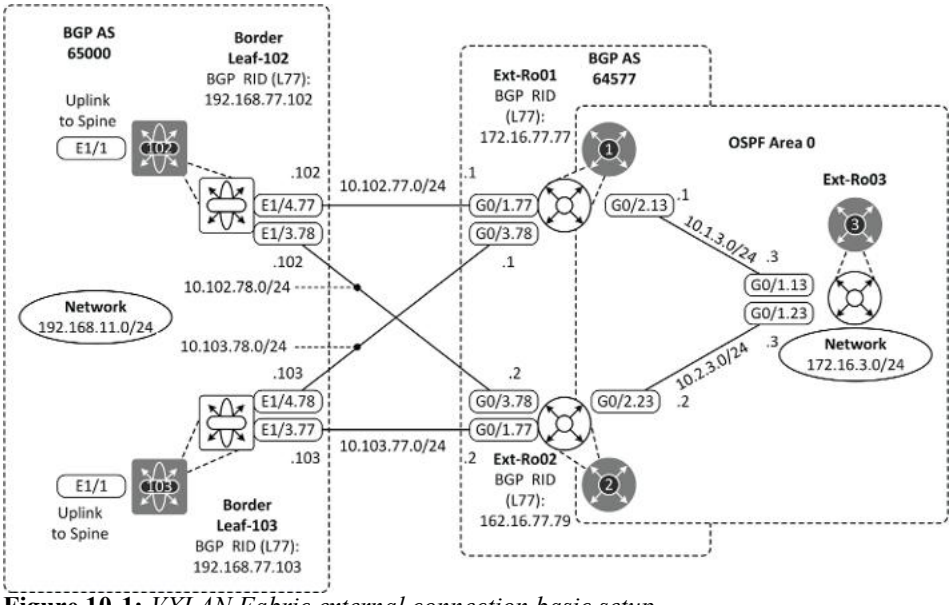


Figure 10-1: *VXLAN Fabric external connection basic setup.*

eBGP Configuration between Border Leaf-102 and Ext-Ro01

This section introduces a simple, single-homed topology before dual-homed full-mesh external BGP peering solution.

Figure 10-2 shows the IP addressing and logical structure of the example lab. There are a sub-interface e1/4.77 in Border Leaf-102 and interface g0/1.77 in Ext-Ro01, both of these interfaces belong to the vrf TENANT77. An eBGP peering is established between these two interfaces. VXLAN Fabric belongs to BGP AS65000 and Ext-Ro01 belongs to BGP AS64577.

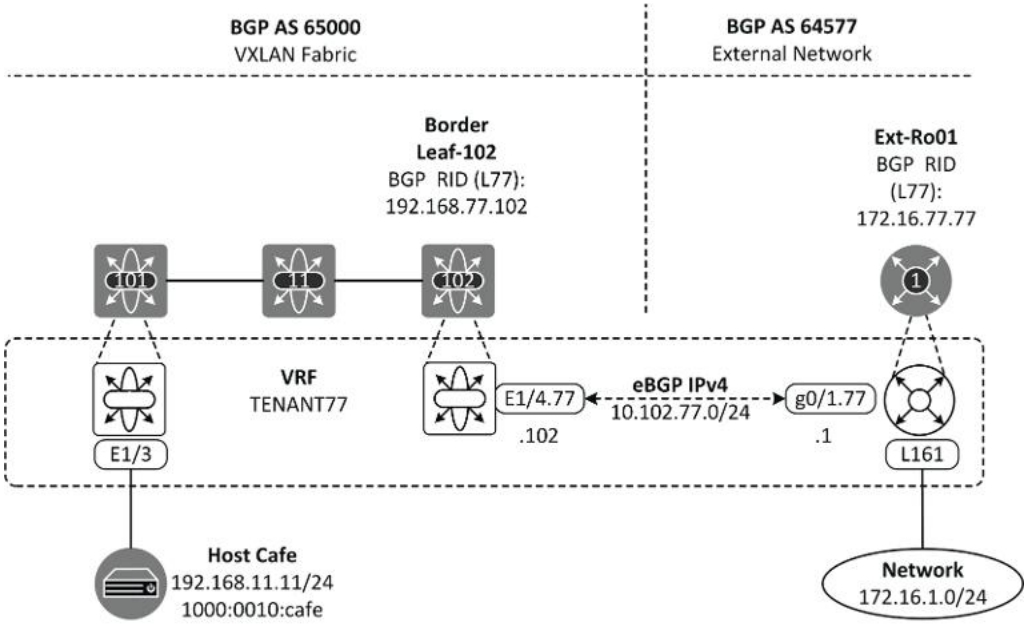


Figure 10-2: *VXLAN Fabric external connection topology*

Example 10-1 shows the BGP configurations on Border Leaf-102. Five last lines are related to peering with Ext-Ro01.

```
router bgp 65000
  router-id 192.168.77.102
  address-family ipv4 unicast
  address-family l2vpn evpn
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
    neighbor 10.102.77.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.77
      address-family ipv4 unicast
```

Example 10-1: *BGP configuration of Border Leaf-102*

Example 10-2 shows the vrf TENANT77 specific configurations on Ext-Ro01. Note that there is no routing information exchange between VXLAN Fabric and external network.

```
Ext-Ro01#sh run vrf TENANT77
<snipped>
ip vrf TENANT77
 rd 65077:1
 route-target export 65077:1
 route-target import 65077:1
!
<snipped>
!
interface GigabitEthernet0/1.77
 encapsulation dot1Q 77
 ip vrf forwarding TENANT77
 ip address 10.102.77.1 255.255.255.0
!
interface Loopback161
 description ** This Interface simulates external net 172.16.1.0/24 **
 ip vrf forwarding TENANT77
 ip address 172.16.1.1 255.255.255.0
!
router bgp 64577
 !
 address-family ipv4 vrf TENANT77
  neighbor 10.102.77.102 remote-as 65000
  neighbor 10.102.77.102 description ** VXLAN Fabric Border Leaf-102 **
  neighbor 10.102.77.102 update-source GigabitEthernet0/1.77
  neighbor 10.102.77.102 activate
 exit-address-family
```

Example 10-2: *VRF TENANT77 configurations on Border Leaf-102*

Starting point

Host Cafe 192.168.11.11/32 is not yet connected to Leaf-101 neither Ext-Ro01 does not advertise network 172.16.1.0/24 to Border Leaf-102.

Step-1: Ext-Ro01 starts advertising network 172.16.1.0/24 to its eBGP peer Border Leaf-102 (Figure 10-3). It generates a BGP Update message.

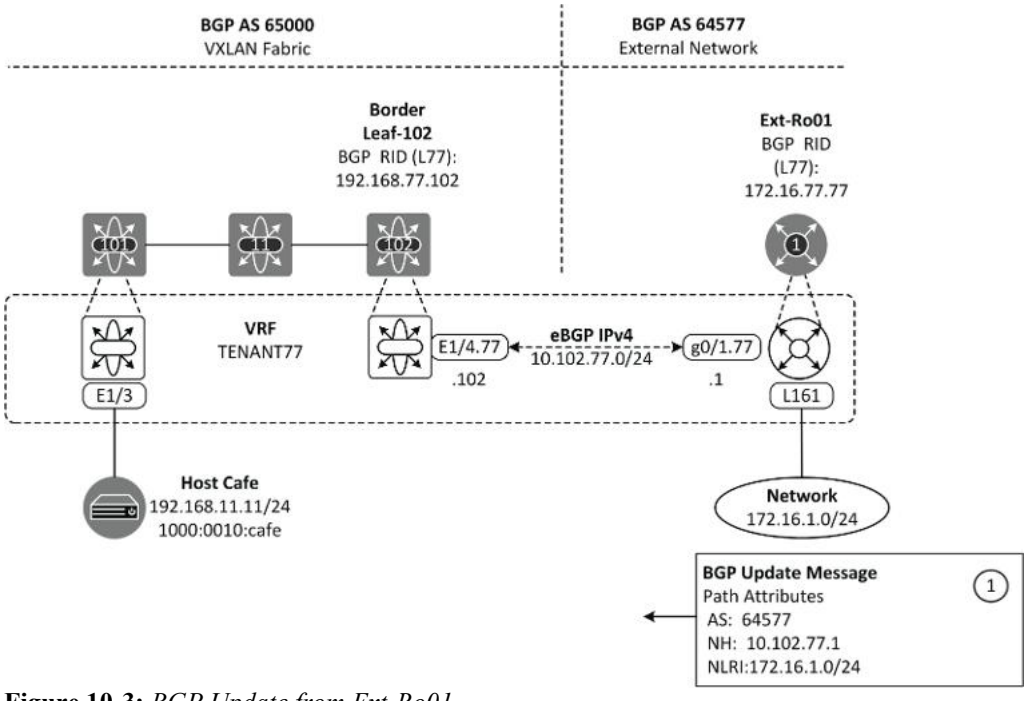


Figure 10-3: *BGP Update from Ext-Ro01*

The BGP Update message taken from the router Ext-Ro01 interface G0/1.77 is shown in capture 10-1. BGP Update message includes Path Attributes: Origin, AS_Path, and Next_Hop and the NLRI which defines the actual network.

```
Ethernet II, Src: fa:16:3e:5a:9b:23,Dst: 5e:00:00:01:00:07
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 77
IPv4, Src: 10.102.77.1, Dst: 10.102.77.102
Transmission Control Protocol,
Src Port: 26323, Dst Port: 179, Seq: 20, Ack: 20, Len: 54
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 54
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 27
    Path attributes
    Network Layer Reachability Information (NLRI)
        172.16.1.0/24
            NLRI prefix length: 24
            NLRI prefix: 172.16.1.0
```

Capture 10-1: *BGP NLRI update from Ext-Ro01 to Border Leaf-102.*

Step-2: Border Leaf-102 receives the BGP Update message from its interface E1/4.77. It creates two BGP routing table entries, one under the IPv4 Unicast AFI (Example 10-3) and the other one under the L2VPN EVPN AFI (Example 10-4). Since the BGP Update about 172.16.1.0/24 was received from the interface that belongs to the vrf context TENANT77, Border Leaf-102 attached the RT 65000:10077 (Extended Community Path Attribute) to BGP table entry to the BGP table of both AFI.

```
Leaf-102# sh ip bgp vrf TENANT77 172.16.1.0
BGP routing table information for VRF TENANT77, address family IPv4 Unicast
BGP routing table entry for 172.16.1.0/24, version 6
Paths: (1 available, best #1)
Flags: (0x880c041a) on xmit-list, is in urib, is best urib route, is in HW,
exported
  vpn: version 6, (0x100002) on xmit-list

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: external, path is valid, is best path, in rib
  AS-Path: 64577 , path sourced external to AS
    10.102.77.1 (metric 0) from 10.102.77.1    (172.16.77.77)
      Origin IGP, MED 0, localpref 100, weight 0
      Extcommunity: RT:65000:10077
```

Example 10-3: *BRIB entry for VRF TENANT77 - AFI IPv4 Unicast (Border Leaf-102)*

Example 10-4 shows that the BGP table entry under L2VPN EVPN AFI also includes Route Distinguisher 192.168.77.102:3 taken from the VRF Context TENANT77 (L3VNI ID 10077). In addition to RT 65000:10077 Extended community there is also encapsulation type-8 which means the VXLAN encapsulation.

```
Leaf-102# sh bgp l2vpn evpn 172.16.1.0 vrf TENANT77
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.102:3    (L3VNI10077)
BGP routing table entry for [5]:[0]:[0]:[24]:[172.16.1.0]:[0.0.0.0]/224,
version 12
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn

  Advertised path-id 1
  Path type: local, path is valid, is best path
  AS-Path: 64577 , path sourced external to AS
    192.168.100.102 (metric 0) from 0.0.0.0 (192.168.77.102)
      Origin IGP, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0001.0007
```

Example 10-4: *BRIB entry for VRF TENANT77 - AFI L2VPN EVPN (Border Leaf-102)*

Step-3: Border Leaf-102 constructs the BGP EVPN Route-Type 5 (IP Prefix Advertisement) Update (Figure 10-4 and Capture 10-2) and sends it toward BGP Route Reflector, which in turn forwards it without modification of the content (though it will add a cluster list as a routing advertisement loop prevention mechanism) to its RR-Client Leaf-101.

Note that switch Spine-11 (BGP RR) is unaware of any VRF information. But it can handle overlapping IPv4 routing updates since each VRF Context in our VXLAN fabric has different dedicated, auto-generated RD value which is used only with the network belonging to particular VRF Context. There is only one VRF context, TENANT77 but if the other one will be created, it will get on unique RD. Just for the recap, VRF Context auto-RD is formed based on the BGP RDI and VRF-Id

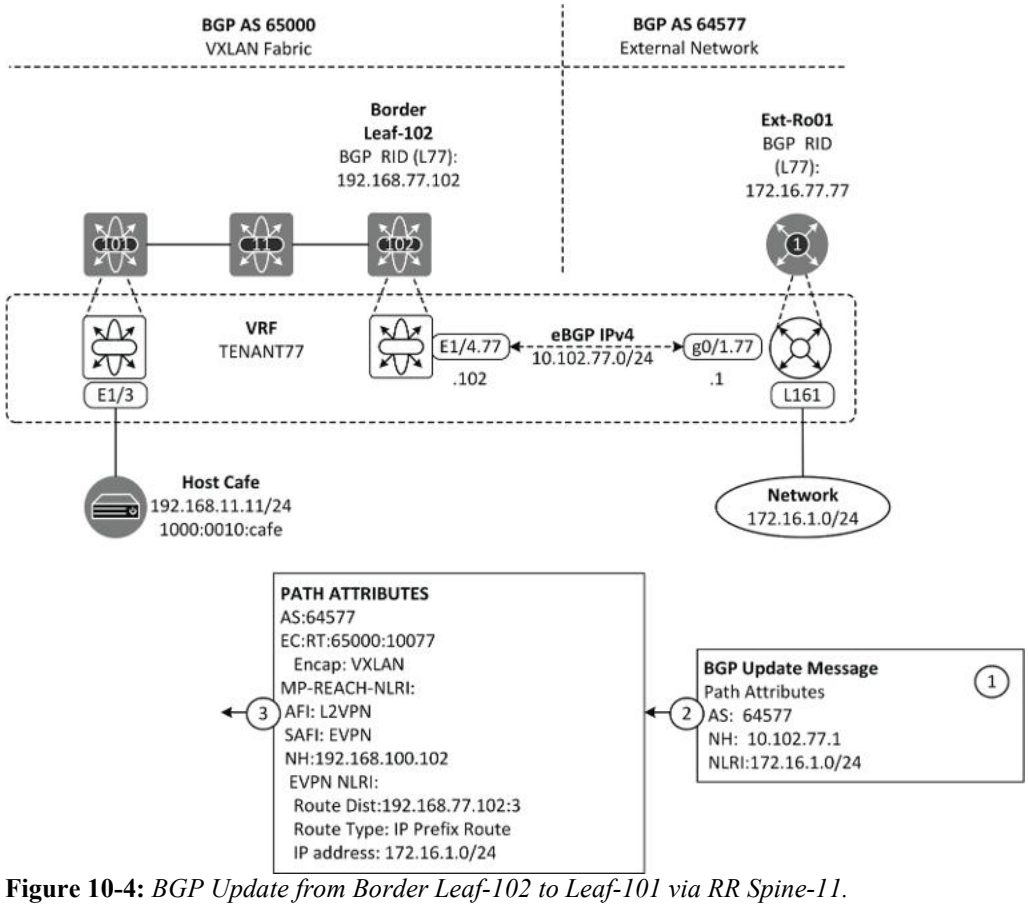


Figure 10-4: *BGP Update from Border Leaf-102 to Leaf-101 via RR Spine-11.*

```
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 126
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 103
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: 64577
        Path Attribute - MULTI_EXIT_DISC: 0
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 24
            Carried extended communities: (3 communities)
                Route Target: 65000:10077
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
        Path Attribute - MP_REACH_NLRI
            Type Code: MP_REACH_NLRI (14)
            Length: 45
```

```
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (36 bytes)
                EVPN NLRI: IP Prefix route
                    Route Type: IP Prefix route (5)
                    Length: 34
                    Route Distinguisher: 192.168.77.102:3
                    ESI: 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    IP prefix length: 24
                    IPv4 address: 172.16.1.0
                    IPv4 Gateway address: 0.0.0.0
                    MPLS Label Stack: 629 (bottom)
```

Capture 10-2: *BGP NLRI update from Border Leaf-102 to Leaf-101 (via RR Spine-11)*

Step-4: Leaf-101 receives the BGP Update, it import routing update based on RT 65000:10077 which is configured under its vrf context TENANT77. It creates an L3VNI entry for the network 172.16.1.0/24.

```
Leaf-101# sh bgp l2vpn evpn 172.16.1.0 vrf TENANT77
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.102:3
BGP routing table entry for [5]:[0]:[0]:[24]:[172.16.1.0]:[0.0.0.0]/224,
version 14
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported to 2 destination(s)
  AS-Path: 64577 , path sourced external to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0001.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
BGP routing table entry for [5]:[0]:[0]:[24]:[172.16.1.0]:[0.0.0.0]/224,
version 15
Paths: (1 available, best #1)
Flags: (0x000002) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported from
192.168.77.102:3:[5]:[0]:[0]:[24]:[172.16.1.0]:[0.0.0.0]/224
  AS-Path: 64577 , path sourced external to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 RouterzMAC:5e00.0001.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111

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

Example 10-5: *BRIB entry in Leaf-101.*

So, what does all of this information tells to the receiver Leaf-101? First, route is exported by Border Leaf-102 with RT 65000:10077 which in turns means that there has to be an import clause for that RT value in Leaf-101 under its vrf context TEANANT77 (remember this is L3 service inside TENANT77), otherwise the route does not end up to BGP table (Control Plane operation). Then, if some of its connected hosts send a packet targeted to network 172.16.1.0/24, the packet needs to be encapsulated with a new header, where the destination IP address is the NVE1 interface address of Border Leaf-102 and the VXLAN Virtual Network Identifier is VNI 10077 (Data Plane operation). We could also compare RIB tables between Border Leaf-102 and Leaf-101. Example 10-6 shows that Border Leaf-102 has learned route via BGP from the 10.102.77.1 (Ext-Ro01), the update is external (Admin Distance 20) and remote AS is 64577.

```
Leaf-102# sh ip route 172.16.1.0 vrf TENANT77 | sec 172.16.1.0
172.16.1.0/24, ubest/mbest: 1/0
    *via 10.102.77.1, [20/0], 00:35:13, bgp-65000, external, tag 64577
```

Example 10-6 shows the RIB of the VTEP Leaf-101. There is additional Data Plane information; VNI segment ID 10077, which is our L3VNI inside TENANT77 used in the VXLAN header VNI field. There is also information about tunnel id.

```
Leaf-101# sh ip route vrf TENANT77 | sec 172.16.1.0
172.16.1.0/24, ubest/mbest: 1/0
    *via 192.168.100.102%default, [200/0], 00:15:15, bgp-65000, internal, tag
64577 (evpn) segid: 10077 tunnelid: 0xc0a86466 encap: VXLAN
```

Example 10-6: *RIB entry in Leaf-101.*

The information in Example 10-7 is not directly related to routing update itself but it useful while doing troubleshooting. We can see that rnh database (Recursive Next Hop) of the VTEP Leaf-101 has information about Border Leaf-102 IP as well as associate tunnel id.

```
Leaf-101# sh nve internal bgp rnh database
--------------------------------------------
Total peer-vni msgs recvd from bgp: 1
Peer add requests: 1
Peer update requests: 0
Peer delete requests: 0
Peer add/update requests: 1
Peer add ignored (peer exists): 0
Peer update ignored (invalid opc): 0
Peer delete ignored (invalid opc): 0
Peer add/update ignored (malloc error): 0
Peer add/update ignored (vni not cp): 0
Peer delete ignored (vni not cp): 0
--------------------------------------------
Showing BGP RNH Database, size : 1 vni 0

Flag codes: 0 - ISSU Done/ISSU N/A        1 - ADD_ISSU_PENDING
            2 - DEL_ISSU_PENDING          3 - UPD_ISSU_PENDING


VNI    Peer-IP            Peer-MAC            Tunnel-ID  Encap     (A/S)  Flags
10077  192.168.100.102    5e00.0001.0007      0xc0a86466 vxlan     (1/0)    0
```

Example 10-7: *RNH database on Leaf-101.*

Example 10-8 shows that the tunnel between Leaf switches is up and running. In addition, it shows that Symmetric IRB (Integrated Route and Bridge) first-hop routing operation is used (draft-ietf-bess-evpn-inter-subnet-forwarding-03).

```
Leaf-101# sh nve peers detail
Details of nve Peers:
---------------------------------------
Peer-Ip: 192.168.100.102
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 01:21:51
    Router-Mac          : 5e00.0001.0007
    Peer First VNI      : 10077
    Time since Create   : 01:21:52
    Configured VNIs     : 10000,10077,20000
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10077
    vni assignment mode : SYMMETRIC
    Peer Location       : N/A
```

Example 10-8: *RNH database on Leaf-101.*

Next, we will take a look at how the route to IP address 192.168.11.11/32 of host Cafe ends up to BGP table and RIB of Ext-Ro01.

Step-4 and 5: Now the host Cafe with IP 192.168.11.11/32 joins the network. It sends a Gratuitous ARP. VTEP switch Leaf-101 learns both the MAC- and IP addresses of Host Cafe from the GARP message. It sends two BGP Update messages to its BGP EVPN peers. The first message contains MAC address information and the second message contains both MAC and IP information. Though messages can be sent within the same update message.

The Host Mobility Manager component of Nexus 9000v installs the route to both L2RIB and L3RIB and from there the route is sent to the BGP VRF AFI process. The BGP process constructs the BGP Update message (figure 10-5) with BGP EVPN Path Attributes. AS field is left empty since this is an internal BGP Update. Both L2 and L3 Route-Targets are attached to Extended Community field as well as Encapsulation type, VXLAN (Type-8).

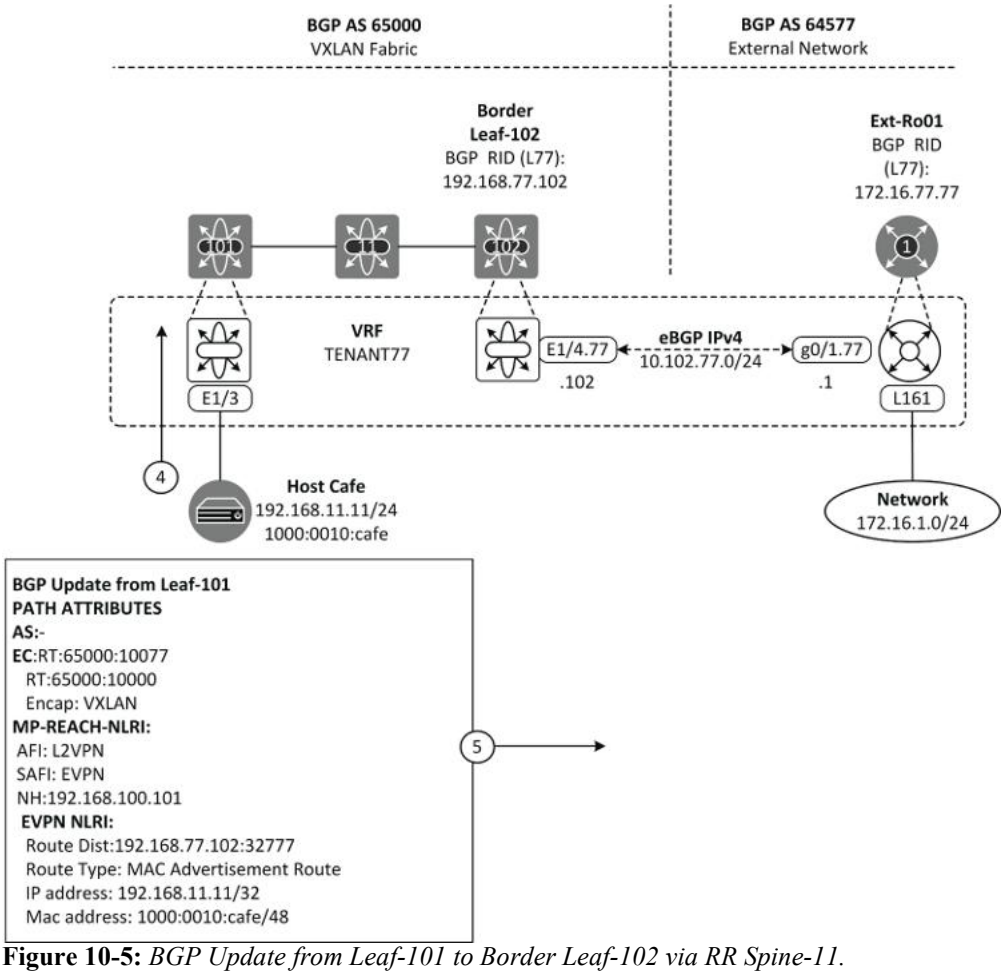


Figure 10-5: *BGP Update from Leaf-101 to Border Leaf-102 via RR Spine-11.*

The BGP Update message is shown in Capture 10-3, which is taken from the Uplink between Spine-11 and Border Leaf-102.

```
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 141
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 118
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 32
            Carried extended communities: (4 communities)
        Path Attribute - ORIGINATOR_ID: 192.168.77.101
            Flags: 0x80, Optional, Non-transitive, Complete
```

```
            Type Code: ORIGINATOR_ID (9)
            Length: 4
            Originator identifier: 192.168.77.101
        Path Attribute - CLUSTER_LIST: 192.168.77.111
            Flags: 0x80, Optional, Non-transitive, Complete
            Type Code: CLUSTER_LIST (10)
            Length: 4
            Cluster List: 192.168.77.111
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transi., Complete
            Type Code: MP_REACH_NLRI (14)
            Length: 51
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (42 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 40
                    Route Distinguisher: 192.168.77.101:32777
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: Private_10:ca:fe (10:00:00:10:ca:fe)
                    IP Address Length: 32
                    IPv4 address: 192.168.11.11
                    MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT
set!)
                    MPLS Label Stack 2: 629 (bottom)
```

Capture 10-3: *BGP NLRI update from Leaf-101 to Border Leaf-102 (via RR Spine-11)*

Example 10-9 shows that Border Leaf-102 has received BGP Update from Leaf-101 and installed it to BRIB based RT 64500:10077 defined under the VRF Context TENANT77.

```
Leaf-102# sh bgp l2vpn evpn 192.168.11.11 | beg L3VNI
Route Distinguisher: 192.168.77.102:3    (L3VNI 10077)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272, version 7
Paths: (1 available, best #1)
Flags: (0x000202) on xmit-list, is not in l2rib/evpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path
             Imported from
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5e00.0000.0007
      Originator: 192.168.77.101 Cluster list: 192.168.77.111

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

Example 10-9: *BRIB entry in Border Leaf-102.*

Step-6: Border Leaf-102 forwards BGP update to Ext-Ro01 (Figure 10-6). Since the peering between the Border Leaf-102 and Ext-Ro01 is done under the AFI IPv4 only the mandatory Path Attributes related to AFI IPv4 are attached to BGP Update.

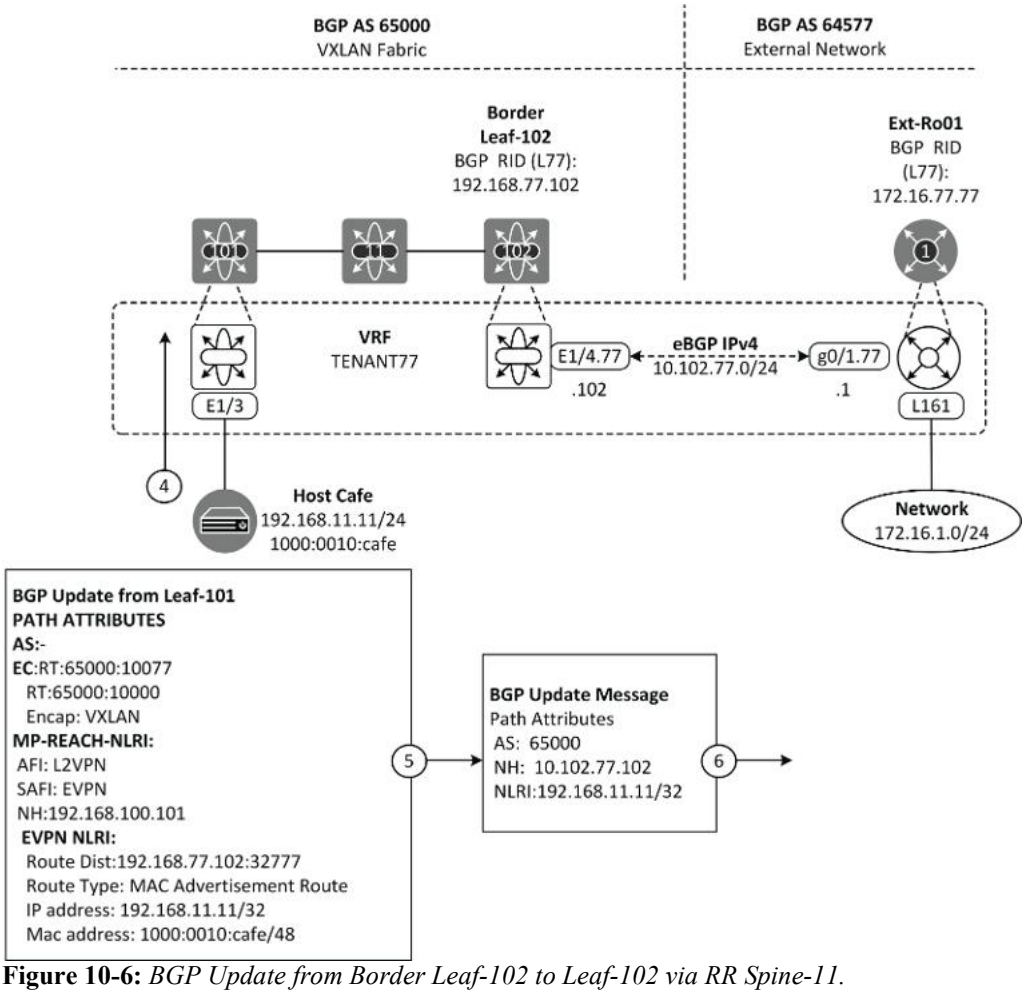


Figure 10-6: *BGP Update from Border Leaf-102 to Leaf-102 via RR Spine-11.*

Capture 10-4 taken from the link between the Border Leaf-102 and Ext-Ro01 shows the BGP Update message send by Border Leaf-102.

```
Border Gateway Protocol - UPDATE Message
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 20
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: 65000
        Path Attribute - NEXT_HOP: 10.102.77.102
    Network Layer Reachability Information (NLRI)
        192.168.11.11/32
            NLRI prefix length: 32
            NLRI prefix: 192.168.11.11
```

Capture 10-4: *BGP NLRI update from Leaf-101 to Border Leaf-102 (via RR Spine-11)*

Example 10-10 illustrates that Ext-Ro01 has information about 192.168.11.11/32 received from Border Leaf-102. Note that there is an RD 65077:1 and RT 65077:1 attached to BGP entry even though those were not included in the BGP update. So where does that information comes from?

```
Ext-Ro01#sh ip bgp vpnv4 vrf TENANT77 192.168.11.11
BGP routing table entry for 65077:1:192.168.11.11/32, version 5
Paths: (1 available, best #1, table TENANT77)
  Not advertised to any peer
  Refresh Epoch 1
  65000
    10.102.77.102 (via vrf TENANT77) from 10.102.77.102 (192.168.11.1)
      Origin IGP, localpref 100, valid, external, best
      Extended Community: RT:65077:1
      rx pathid: 0, tx pathid: 0x0
Ext-Ro01#
```

Example 10-10: *BRIB entry in Ext-Ro01.*

Those were defined under the local vrf configuration in Ext-Ro01.

```
Ext-Ro01#sh run vrf TENANT77 | sec ip vrf
ip vrf TENANT77
 rd 65077:1
 route-target export 65077:1
 route-target import 65077:1
```

Example 10-11: *VRF information in Ext-Ro01.*

Information is installed from the BRIB to RIB.

```
Ext-Ro01#sh ip route vrf TENANT77 bgp | sec 192
      192.168.11.0/32 is subnetted, 1 subnets
B        192.168.11.11 [20/0] via 10.102.77.102, 00:33:13
```

Example 10-12: *VRF information in Ext-Ro01.*

Example 10-13 verifies that the Data Plane is ok and there is an IP connectivity between the network 172.16.1.0/24 connected to Ext-Ro01 and host Cafe 192.168.11.11 connected to VTEP Leaf-101.

```
Ext-Ro01#ping vrf TENANT77 192.168.11.11 source 172.16.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.11.11, timeout is 2 seconds:
Packet sent with a source address of 172.16.1.1
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 24/34/54 ms
```

Example 10-13: *Ping from Ext-Ro01 to host Cafe.*

Now host Abba with IP address 192.168.11.100 connected to Leaf-101 is brought up. Example 10-14, verifies that Ext-Ro01 receives BGP Update just like it should be.

```
Ext-Ro01#debug ip routing
IP routing debugging is on
Ext-Ro01#
*May 27 17:44:13.722: RT(TENANT77): updating bgp 192.168.11.100/32 (0x1)  :
    via 10.102.77.102   0 1048577

*May 27 17:44:13.723: RT(TENANT77): add 192.168.11.100/32 via 10.102.77.102,
bgp metric [20/0]
```

Example 10-14: *RIB update in Ext-Ro01.*

And now it has separate routes to both hosts Cafe 192.168.11.11 and Abba 192.168.11.100 as can see from the example 10-14

```
Ext-Ro01#sh ip route vrf TENANT77 bgp | sec 192.
      192.168.11.0/32 is subnetted, 2 subnets
B        192.168.11.11 [20/0] via 10.102.77.102, 00:48:32
B        192.168.11.100 [20/0] via 10.102.77.102, 00:04:53
```

Example 10-15: RIB in Ext-Ro01.

Even though there is an IP connectivity now from the external network to network in VXLAN Fabric and vice versa, we do not want to install each and every host route to the Ext-Ro01 RIB. As a next step, the host routes are aggregated within a summary-route. That is done under the vrf TENANT77 ipv4 unicast afi.

```
router bgp 65000
  router-id 192.168.77.102
  address-family ipv4 unicast
  address-family l2vpn evpn
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 192.168.11.0/24 summary-only
    neighbor 10.102.77.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.77
      address-family ipv4 unicast
```

Example 10-16: *Aggregation in Border Leaf-102.*

After aggregation, there is only one routing entry in Ext-Ro01 RIB As can be seen from example 10-17.

```
Ext-Ro01#sh ip route vrf TENANT77 bgp | b 192
B     192.168.11.0/24 [20/0] via 10.102.77.102, 00:06:5
```

Example 10-17: *RIB in Ext-Ro01.*

This section explains how to set up a dual-homed, full-mesh eBGP peering. The focus of this section is to build a BGP policy model, where both incoming and outgoing paths of specifics networks can be controlled via Border Leaf switches without doing any changes in external routers Ext-Ro01 and Ext-Ro02. The section includes the policy model, which only affects the path selection. It does not include any incoming and outgoing route filtering, neither optimize the BGP convergence time by using BFD or object/interface tracking or changing BGP keepalive/hold-down timers. The private networks defined in RFC1918 (since we are using those) and default route 0.0.0.0/0 are not filtered out. In addition the policy defined in this section prevents the External network AS64577 to use VXLAN Fabric as a transit network between Ext-Ro01 and Ext-Ro02 in case of backbone failure in AS65077.

OSPF is used as an IGP inside the AS64577. Routes learned from BGP are redistributed into OSPF. Ext-Ro01 redistributes routes with metric 10 while the Ext-Ro02 uses metric 100. This way the Ext-Ro03 will prefer route learned from the Ext-Ro01. I also set the "metric-type 1" in both routers to make sure that the metric to ASBR is included in the path cost. Figure 10-7 shows the topology used in this example.

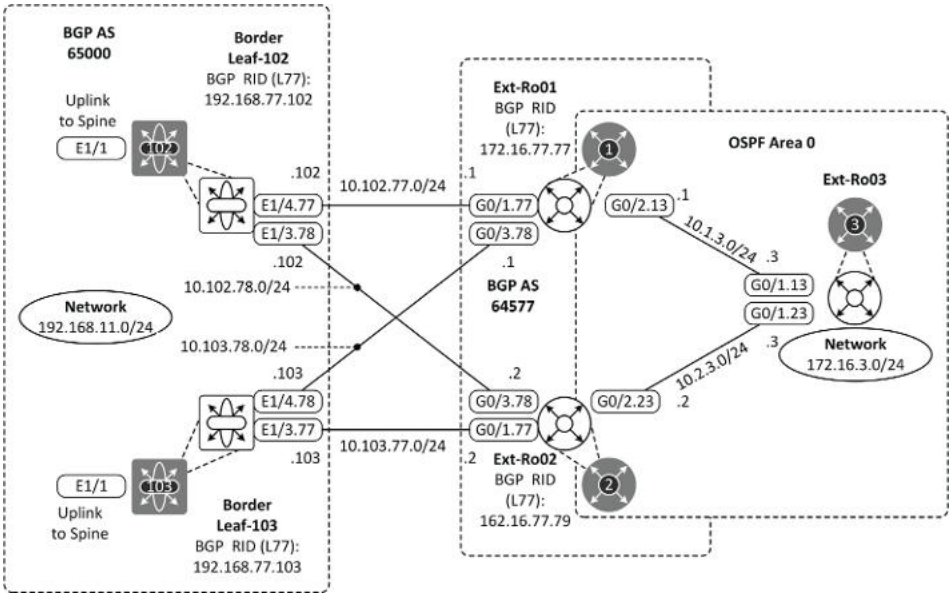


Figure 10-7: *full-mesh external BGP peering topology.*

BGP configurations are shown in Examples through 10-18 to 10-21. The configuration of Ext-Ro01 and Ext-Ro02 also includes the OSPF configuration.

```
Leaf-102# sh run | sec bgp
feature bgp
  host-reachability protocol bgp
router bgp 65000
  timer bgp 3 9
  router-id 192.168.77.102
  address-family ipv4 unicast
  address-family l2vpn evpn
```

```
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 192.168.11.0/24 summary-only
    neighbor 10.102.77.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.77
      address-family ipv4 unicast
        send-community
        send-community extended
    neighbor 10.102.78.2
      remote-as 64577
      description ** External Network - Ext-Ro02 **
      update-source Ethernet1/3.78
      address-family ipv4 unicast
```

Example 10-18: *Border Leaf-102 BGP configuration.*

```
Leaf-103# sh run | sec bgp
feature bgp
  host-reachability protocol bgp
router bgp 65000
  timer bgp 3 9
  router-id 192.168.77.103
  address-family ipv4 unicast
  address-family l2vpn evpn
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 192.168.11.0/24 summary-only
    neighbor 10.103.77.2
      remote-as 64577
      description ** External Network - Ext-Ro02 **
      update-source Ethernet1/3.77
      address-family ipv4 unicast
    neighbor 10.103.78.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.78
      address-family ipv4 unicast
```

Example 10-19: *Border Leaf-103 BGP configuration.*

```
Ext-Ro01#
router ospf 1 vrf TENANT77
 redistribute bgp 64577 metric 10 metric-type 1 subnets
!
router bgp 64577
 timer bgp 3 9
 bgp router-id 172.16.77.77
```

```
 bgp log-neighbor-changes
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 vrf TENANT77
  network 172.16.1.0 mask 255.255.255.0
  network 172.16.3.0 mask 255.255.255.0
  neighbor 10.102.77.102 remote-as 65000
  neighbor 10.102.77.102 description ** VXLAN Fabric Border Leaf-102 **
  neighbor 10.102.77.102 update-source GigabitEthernet0/1.77
  neighbor 10.102.77.102 activate
  neighbor 10.103.78.103 remote-as 65000
  neighbor 10.103.78.103 description ** VXLAN Fabric Border Leaf-103 **
  neighbor 10.103.78.103 update-source GigabitEthernet0/3.78
  neighbor 10.103.78.103 activate
 exit-address-family
Ext-Ro01#
```

Example 10-20: *Ext-Ro01 BGP configuration.*

```
Ext-Ro02#
router ospf 1 vrf TENANT77
 redistribute bgp 64577 metric 100 metric-type 1 subnets
!
router bgp 64577
 timer bgp 3 9
 bgp router-id 172.16.77.79
 bgp log-neighbor-changes
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 vrf TENANT77
  network 172.16.3.0 mask 255.255.255.0
  neighbor 10.102.78.102 remote-as 65000
  neighbor 10.102.78.102 description ** VXLAN Fabric Border Leaf-102 **
  neighbor 10.102.78.102 update-source GigabitEthernet0/3.78
  neighbor 10.102.78.102 activate
  neighbor 10.103.77.103 remote-as 65000
  neighbor 10.103.77.103 description ** VXLAN Fabric Border Leaf-103 **
  neighbor 10.103.77.103 update-source GigabitEthernet0/1.77
  neighbor 10.103.77.103 activate
 exit-address-family
Ext-Ro02#
```

Example 10-21: *Ext-Ro02 BGP configuration.*

Example 10-22 shows that the Border Leaf-102 has learned route 172.16.3.0/24 from Ext-R01 (best), from Ext-Ro02 and from Spine-11. This decision is based on the lower RID of Ext-Ro01 (Ext-Ro01 BGP RID 172.16.77.78 and Ext-Ro02 BGP RID 172.16.77.79).

```
Leaf-102# sh ip bgp vrf TENANT77
<snipped>
   Network            Next Hop            Metric  LocPrf  Weight Path
* i172.16.1.0/24      192.168.100.103     0        100        0 64577 i
*>e                   10.102.77.1         0                   0 64577 i
* i172.16.3.0/24      192.168.100.103     2        100        0 64577 i
* e                   10.102.78.2         2                   0 64577 i
*>e                   10.102.77.1               2             0 64577 i
  a192.168.11.0/24    0.0.0.0                     100      32768 i
```

Example 10-22: *Leaf-102 BGP routes.*

Example 10-23 shows that also the Border Leaf-103 has learned route 172.16.3.0/24 from Ext-R01 (best), from Ext-Ro02 and from Spine-11. This decision is also based on the lower RID of Ext-Ro01. Note that both Border Leaf switches are receiving BGP Update about 172.16.3.0/24 also from the VXLAN Fabric Spine switch, which is BGP Route-Reflector. Since the internal BGP has worse Administrative Distance (200) than an external BGP (20), it is only a third-best route.

```
Leaf-103# sh ip bgp vrf TENANT77
<snipped>
   Network            Next Hop          Metric  LocPrf  Weight Path
* i172.16.1.0/24      192.168.100.102      0    100          0 64577 i
*>e                   10.103.78.1          0                 0 64577 i
* i172.16.3.0/24      192.168.100.102      2    100          0 64577 i
*>e                   10.103.78.1          2                 0 64577 i
* e                   10.103.77.2          2                 0 64577 i
  a192.168.11.0/24    0.0.0.0                   100           32768 i
```

Example 10-23: *Leaf-102 BGP routes.*

Example 10-24 shows that the Border Ext-Ro01 has learned aggregate route 192.168.11.0/24 from Leaf-102 (best) and from Leaf-103.

```
Ext-Ro01#sh ip bgp vpnv4 vrf TENANT77
<snipped>
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 65077:1 (default for vrf TENANT77)
 *>   172.16.1.0/24    0.0.0.0                  0         32768 i
 *>   172.16.3.0/24    10.1.3.3                 2         32768 i
 *>   192.168.11.0     10.102.77.102                          0 65000 i
 *                     10.103.78.103                          0 65000 i
```

Example 10-24: *Ext-Ro01 BGP routes.*

Example 10-25 shows that the Border Ext-Ro02 has learned aggregate route 192.168.11.0/24 from Leaf-102 (best) and from Leaf-103. Once again the best path selection is based on the lowest BGP peer RID.

```
Ext-Ro01#sh ip bgp vpnv4 vrf TENANT77
<snipped>
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 65077:1 (default for vrf TENANT77)
 *>   172.16.1.0/24    0.0.0.0                  0         32768 i
 *>   172.16.3.0/24    10.1.3.3                 2         32768 i
 *>   192.168.11.0     10.102.77.102                          0 65000 i
 *                     10.103.78.103                          0 65000 i
```

Example 10-25: *Ext-Ro02 BGP routes.*

At this point, there is no BGP policy between the eBGP peers. The IP connectivity between the network 192.168.11.0/24 in AS65000 and network 172.16.3.0/24 in AS64577 is tested by pinging from host Cafe (192.168.11.11) to address 172.16.3.1 (Loopback 163 on Ext-Ro03) (Example 10-26).

```
Cafe#ping 172.16.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/17/22 m
```

Example 10-26: *Ping from 192.168.11.1 to 172.16.3.1.*

BGP policy configuration

Step-1: Tag the BGP updates about network 192.168.11.0/24 sent external peers by Border-Leaf-102 with the Community Path Attribute 64577:999.

Border Leaf-102:

Step-1.1: Define the prefix-list for VXLAN Fabric internal network 192.168.11.0/24.

Step-1.2: Define the route-map that matches (permit) the previously defined ip prefix-list and set the community 64577:999 for it. Add implicit permit as a last line of route-map.

Step-1.3: Implement an outgoing policy towards both external BGP peers Ext-Ro01 and Ext-Ro02.

Step-1.4: Since extended communities are not sent to BGP peer by default, allow extended communities to be sent to eBGP peer. You could allow just the standard communities even though in example configuration both standard and extended communities are permitted.

```
ip prefix-list TENANT77_LOCAL seq 10 permit 192.168.11.0/24
!
route-map OUTGOING_POLICIES permit 10
  match ip address prefix-list TENANT77_LOCAL
  set community 64577:999
!
route-map OUTGOING_POLICIES permit 100
!
Router bgp 65000
  vrf TENANT77
    address-family ipv4 unicast
    neighbor 10.102.77.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.77
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map OUTGOING_POLICIES out
    neighbor 10.102.78.2
      remote-as 64577
      description ** External Network - Ext-Ro02 **
      update-source Ethernet1/3.78
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map OUTGOING_POLICIES out
```

Example 10-27: *Border Leaf-102 outgoing BGP policy.*

Step-2: Tag BGP updates about network 192.168.11.0/24 sent to external eBGP peer by Border-Leaf-103 with Community Path Attribute 64577:9

Border Leaf-103

Step-2.5: Define the prefix-list for VXLAN Fabric internal network 192.168.11.0/24.

Step-2.6: Define the route-map that matches (permit) the previously defined ip prefix-list and set the community 64577:9 for it. Add implicit permit as a last line of route-map.

Step-2.7: Implement outgoing policy towards both external BGP peers Ext-Ro01 and Ext-Ro02.

```
ip prefix-list TENANT77_LOCAL seq 10 permit 192.168.11.0/24
!
route-map OUTGOING_POLICIES permit 10
  match ip address prefix-list TENANT77_LOCAL
  set community 64577:9
!
route-map OUTGOING_POLICIES permit 100
!
Router bgp 65000
  vrf TENANT77
    address-family ipv4 unicast
        neighbor 10.103.77.2
      remote-as 64577
      description ** External Network - Ext-Ro02 **
      update-source Ethernet1/3.77
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map OUTGOING_POLICIES out
    neighbor 10.103.78.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.78
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map OUTGOING_POLICIES out
```

Example 10-28: *Border Leaf-103 outgoing BGP policy.*

Figure 10-8 shows how Border switch Leaf-102 attaches the BGP community 64577:999 to updates sent to the eBGP peers. Border Leaf-103, in turn, attaches the BGP community 64577:9 to updates sent to its' eBGP peers.

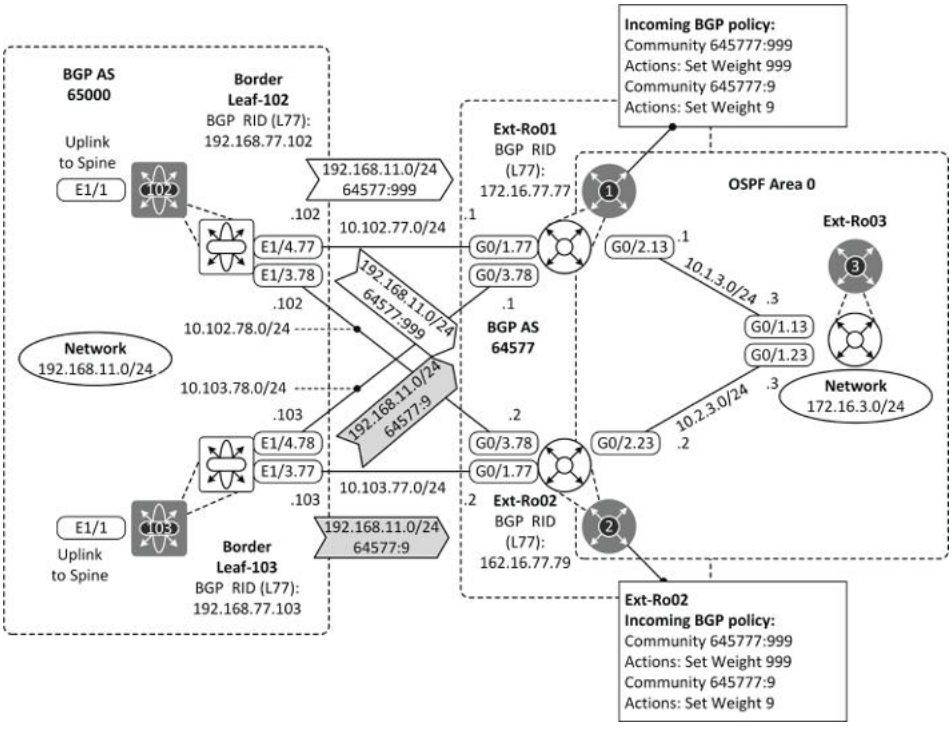


Figure 10-8: *BGP Update from Border Leaf-102 and 103 to External routers.*

Step-3: Define ingress Policy in Ext-Ro1 and Ext-Ro02. The ingress policy will select the best path based on the Community PA in incoming BGP Updates. Both Border routers in AS64577 will set the weight 999 for all BGP NLRI updates that include the Community PA 64577:999 and weight 9 for all BGP NLRI updates that includes the Community PA 64577:9. The BGP NLRI Updates received from the Border Leaf-102 will get higher Weight value in both routers Ext-Ro01 and Ext-Ro02, which in turn means that the route to the network 192.168.11.0/24 via Border Leaf-102 will be best in a stable state.

Ext-Ro01 and Ext-Ro02:

Step-3.1: Define the community-list that permits community PA 64577:999.

Step-3.2: Define the community-list that permits community PA 64577:9.

Step-3.3: Define the route-map that set the weight 999 for all of the BGP NLRI updates that carry community PA 64577:999 and weight 9 for all of the BGP NLRI updates that carries community attribute 64577:9.

Step-3: Implement ingress policy towards both Border Leaf Leaf-102 and Leaf 103.

Step-4: enable bgp-community new-format.

```
ip bgp-community new-format
!
ip community-list standard SET_WEIGHT_999 permit 64577:999
ip community-list standard SET_WEIGHT_9 permit 64577:9
!
route-map SET_WEIGHT permit 10
 match community SET_WEIGHT_999
 set weight 999
!
route-map SET_WEIGHT permit 100
 match community SET_WEIGHT_9
 set weight 9
!
router bgp 64577
 bgp router-id 172.16.77.77
 bgp log-neighbor-changes
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 vrf TENANT77
  network 172.16.1.0 mask 255.255.255.0
  network 172.16.3.0 mask 255.255.255.0
  neighbor 10.102.77.102 remote-as 65000
  neighbor 10.102.77.102 description ** VXLAN Fabric Border Leaf-102 **
  neighbor 10.102.77.102 update-source GigabitEthernet0/1.77
  neighbor 10.102.77.102 activate
  neighbor 10.102.77.102 route-map SET_WEIGHT in
  neighbor 10.103.78.103 remote-as 65000
  neighbor 10.103.78.103 description ** VXLAN Fabric Border Leaf-103 **
  neighbor 10.103.78.103 update-source GigabitEthernet0/3.78
  neighbor 10.103.78.103 activate
  neighbor 10.103.78.103 route-map SET_WEIGHT in
 exit-address-family
```

Example 10-29: *Ext-Ro01 ingress BGP policy.*

```
ip bgp-community new-format
ip community-list standard SET_WEIGHT_999 permit 64577:999
ip community-list standard SET_WEIGHT_9 permit 64577:9
!
route-map SET_WEIGHT permit 10
 match community SET_WEIGHT_999
 set weight 999
!
route-map SET_WEIGHT permit 100
 match community SET_WEIGHT_9
 set weight 9
!
router bgp 64577
 bgp router-id 172.16.77.77
 bgp log-neighbor-changes
 !
 address-family ipv4
```

```
 exit-address-family
 !
  address-family ipv4 vrf TENANT77
  network 172.16.3.0 mask 255.255.255.0
  neighbor 10.102.78.102 remote-as 65000
  neighbor 10.102.78.102 description ** VXLAN Fabric Border Leaf-102 **
  neighbor 10.102.78.102 update-source GigabitEthernet0/3.78
  neighbor 10.102.78.102 activate
  neighbor 10.102.78.102 route-map SET_WEIGHT in
  neighbor 10.103.77.103 remote-as 65000
  neighbor 10.103.77.103 description ** VXLAN Fabric Border Leaf-103 **
  neighbor 10.103.77.103 update-source GigabitEthernet0/1.77
  neighbor 10.103.77.103 activate
  neighbor 10.103.77.103 route-map SET_WEIGHT in
 exit-address-family
```

Example 10-30: *Ext-Ro02 ingress BGP policy.*

Figure 10-9 shows the egress policy for network 192.168.11.0/24 implemented in AS65000 and the ingress policy implemented in AS64577.

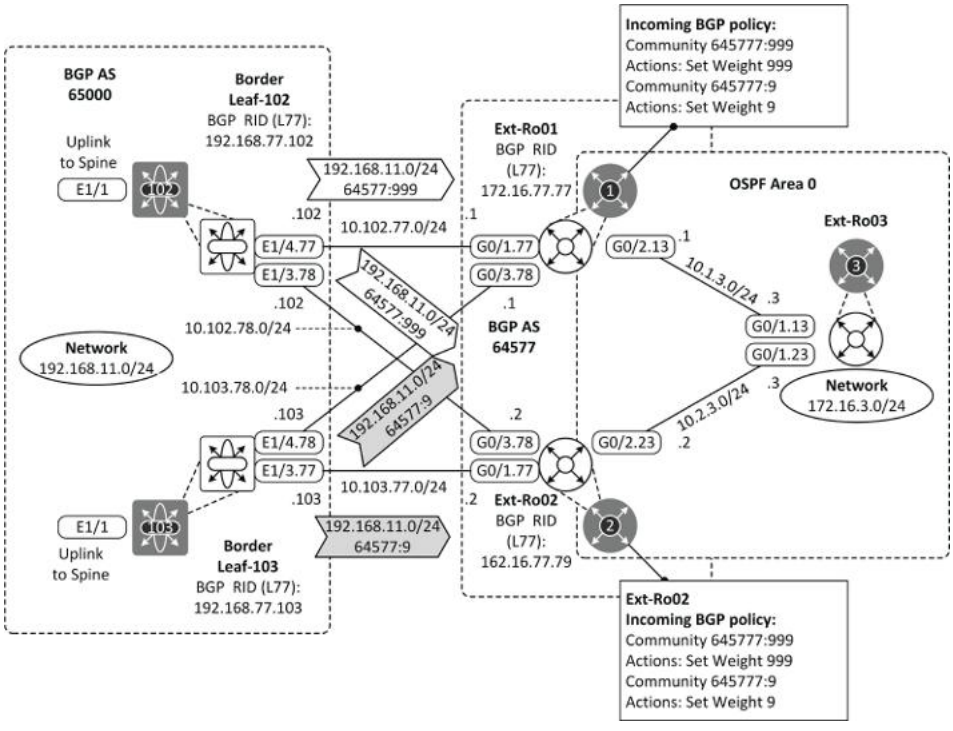


Figure 10-9: *BGP policy concerning network 192.168.11.0/24.*

Policy verification:

Example 10-31 shows that Ext-Ro01 has received BGP NLRI Update about network 192.168.11.11 from the Border Leaf-102 with community PA 645777:999. Based on the community PA, Ext-Ro01 has set the BGP weight 999 for the route received from Border Leaf-102. It also has received BGP NLRI Update from the Border-Leaf 103 but with the community PA 64577:99 which gets the weight 9.

```
Ext-Ro01#sh ip bgp vpnv4 vrf TENANT77 192.168.11.0
BGP routing table entry for 65077:1:192.168.11.0/24, version 18
Paths: (2 available, best #1, table TENANT77)
  Advertised to update-groups:
     4
  Refresh Epoch 1
  65000, (aggregated by 65000 192.168.11.1)
    10.102.77.102 (via vrf TENANT77) from 10.102.77.102 (192.168.11.1)
      Origin IGP, localpref 100, weight 999, valid, external, atomic-aggregate,
best
      Community: 64577:999
      Extended Community: RT:65077:1
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  65000, (aggregated by 65000 192.168.11.1)
    10.103.78.103 (via vrf TENANT77) from 10.103.78.103 (192.168.11.1)
      Origin IGP, localpref 100, weight 9, valid, external, atomic-aggregate
      Community: 64577:9
      Extended Community: RT:65077:1
      rx pathid: 0, tx pathid: 0
```

Example 10-31: *BGP table entry of Ext-Ro01 about the network 192.168.11.11.*

Example 8-32 shows the BRIB of Ext-Ro02.

```
Ext-Ro02#sh ip bgp vpnv4 vrf TENANT77 192.168.11.0
BGP routing table entry for 65077:1:192.168.11.0/24, version 15
Paths: (2 available, best #1, table TENANT77)
  Advertised to update-groups:
     4
  Refresh Epoch 1
  65000, (aggregated by 65000 192.168.11.1)
    10.102.78.102 (via vrf TENANT77) from 10.102.78.102 (192.168.11.1)
      Origin IGP, localpref 100, weight 999, valid, external, atomic-aggregate,
best
      Community: 64577:999
      Extended Community: RT:65077:1
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
```

```
  65000, (aggregated by 65000 192.168.11.1)
    10.103.77.103 (via vrf TENANT77) from 10.103.77.103 (192.168.11.1)
      Origin IGP, localpref 100, weight 9, valid, external, atomic-aggregate
      Community: 64577:9
      Extended Community: RT:65077:1
      rx pathid: 0, tx pathid: 0
```

Example 10-32: *BGP table entry of Ext-Ro01 about the network 192.168.11.11.*

Step-4: Define the incoming BGP policy in Border Leaf switches. Set weight 999 for the network 172.16.3/24 received from Ext-Ro01 and weigh 9 received from Ext-Ro02.

Step-4.1 Define the prefix-list that permits network 172.16.3.0/24
Step-4.2. Define the route-map that match the prefix-list and sets the weight 999 for routes received from Ext-Ro01 and weight 9 for routes received from Ext-Ro02.

```
router bgp 65000
  router-id 192.168.77.102
  timers bgp 3 9
  address-family ipv4 unicast
  address-family l2vpn evpn
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 192.168.11.0/24 summary-only
    neighbor 10.102.77.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.77
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map INCOMING_POLICIES_FROM_ExtRo01 in
        route-map OUTGOING_POLICIES out
    neighbor 10.102.78.2
      remote-as 64577
      description ** External Network - Ext-Ro02 **
      update-source Ethernet1/3.78
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map INCOMING_POLICIES_FROM_ExtRo02 in
        route-map OUTGOING_POLICIES out
```

```
!
ip prefix-list EXTERNAL_GROUP_1 seq 10 permit 172.16.3.0/24
!
route-map INCOMING_POLICIES_FROM_ExtRo01 permit 10
  match ip address prefix-list EXTERNAL_GROUP_1
  set weight 999
route-map INCOMING_POLICIES_FROM_ExtRo01 permit 100
!
route-map INCOMING_POLICIES_FROM_ExtRo02 permit 10
  match ip address prefix-list EXTERNAL_GROUP_1
  set weight 9
route-map INCOMING_POLICIES_FROM_ExtRo02 permit 100
```

Example 10-33: *Ingress BGP policy on Border Leaf-102.*

The same logic is applied to Border Leaf-103 (Example 10-34).

```
router bgp 65000
  router-id 192.168.77.103
  timers bgp 3 9
  address-family ipv4 unicast
  address-family l2vpn evpn
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
      aggregate-address 192.168.11.0/24 summary-only
    neighbor 10.103.77.2
      remote-as 64577
      description ** External Network - Ext-Ro02 **
      update-source Ethernet1/3.77
      address-family ipv4 unicast
        send-community
        send-community extended
        route-map INCOMING_POLICIES_FROM_ExtRo02 in
        route-map OUTGOING_POLICIES out
    neighbor 10.103.78.1
      remote-as 64577
      description ** External Network - Ext-Ro01 **
      update-source Ethernet1/4.78
      address-family ipv4 unicast
        send-community
        route-map INCOMING_POLICIES_FROM_ExtRo01 in
        route-map OUTGOING_POLICIES out
```

```
!
ip prefix-list EXTERNAL_GROUP_1 seq 10 permit 172.16.3.0/24
!
route-map INCOMING_POLICIES_FROM_ExtRo01 permit 10
  match ip address prefix-list EXTERNAL_GROUP_1
  set weight 999
route-map INCOMING_POLICIES_FROM_ExtRo01 permit 100
!
route-map INCOMING_POLICIES_FROM_ExtRo02 permit 10
  match ip address prefix-list EXTERNAL_GROUP_1
  set weight 9
route-map INCOMING_POLICIES_FROM_ExtRo02 permit 100
```

Example 10-34: *Ingress BGP policy on Border Leaf-103.*

As can be seen from the Example 10-35 eBGP update with the highest Weight has been chosen to the best path to network 172.16.3.0/24 in both Border Leaf switches (Examples 10-35 and 10-36).

```
Leaf-102# sh ip bgp vrf TENANT77
<snipped>
   Network            Next Hop            Metric   LocPrf     Weight Path
* i172.16.1.0/24      192.168.100.103          0   100          0 64577 i
*>e                   10.102.77.1              0                0 64577 i
* i172.16.3.0/24      192.168.100.103          2   100          0 64577 i
*>e                   10.102.77.1              2              999 64577 i
* e                   10.102.78.2              2                9 64577 i
* i192.168.11.0/24    192.168.100.103              100          0 i
*>a                   0.0.0.0                      100      32768 i
s>i192.168.11.11/32   192.168.100.101                   100          0 i
```

Example 10-35: *BGP table on Border Leaf-102.*

```
Leaf-103# sh ip bgp vrf TENANT77
<snipped>
   Network            Next Hop            Metric   LocPrf   Weight Path
* i172.16.1.0/24      192.168.100.102          0   100          0 64577 i
*>e                   10.103.78.1              0                0 64577 i
* i172.16.3.0/24      192.168.100.102          2   100          0 64577 i
* e                   10.103.77.2              2                9 64577 i
*>e                   10.103.78.1              2              999 64577 i
* i192.168.11.0/24    192.168.100.102              100          0 i
*>a                   0.0.0.0                      100      32768 i
s>i192.168.11.11/32   192.168.100.101              100          0 i
```

Example 10-36: *BGP table on Border Leaf-103.*

Now the BGP configuration is ready. Example 10-37 shows that there is an IP connectivity between 192.168.11.11 and 172.16.3.1.

```
Cafe#traceroute 172.16.3.1
Type escape sequence to abort.
Tracing the route to 172.16.3.1
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.11.1 5 msec 5 msec 8 msec
  2 10.102.77.102 17 msec 31 msec 21 msec
  3 10.102.77.1 26 msec 9 msec 21 msec
  4 10.1.3.3 25 msec 18 msec *
Cafe#
```

Example 10-37: *Trace from 192.168.11.11 to 172.16.3.1*

```
Ext-Ro03#traceroute vrf TENANT77 192.168.11.11 source 172.16.3.1
Type escape sequence to abort.
Tracing the route to 192.168.11.11
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.3.1 4 msec 7 msec 4 msec
  2 10.102.77.102 8 msec 7 msec 3 msec
  3 192.168.11.1 11 msec 16 msec 25 msec
  4 192.168.11.11 27 msec 16 msec *
Ext-Ro03#
```

Example 10-38: *Trace from 172.16.3.1 to 192.168.11.11*

As can be seen from the traceroute examples, the routing is symmetric and the path goes as expected.

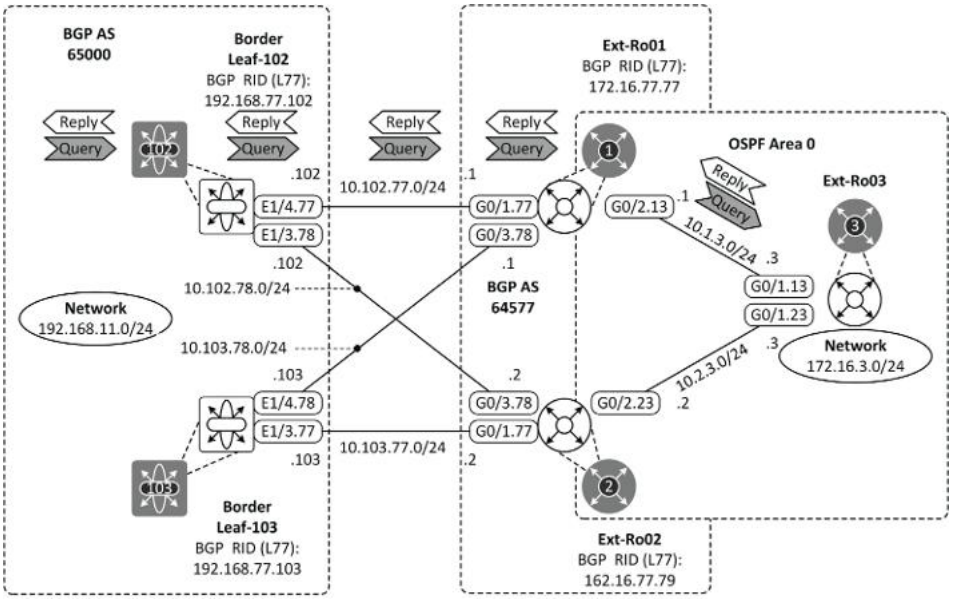


Figure 10-10: *trace test#1*

Now the interface g0/1.77 is set to downstate on Ext-Ro01. This means that based on the policy, the trace from left to right should use the path Leaf-102 > Ext-Ro02 > Ext-Ro03 and trace from right to left should use the path Ext-Ro01 > Border Leaf-103.

As can be seen from the Examples 8-39 and 8-40, the network converges as expected.

```
Cafe#traceroute 172.16.3.1 source 192.168.11.11
Type escape sequence to abort.
Tracing the route to 172.16.3.1
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.11.1 13 msec 6 msec 5 msec
  2 10.102.77.102 12 msec 19 msec 14 msec
  3 10.102.78.2 23 msec 22 msec 18 msec
  4 10.2.3.3 26 msec 21 msec *
Cafe#
```

Example 10-39: *Trace from 192.168.11.11 to 172.16.3.1*

```
Ext-Ro03#traceroute vrf TENANT77 192.168.11.11 source 172.16.3.1
Type escape sequence to abort.
Tracing the route to 192.168.11.11
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.3.1 13 msec 4 msec 4 msec
  2 10.103.78.103 6 msec 4 msec 8 msec
  3 192.168.11.1 35 msec 15 msec 10 msec
  4 192.168.11.11 18 msec 66 msec *
Ext-Ro03#
```

Example 10-40: *Trace from 172.16.3.1 to 192.168.11.11*

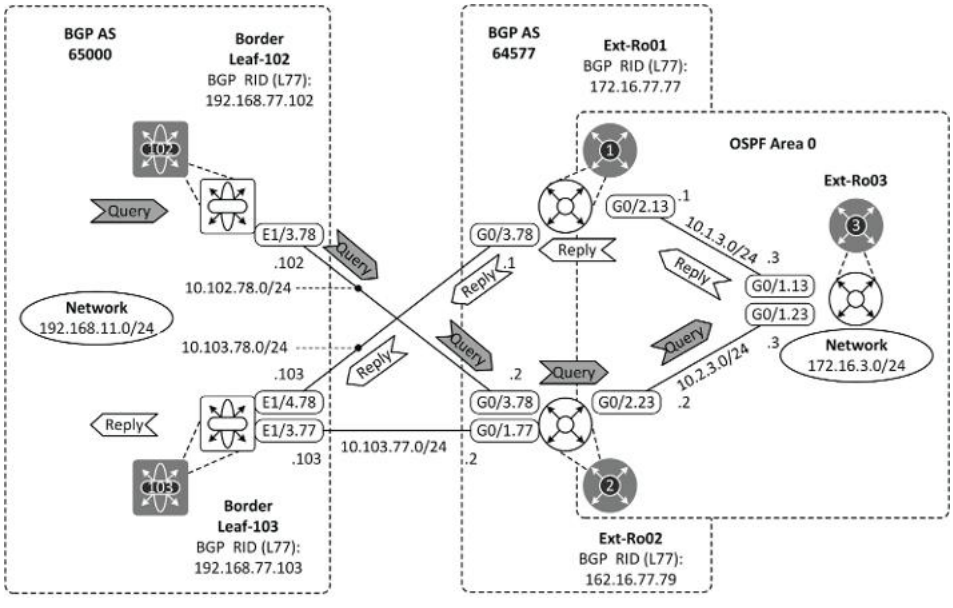


Figure 10-11: *trace test#2*

Next, the interface g0/3.78 is set to down on Ext-Ro01. Then both paths from right to left and from left to right start using the path through the Border Leaf-102 and Ext-Ro02.

```
Cafe#traceroute 172.16.3.1 source 192.168.11.11
Type escape sequence to abort.
Tracing the route to 172.16.3.1
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.11.1 3 msec 20 msec 9 msec
  2 10.102.77.102 38 msec 6 msec 8 msec
  3 10.102.78.2 15 msec 14 msec 20 msec
  4 10.2.3.3 43 msec 20 msec *
Cafe#
```

Example 10-41: *Trace from 192.168.11.11 to 172.16.3.1*

```
Ext-Ro03#traceroute vrf TENANT77 192.168.11.11 source 172.16.3.1
Type escape sequence to abort.
Tracing the route to 192.168.11.11
VRF info: (vrf in name/id, vrf out name/id)
  1 10.2.3.2 9 msec 9 msec 2 msec
  2 10.102.78.102 3 msec 36 msec 5 msec
  3 192.168.11.1 34 msec 10 msec 6 msec
  4 192.168.11.11 16 msec 20 msec *
Ext-Ro03#
```

Example 10-42: *Trace from 172.16.3.1 to 192.168.11.11*

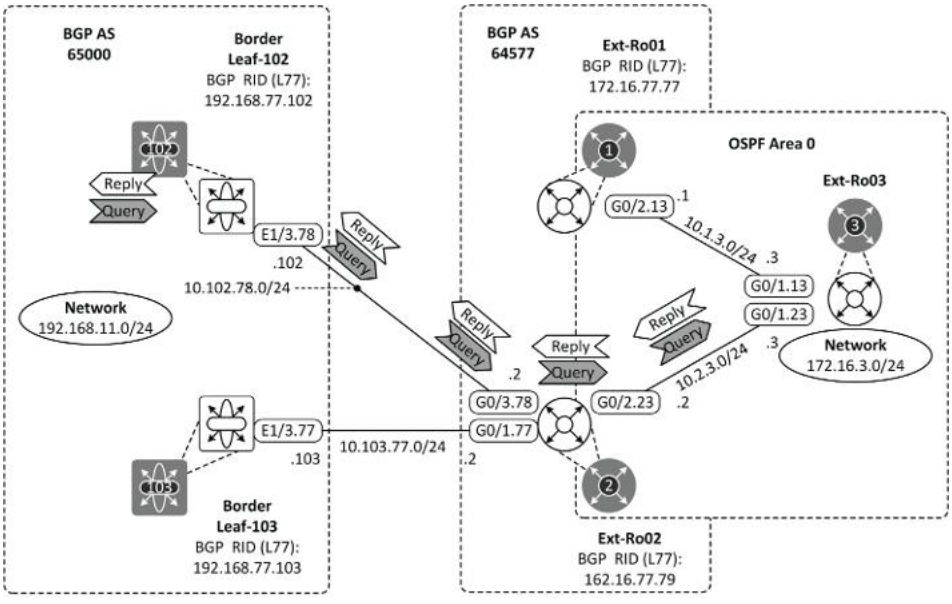


Figure 10-12: *trace test#3*

Next, the interface g0/3.78 is set to down on Ext-Ro02 to make sure that path moves from the Border Leaf-102 to Border Leaf-103.

```
Cafe#traceroute 172.16.3.1 source 192.168.11.11
Type escape sequence to abort.
Tracing the route to 172.16.3.1
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.11.1 9 msec 14 msec 23 msec
  2 10.103.77.103 28 msec 10 msec 11 msec
  3 10.103.77.2 12 msec 20 msec 17 msec
  4 10.2.3.3 13 msec 15 msec *
Cafe#
```

Example 10-43: *Trace from 192.168.11.11 to 172.16.3.1*

```
Ext-Ro03#traceroute vrf TENANT77 192.168.11.11 source 172.16.3.1
Type escape sequence to abort.
Tracing the route to 192.168.11.11
VRF info: (vrf in name/id, vrf out name/id)
  1 10.2.3.2 8 msec 3 msec 4 msec
  2 10.103.77.103 8 msec 7 msec 3 msec
  3 192.168.11.1 15 msec 11 msec 11 msec
  4 192.168.11.11 29 msec 17 msec *
Ext-Ro03#
```

Example 10-44: *Trace from 172.16.3.1 to 192.168.11.11*

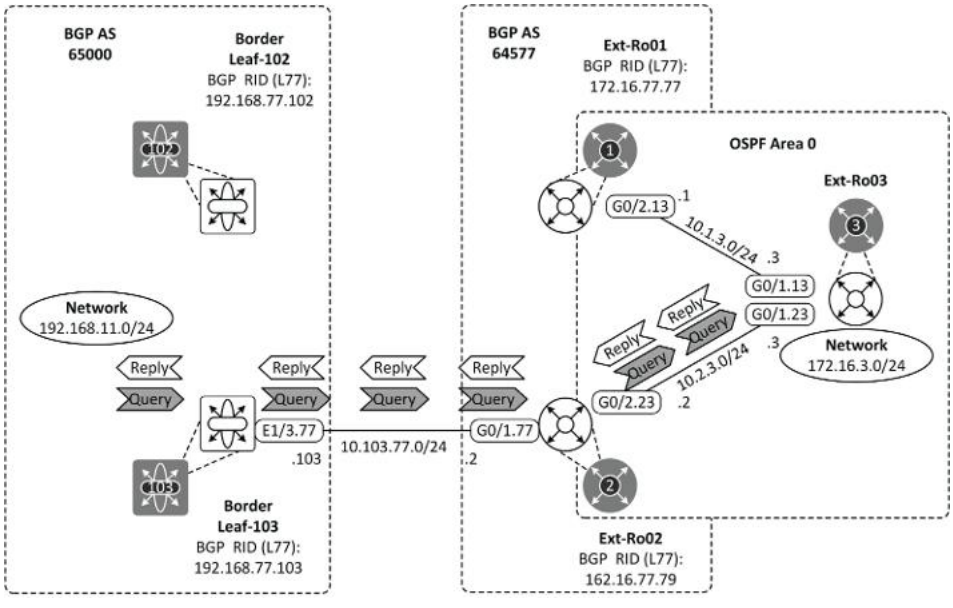


Figure 10-13: *trace test#4*

Network converge as expected.

Chapter 11: Multihoming with vPC

This chapter describes how the Multi-Chassis Link Aggregation Group (MC-LAG) technology using virtual PortChannel (vPC) works in a VXLAN BGP EVPN fabric. It first discusses vPC configuration with a short explanation and then explains the Control- and Data Plane operation from the VXLAN BGP EVPN perspective by using various show commands and packet capture. It also explains the "Advertising VIP/PIP" options using the external connection. An example topology is shown in Figure 11-1.

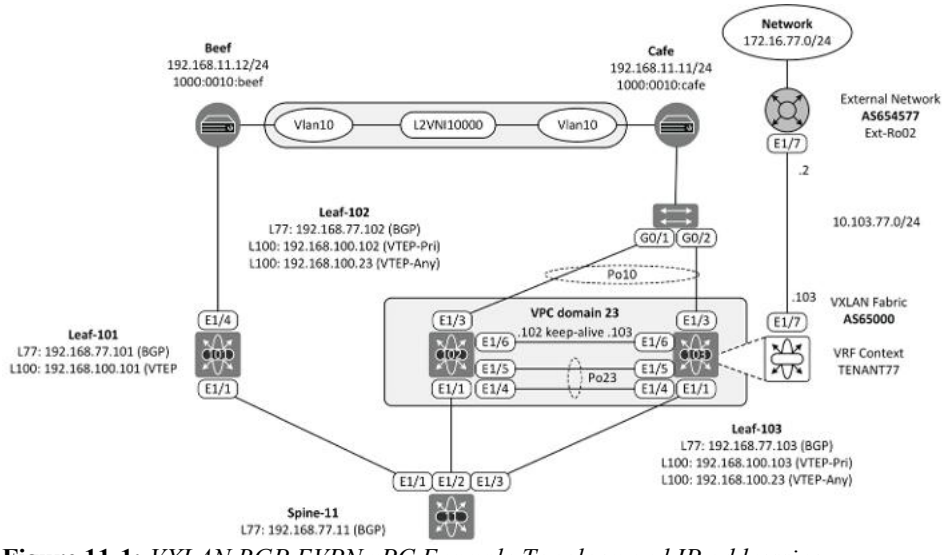


Figure 11-1: *VXLAN BGP EVPN vPC Example Topology and IP addressing*

Virtual Port Channel Configuration

In this example, there are two vPC VTEP switches Leaf-102 and Leaf-103. Inter-switch links, vPC related link terms and IP addressing and PortChannel numbering can be seen from the figure 11-2.

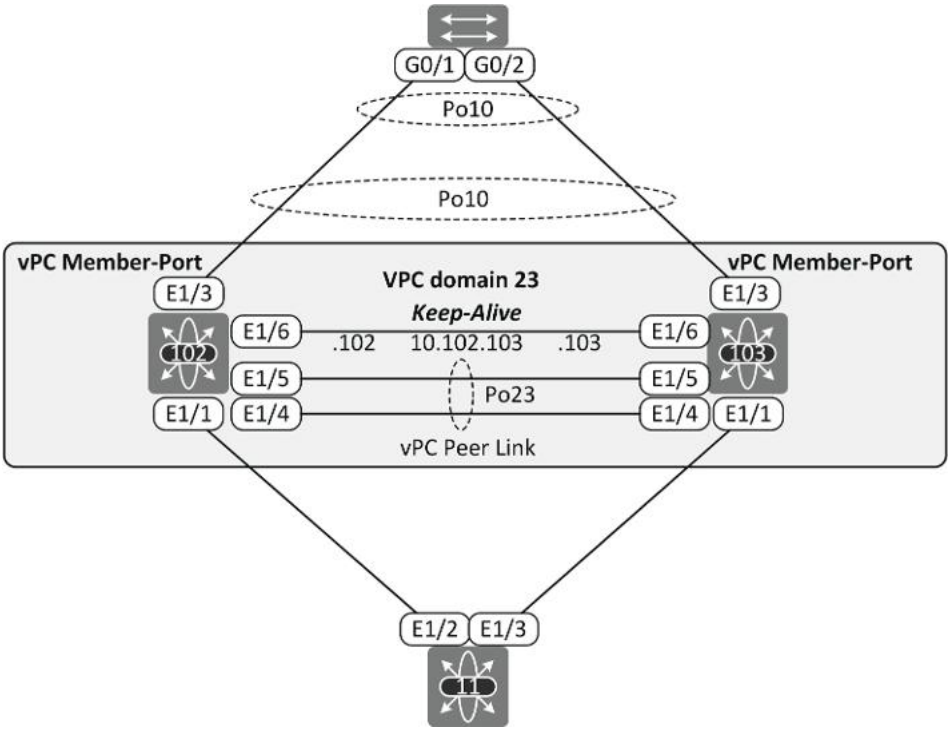


Figure 11-2: *vPC domain*

Step 1: Enable vPC and LACP features on both vPC VTEP switches.

```
feature vpc
feature lacp
```

Example 11-1: *vPC and LACP features*

Step 2: Configure the Peer-Keepalive link.

VPC Peer-Keepalive link is used as a heartbeat link between the vPC peers to make sure that both vPC peers are alive. There is a dedicated VRF "VPC-Peer-Keepalive" for the vPC Peer-Keepalive link. Example 11-2 shows the configuration of vPC on VTEP Leaf-102.

```
vrf context VPC-Peer-Keepalive
!
interface Ethernet1/6
  no switchport
  vrf member VPC-Peer-Keepalive
  ip address 10.102.103.102/24
  no shutdown
!
vpc domain 23
  peer-keepalive destination 10.102.103.103 source 10.102.103.102 vrf VPC-Peer-
Keepalive
```

Example 11-2: *vPC Peer-Keepalive (Leaf-102)*

Step 2.1: Verify Peer-Keepalive link operation

```
Leaf-102# show vpc peer-keepalive

vPC keep-alive status             : peer is alive
--Peer is alive for              : (685) seconds, (480) msec
--Send status                    : Success
--Last send at                   : 2018.08.11 09:38:44 791 ms
--Sent on interface              : Eth1/6
--Receive status                 : Success
--Last receive at                : 2018.08.11 09:38:45 314 ms
--Received on interface          : Eth1/6
--Last update from peer          : (0) seconds, (293) msec

vPC Keep-alive parameters
--Destination                    : 10.102.103.103
--Keepalive interval             : 1000 msec
--Keepalive timeout              : 5 seconds
--Keepalive hold timeout         : 3 seconds
--Keepalive vrf                  : VPC-Peer-Keepalive
--Keepalive udp port             : 3200
--Keepalive tos                  : 192
```

Example 11-3: *vPC Peer-Keepalive (Leaf-102) status check*

Note! vPC domain 23 was created in step 2. VPC peer switches will automatically create a unique vPC system MAC address. The vPC system MAC address has a fixed part = 0023.04ee.be.xx and the two last digits (xx) are taken from vPC domain ID. Our example vPC domain has ID 23, which HEX format is 17. So the vPC system address in our example will be 0023.04ee.be17. This can be verified from both switches. As can be seen from the examples 11-4 and 11-5 there are also vPC local system-mac. The vPC system MAC is common for both vPC peer switches and it is represented when the vPC system, formed by two vPC peer switches, represents itself as a unit. The vPC local system-mac is unique per vPC peer switch and it is used when switch presents itself as an individual switch, not as a vPC system. This is the case with Orphan ports for example.

```
Leaf-102# sh vpc role

vPC Role status
----------------------------------------------------
vPC role                          : primary
Dual Active Detection Status      : 0
vPC system-mac                    : 00:23:04:ee:be:17
vPC system-priority               : 32667
vPC local system-mac              : 5e:00:00:01:00:07
vPC local role-priority           : 32667
vPC local config role-priority    : 32667
vPC peer system-mac               : 5e:00:00:06:00:07
vPC peer role-priority            : 32667
vPC peer config role-priority     : 32667
```

Example 11-4: *vPC system MAC Leaf-102*

```
Leaf-103# sh vpc role

vPC Role status
----------------------------------------------------
vPC role                          : secondary
Dual Active Detection Status      : 0
vPC system-mac                    : 00:23:04:ee:be:17
vPC system-priority               : 32667
vPC local system-mac              : 5e:00:00:06:00:07
vPC local role-priority           : 32667
vPC local config role-priority    : 32667
vPC peer system-mac               : 5e:00:00:01:00:07
vPC peer role-priority            : 32667
vPC peer config role-priority     : 32667
```

Example 11-5: *vPC system MAC Leaf-103*

Step 3: Create vPC Peer-Link

VPC Peer-Link is an 802.1Q trunk link that carries vPC and non-vPC VLANs, Cisco Fabric Service Messages (consistency check, MAC address synchronization, advertisement of vPC member port status, STP management and synchronization of HSRP and IGMP snooping), flooded traffic from the peer vPC, STP BPDUs, HSRP Hello messages and IGMP updates. In this example, Port-Channel 23 is configured for the vPC peer-link with LACP as a channel protocol.

```
interface port-channel23
  switchport mode trunk
  spanning-tree port type network
  vpc peer-link
!
interface Ethernet1/5
  description ** Po23 member - vPC PEER-link **
  switchport mode trunk
  channel-group 23 mode active
!
interface Ethernet1/5
  description ** Po23 member - vPC PEER-link **
  switchport mode trunk
  channel-group 23 mode active
```

Example 11-6: *vPC Peer-Link on switch Leaf-102*

Note! If the vPC Peer-Link goes down while vPC Peer-Keepalive link is still up, the secondary switch suspends its vPC member port and shuts down the SVI associated to the vPC VLAN. Once this failure happens, Orphan ports connected to Secondary switch will be isolated. That is the reason for the recommendation to connect Orphan hosts to Primary switch.

Step 4: Configure vPC Member Ports

From the access device perspective, Ethernet switch in this example, the uplink port is a classical Ether-Channel while from the vPC VTEP point of view the link to access device is attached to a vPC Member Port. The recommended mode for a channel protocol is Link Aggregation Control Protocol (LACP) because it is a standard protocol and it has built-in misconfiguration protection and fast failure detection mechanism, though the example use static configuration.

```
interface port-channel10
  switchport mode trunk
  vpc 10
!
interface Ethernet1/3
  description ** Link to Ethernet SW **
  switchport mode trunk
  channel-group 10
```

Example 11-7: *vPC Member Port on Leaf-102 and Leaf-103*

Step 5: Verification of vPC operational status.

```
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 23
Peer status                       : peer adjacency formed ok
vPC keep-alive status             : peer is alive
Configuration consistency status  : success
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : primary
Number of vPCs configured         : 1
Peer Gateway                      : Enabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Disabled
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled

vPC Peer-link status
---------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ --------------------------------------------------
1     Po23   up     1,10,20,77


vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
10    Po10          up     success     success               1,10,20,77
```

Example 11-8: *vPC verification*

Step 6: Configure vPC peer-gateway under vpc domain configuration

Some devices, such as NAS and Load Balancer, might not perform standard ARP-request for IP of the default gateway during the boot process. Instead of it, they take the first source MAC address that they hear from the wire and then bind that MAC address with the IP address of the default gateway. This kind of behavior might cause forwarding problems.

In figure 11-3, there are two hosts, Host-A in VLAN 10 and Host-B in VLAN 20. Let's say that Host-B is a NAS device that binds the first source MAC that it hears, to the default gateway IP. (1) vPC VTEP Leaf-102 sends some data towards host-B. (2) Host-B has just booted up and it receives the frame sent from Leaf-103 and binds the source MAC address from the frame to the default gateway IP. (3) Then Host-B starts sending data to Host-A which is in VLAN 10, so the IP packet is sent to the default gateway. (4) The Ethernet switch, where the Host-B is connected, receives the IP packet and runs the channel hash algorithm and choose the link towards vPC VTEP Leaf-103. (5) Leaf-103 receives the IP packet and since the destination MAC address in Ethernet header belongs to Leaf-102, the IP packet is sent over the vPC Peer-link to the vPC VTEP peer Leaf-102. (6) Now the loop prevention mechanism kicks in, the data received from the vPC member port and the crossing over vPC peer-link is not allowed to send out any vPC member port. So in this case, Leaf-102 drops the data packet.

Note! There is one exception to loop prevention mechanism "frame received from vPC member ports and crossed over vPC Peer-Link are not allowed to egress from vPC member port". If the vPC member port between Leaf-103 and host A is down, then the frame is allowed to egress from Leaf-102 port e1/3.

By using the vPC Peer-Gateway option, Leaf-103 is allowed to act as an active default gateway in VLAN 10 (and of course in VLAN 20) also in a situation where the IP packet received over the vPC member port has a destination MAC address, that belongs to the vPC peer Leaf-102. So in our example, Leaf-103 is allowed to send data packet straight to Host-A without sending it to vPC peer Leaf-102.

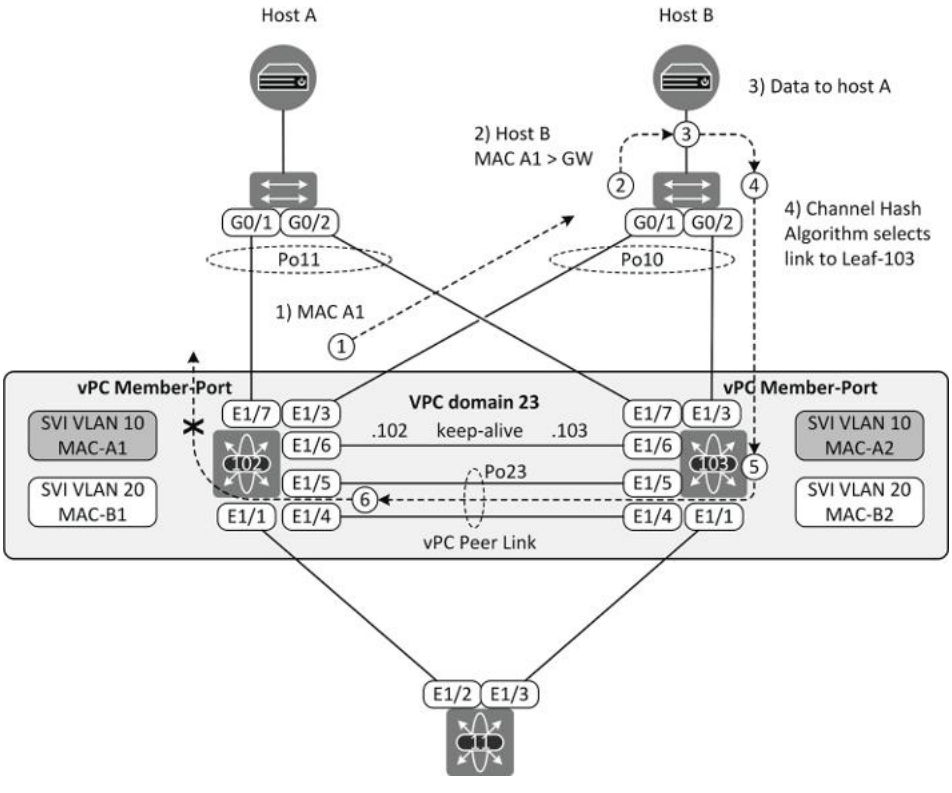


Figure 11-3: *vPC peer-gateway*

```
Vpc domain 23
  Peer-gateway
```

Example 11-9: *vPC peer-gateway configuration*

Step 7: Configure ARP sync under vpc domain configuration

ARP sync is used to synchronize the MAC address information in recovery situation after the vPC Peer-link has failed. Synchronization is done by using the Cisco Fabric Service protocol (CFS). The direction is from the primary vPC peer (Leaf-102 in our lab) to the secondary vPC peer (Leaf-103).

```
Vpc domain 23
  ip arp synchronize
```

Example 11-10: *vPC ARP synch configuration*

Step 6: Tune vPC Delay Restore timers (optional)

By using vPC delay restore, the vPC peer switch holds down the vPC links and SVIs until the routing protocols are converged. This property is enabled by default with a timer value 30 and 10 seconds (vPC link/SVI). We are using timers 240/80. These values are related to the size of the network.

```
vpc domain 23
  delay restore 240
  delay restore interface-vlan 80
```

Example 11-11: *vPC Delay Restore configuration*

Some other consideration for vPC:

Since the primary subject of this chapter is to show how the VXLAN BGP EVPN works with vPC this section does not cover each and every vPC feature in detail but still, here are some other considerations when implementing vPC.

The vPC priority should be statically defined in both primary and secondary vPC peer switch. This way we know which one is the primary switch. Orphan hosts should be connected to the primary vPC peer switch. By doing this they are not restricted from the network in case of vPC Peer-Link failure. At this moment, there is a following vPC configuration on switch Leaf-102 and Leaf-103.

```
feature vpc
!
vpc domain 23
  peer-switch
  peer-keepalive destination 10.102.103.103 source 10.102.103.102 vrf VPC-Peer-
Keepalive
  delay restore 240
  peer-gateway
  delay restore interface-vlan 80
  ip arp synchronize
!
interface port-channel10
  vpc 10
!
interface port-channel23
  vpc peer-link
!
interface Ethernet1/6
  no switchport
  vrf member VPC-Peer-Keepalive
  ip address 10.102.103.102/24
  no shutdown
!
interface Ethernet1/4
  description ** Po23 member - vPC PEER-link **
  switchport mode trunk
  channel-group 23 mode active
!
interface Ethernet1/5
  description ** Po23 member - vPC PEER-link **
  switchport mode trunk
  channel-group 23 mode active
!
interface Ethernet1/3
  description ** Link to Ethernet SW **
  switchport mode trunk
  channel-group 10
```

Example 11-12: *vPC Delay Restore configuration*

VTEP redundancy with vPC

When vPC is implemented into VXLAN fabric, both vPC VTEP peers start using a Virtual IP (VIP) address as a source address instead of their physical IP address (PIP). This also means that BGP EVPN starts advertising both Route Types 2 (MAC/IP advertisement) and 5 (IP prefix-route) with VIP as a next-hop (default behavior). There are two IP addresses configured into Loopback 0 interface, Primary IP 192.168.100.102/32 (PIP) and secondary IP 192.168.100.23/32 (VIP) in our example lab (figure 11-4).

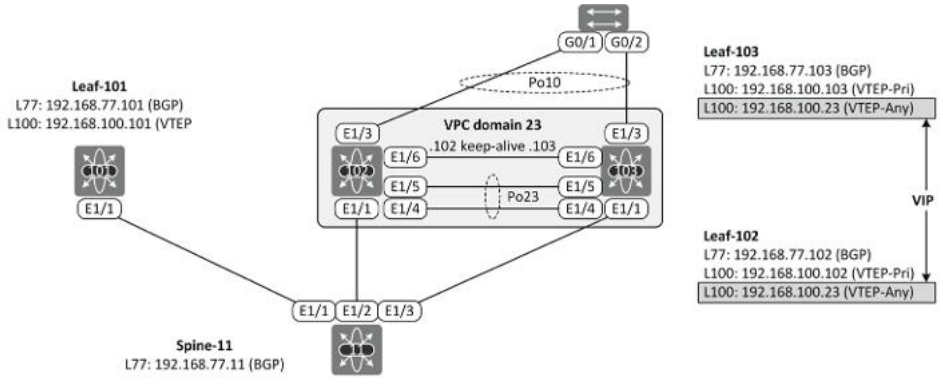


Figure 11-4: *vPC PIP and VIP addressing*

First, configure the same secondary IP address 192.168.100.23 under the Loopback 0 interface on both vPC VTEP switches. An example is taken from VTEP-102. At this phase, that is the only change for the current configuration.

```
interface loopback100
  description ** VTEP/Overlay **
  ip address 192.168.100.102/32
  ip address 192.168.100.23/32 secondary
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
```

Example 11-13: *Secondary IP address into Loopback 0*

Next, the host Cafe joins the network. The Control Plane operation is verified by capturing traffic from wire to see the BGP EVPN MAC/IP advertisements. Next, the Data Plane operation is verified by pinging from Cafe to Beef.

Phase-1: Host Cafe boot up and sends a Gratuitous ARP message to verify the uniqueness of its IP address and inform the location of its own MAC address. Channel hash algorithm selects the interface g0/1 and the broadcast messages are sent out of it.

Phase-2: Leaf-102 receives the first broadcast frame. Its L2FWDER component notices the incoming frame from the interface Po10, with source MAC address 1000.0010.cafe. After the MAC address-table update the L2FWDER installs the MAC address into L2RIB where it is sent to the BGP EVPN process. Since LEAF-102 has vPC peer switch Leaf-103, it synchronizes the MAC address-table over the vPC Peer-Link with CFS. This way Leaf-103 learns that the MAC address 1000.0010.cafe is located behind the PortChannel 10. Since the destination of the frame is a broadcast address, it is also flooded over the vPC Peer-link. These broadcast messages are also sent to the corresponding multicast group of VNI 10000.

Phase-3: At this phase, the L2FWDER component has sent MAC address information from L2RIB to the BGP EVPN process in both switches. They both send two BGP EVPN Route Type-2 Update to the Spine switch Spine-11, first one is the host Cafe MAC-address and the second on is MAC/IP information. For simplicity, we concentrate only on MAC address advertisements sent by vPC peer switches Leaf-102 and Leaf-103. The BGP EVPN Update messages can be seen in Capture 11-1 (Leaf-102) and 11-2 (Leaf-103) right after the figure 11-5. From these captures, we can see that the Path Attribute MP_REACH_NLRI Path Attribute Next-Hop is set 192.168.100.23. This information is not visible in the captured packet as binary mode, but it can be found from the HEX part, where we can see the HEX value c0 a8 64 17 (192.168.100.23 in Binary). Note that the EVPN NLRI Route Distinguisher includes the original sender RID which is how the Spine switch can differentiate these updates.

Phase-4: Spine-11 sends these BGP EVPN Updates to Leaf-101 without modification of the Path Attribute, it just adds a Cluster List Path Attribute, which is used as a loop prevention mechanism (Spine-11 is a Route-Reflector). If we compare these updates received from Leaf-102 and Leaf-103, the only notable difference in BGP EVPN Update is the Route Distinguisher (RD). By checking the RD value, the Spine-11 knows that updates are from different Leaf.

Phase-5: Capture 11-3, taken from the Leaf-103 interface e1/1, shows that the RR Spine-11 sends the BGP EVPN Update about host Cafe MAC, sent by Leaf-102, to Leaf-103. This Update is blocked by Leaf-103 based Site of Origin (SoO) Extended Community Attribute 192.168.100.23:0 (Route Origin field in Capture).

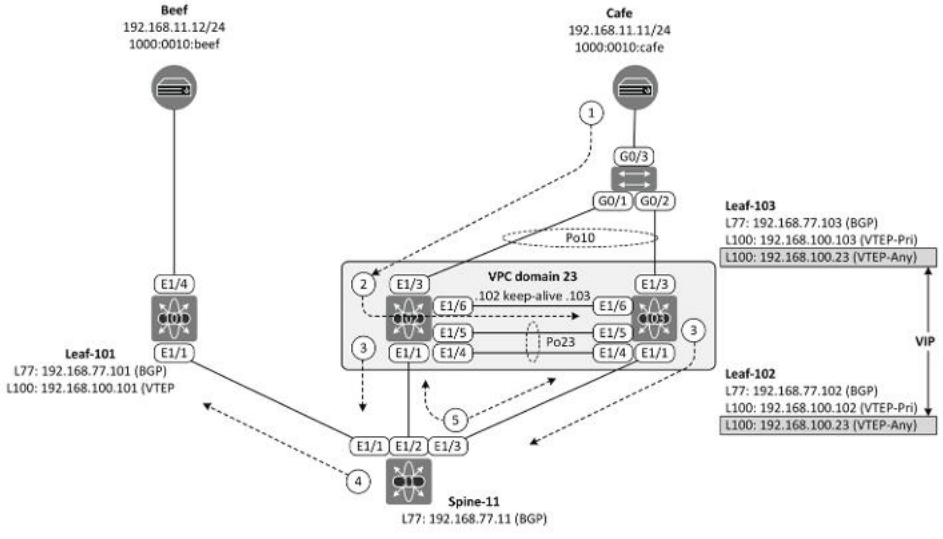


Figure 11-5: *BGP EVPN Update*

```
Ethernet II, Src: 5e:00:00:01:00:07 (5e:00:00:01:00:07), Dst: 5e:00:00:02:00:07
(5e:00:00:02:00:07)
Internet Protocol Version 4, Src: 192.168.77.102, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 55415, Dst Port: 179, Seq: 630, Ack:
644, Len: 112
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 112
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 89
    Path attributes
        Path Attribute - ORIGIN: IGP
            Flags: 0x40, Transitive, Well-known, Complete
            Type Code: ORIGIN (1)
            Length: 1
            Origin: IGP (0)
        Path Attribute - AS_PATH: empty
            Flags: 0x40, Transitive, Well-known, Complete
            Type Code: AS_PATH (2)
            Length: 0
        Path Attribute - LOCAL_PREF: 100
            Flags: 0x40, Transitive, Well-known, Complete
            Type Code: LOCAL_PREF (5)
            Length: 4
            Local preference: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 24
            Carried extended communities: (3 communities)
                Route Target: 65000:10000
```

```
                Route Origin: 192.168.100.23:0
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive
            Type Code: MP_REACH_NLRI (14)
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 33
                    Route Distinguisher: (192.168.77.102:32777)
                    ESI: 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: Private_10:ca:fe (10:00:00:10:ca:fe)
                    IP Address Length: 0
                    IP Address: NOT INCLUDED
                    MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)

0000  5e 00 00 02 00 07 5e 00 00 01 00 07 08 00 45 c0   ^.....^.......E.
0010  00 a4 4d 0a 00 00 40 06 10 c8 c0 a8 4d 66 c0 a8   ..M...@.....Mf..
0020  4d 0b d8 77 00 b3 16 3c 58 90 62 4d 92 9a 80 18   M..w...<X.bM....
0030  0f 4e e9 14 00 00 01 01 08 0a 00 14 b7 ad 00 14   .N..............
0040  cf d2 ff ff ff ff ff ff ff ff ff ff ff ff ff ff   ................
0050  ff ff 00 70 02 00 00 00 59 40 01 01 00 40 02 00   ...p....Y@...@..
0060  40 05 04 00 00 00 64 c0 10 18 00 02 fd e8 00 00   @.....d.........
0070  27 10 01 03 c0 a8 64 17 00 00 03 0c 00 00 00 00   '.....d.........
0080  00 08 90 0e 00 2c 00 19 46 04 c0 a8 64 17 00 02   .....,..F...d...
0090  21 00 01 c0 a8 4d 66 80 09 00 00 00 00 00 00 00   !....Mf.........
00a0  00 00 00 00 00 00 00 30 10 00 00 10 ca fe 00 00   .......0........
00b0  27 10
```

Capture 11-1: *BGP EVPN Update from Leaf-102 to Spine-11*

```
Ethernet II, Src: 5e:00:00:06:00:07 (5e:00:00:06:00:07), Dst: 5e:00:00:02:00:07
(5e:00:00:02:00:07)
Internet Protocol Version 4, Src: 192.168.77.103, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 40773, Dst Port: 179, Seq: 573, Ack:
732, Len: 112
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 112
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 89
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 24
            Carried extended communities: (3 communities)
                Route Target: 65000:10000
                Route Origin: 192.168.100.23:0
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive
```

```
            Type Code: MP_REACH_NLRI (14)
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 33
                    Route Distinguisher: 192.168.77.103:32777
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: Private_10:ca:fe (10:00:00:10:ca:fe)
                    IP Address Length: 0
                    IP Address: NOT INCLUDED
                    MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)

0000  5e 00 00 02 00 07 5e 00 00 06 00 07 08 00 45 c0   ^.....^.......E.
0010  00 a4 58 87 00 00 40 06 05 4a c0 a8 4d 67 c0 a8   ..X...@..J..Mg..
0020  4d 0b 9f 45 00 b3 3d ba 4d b4 2f ff c2 c2 80 18   M..E..=.M./.....
0030  0f 4e 0a a8 00 00 01 01 08 0a 00 14 b3 7c 00 14   .N...........|..
0040  d0 25 ff ff ff ff ff ff ff ff ff ff ff ff ff ff   .%..............
0050  ff ff 00 70 02 00 00 00 59 40 01 01 00 40 02 00   ...p....Y@...@..
0060  40 05 04 00 00 00 64 c0 10 18 00 02 fd e8 00 00   @.....d.........
0070  27 10 01 03 c0 a8 64 17 00 00 03 0c 00 00 00 00   '.....d.........
0080  00 08 90 0e 00 2c 00 19 46 04 c0 a8 64 17 00 02   .....,..F...d...
0090  21 00 01 c0 a8 4d 67 80 09 00 00 00 00 00 00 00   !....Mg.........
00a0  00 00 00 00 00 00 00 30 10 00 00 10 ca fe 00 00   .......0........
00b0  27 10                                            '.
```

Capture 11-2: *BGP EVPN Update from Leaf-103 to Spine-11*

```
Frame 131: 192 bytes on wire (1536 bits), 192 bytes captured (1536 bits)
Ethernet II, Src: 5e:00:00:02:00:07 (5e:00:00:02:00:07), Dst: 5e:00:00:06:00:07
(5e:00:00:06:00:07)
Internet Protocol Version 4, Src: 192.168.77.11, Dst: 192.168.77.103
Transmission Control Protocol, Src Port: 179, Dst Port: 40773, Seq: 606, Ack:
573, Len: 126
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 126
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 103
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 24
            Carried extended communities: (3 communities)
        Path Attribute - ORIGINATOR_ID: 192.168.77.102
            Flags: 0x80, Optional, Non-transitive, Complete
            Type Code: ORIGINATOR_ID (9)
            Length: 4
            Originator identifier: 192.168.77.102
        Path Attribute - CLUSTER_LIST: 192.168.77.111
            Flags: 0x80, Optional, Non-transitive, Complete
            Type Code: CLUSTER_LIST (10)
            Length: 4
```

```
            Cluster List: 192.168.77.111
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive, Complete
            Type Code: MP_REACH_NLRI (14)
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 33
                    Route Distinguisher: 192.168.77.102:32777
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: Private_10:ca:fe (10:00:00:10:ca:fe)
                    IP Address Length: 0
                    IP Address: NOT INCLUDED
                    MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)
```

Capture 11-3: *BGP EVPN Update originated by Leaf-102 and sent to Leaf-103 by Spine-11.*

Example 11-13 shows that the host Cafe MAC- and IP information is produced into L2RIB of Leaf-101 by BGP with the next-hop address of VIP/Anycast-IP address of the VCP domain 23 switches Leaf-102 and Leaf-103. The same output also shows that the MAC-IP binding is sent to ARP cache, actually to ARP suppression-cache.

```
Leaf-101# sh l2route evpn mac-ip evi 10 detail
Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv(D):Del Pending (S):Stale (C):Clear
(Ps):Peer Sync (Ro):Re-Originated
Topology    Mac Address    Prod   Flags Seq No Host IP Next-Hops
----------- -------------- ------ ---------- -------------------------
10          1000.0010.cafe BGP    --    0  192.168.11.11 192.168.100.23
            Sent To: ARP
            SOO: 775043377
10          1000.0010.beef HMM    --    0  192.168.11.12  Local
            Sent To: BGP
            L3-Info: 10077
```

Example 11-14: *Leaf-101 L2RIB*

Example 11-15 shows the ARP suppression-cache

```
Leaf-101# sh ip arp suppression-cache detail

Flags: + - Adjacencies synced via CFSoE
       L - Local Adjacency
       R - Remote Adjacency
       L2 - Learnt over L2 interface
       PS - Added via L2RIB, Peer Sync
       RO - Dervied from L2RIB Peer Sync Entry

Ip Address      Age      Mac Address    Vlan Physical-ifindex    Flags
Remote Vtep Addrs

192.168.11.12   00:04:30 1000.0010.beef   10 Ethernet1/4         L
192.168.11.11   01:28:40 1000.0010.cafe   10 (null)              R
192.168.100.23
```

Example 11-15: *ARP suppression-cache on Leaf-101.*

Ping shows that there is an IP connectivity between Cafe and Beef in VLAN 10.

```
Beef#ping 192.168.11.11
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.11.11, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/15/21 ms
```

Example 11-16: *ping from host Beef to Host Café*

Capture 11-4 shows that the outer destination IP address is correctly set to 192.168.100.23.

```
Frame 9: 164 bytes on wire (1312 bits), 164 bytes captured (1312 bits)
Ethernet II, Src: 5e:00:00:00:00:07 (5e:00:00:00:00:07), Dst: 5e:00:00:02:00:07
(5e:00:00:02:00:07)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 192.168.100.23
User Datagram Protocol, Src Port: 60540, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: Private_10:be:ef (10:00:00:10:be:ef), Dst: Private_10:ca:fe
(10:00:00:10:ca:fe)
Internet Protocol Version 4, Src: 192.168.11.12, Dst: 192.168.11.11
Internet Control Message Protocol
```

Capture 11-4: *Ping from Beef to Cafe.*

This is the basic operation vPC is implemented in VXLAN BGP EVPN fabric.

Advertising Primary IP address

In figure 11-6, there is an external network behind the vPC peer VTEP Leaf-103. This kind of setup might lead to the situation, where the spine switch Spine-11 sends dataflow to an external network to the vPC Peer switch Leaf-102 that has no how to route the packet and the data flow is black-holed. This might happens since, in basic setup, both vPC peers send a BGP EVPN Updates to Spine-11 with VIP/Anycast-IP. Note that vPC peers switches do not have a synchronization mechanism to synchronize Layer 3 prefix information.

The process is shown in Figure 11-6. (1) Router Ext-Ro02 sends the BGP Update about network 172.16.77.0/24 to Leaf-103, which in turn forwards the Update to the Spine-11 by using it's VIP/Anycast IP 192.168.100.23 as a next-hop. (3) Spine-11 sends the Update to its BGP RR Clients Leaf-101 and Leaf-102. Leaf-102 ignores the Update (Same SoO) and Leaf-101 installs the received information in TENANT77 specific tables (BGP, RIB). That is a simplified Control Plane operation. (4) Then the Data Plane operation, host Beef sends data to a host located in the network 172.16.77.0/24. It sends the IP packets to its Default Gateway Leaf-101, which knows that the destination network is reachable through the next-hop address 192.168.100.23 (vPC peers Leaf-102 and Leaf-103 VIP/Anycast address). (5) Leaf-101 sends the packet to Spine-11. Spine-11 has two possible paths towards the next-hop 192.168.100.23, to either via Leaf-102 or via Leaf-103. It might select the path to Leaf-102, which does not know how to route packet to destination network 172.16.77.0/24 and it drops the packet.

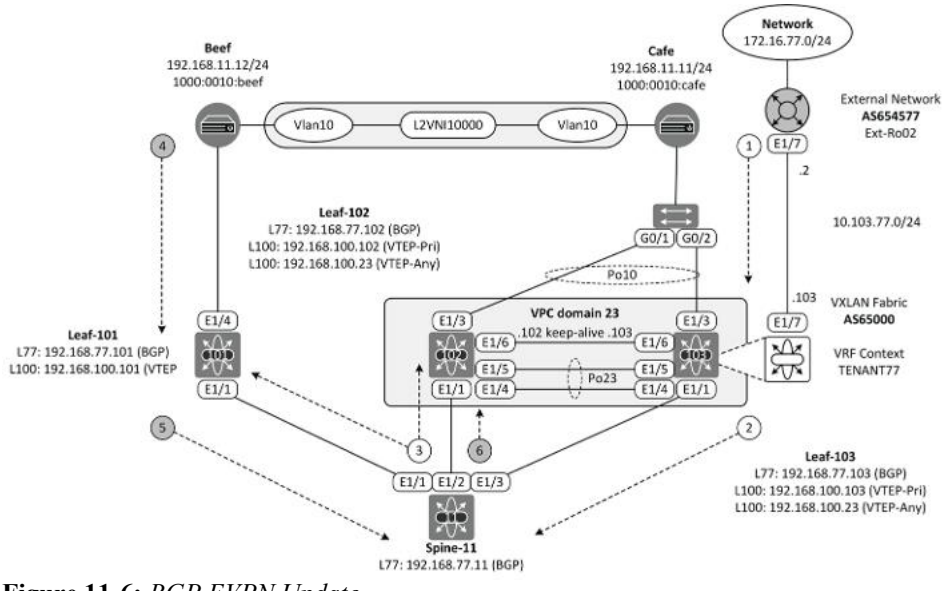


Figure 11-6: *BGP EVPN Update*

From Example 11-17, we can see that the external network is advertised with the next-hop address of vPC peer switches VIP/Anycast address 192.168.100.23.

```
Leaf-101# sh ip bgp vrf TENANT77 172.16.77.0
BGP routing table information for VRF TENANT77, address family IPv4 Unicast
BGP routing table entry for 172.16.77.0/24, version 6
Paths: (1 available, best #1)
Flags: (0x8008041a) on xmit-list, is in urib, is best urib route, is in HW
  vpn: version 6, (0x100002) on xmit-list

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: internal, path is valid, is best path, in rib
             Imported from
192.168.77.103:3:[5]:[0]:[0]:[24]:[172.16.77.0]:[0.0.0.0]/224
  AS-Path: 64577 , path sourced external to AS
    192.168.100.23 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0006.0007
      Originator: 192.168.77.103 Cluster list: 192.168.77.111

<snipped>
```

Example 11-17: *BGP table Leaf-101*

This behavior can be changed in a way that instead of advertising VIP as a Next-Hop for external prefixes, the PIP (Primary/Physical IP) is used. This is achieved with commands "***Advertise-pip"*** command under BGP AFI together with "***advertise virtual-rmac"*** command under NVE interface which together lets BGP use Primary IP address as a next-hop when advertising prefix-routes. These commands are enabled on both vPC peers switch.

```
router bgp 65000
  address-family l2vpn evpn
    advertise-pip
!
interface nve1
    advertise virtual-rmac
```

Example 11-18: *Advertise-pip and advertise virtual-rmac.*

After changes, we can see from the Leaf-101 that the next-hop is set to Primary IP (PIP) of Leaf-103.

```
Leaf-101# sh ip bgp vrf TENANT77 172.16.77.0
BGP routing table information for VRF TENANT77, address family IPv4 Unicast
BGP routing table entry for 172.16.77.0/24, version 19
Paths: (1 available, best #1)
Flags: (0x8008041a) on xmit-list, is in urib, is best urib route, is in HW
  vpn: version 21, (0x100002) on xmit-list

  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: internal, path is valid, is best path, in rib
             Imported from
192.168.77.103:3:[5]:[0]:[0]:[24]:[172.16.77.0]:[0.0.0.0]/224
  AS-Path: 64577 , path sourced external to AS
    192.168.100.103 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED 0, localpref 100, weight 0
      Received label 10077
      Extcommunity: RT:65000:10077 ENCAP:8 Router MAC:5e00.0006.0007
      Originator: 192.168.77.103 Cluster list: 192.168.77.111

  VRF advertise information:
  Path-id 1 not advertised to any peer

  VPN AF advertise information:
  Path-id 1 not advertised to any peer
```

Example 11-19: *Advertise-pip and advertise virtual-rmac.*

By using Advertise-pip and advertise virtual-rmac commands, the next-hop operation changes a little bit. From the Capture 11-5, we can see that MAC Advertisement Route (Type-2) still use VIP as next-hop (HEX 17 = DEC 23).

```
Frame 16: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits)
Ethernet II, Src: 5e:00:00:06:00:07 (5e:00:00:06:00:07), Dst: 5e:00:00:02:00:07
(5e:00:00:02:00:07)
Internet Protocol Version 4, Src: 192.168.77.103, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 49687, Dst Port: 179, Seq: 174, Ack:
188, Len: 112
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 112
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 89
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
```

```
            Length: 24
            Carried extended communities: (3 communities)
                Route Target: 65000:10000
                Route Origin: 192.168.100.23:0
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive, Complete
            Type Code: MP_REACH_NLRI (14)
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 33
                    Route Distinguisher: 192.168.77.103:32777
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: Private_10:ca:fe (10:00:00:10:ca:fe)
                    IP Address Length: 0
                    IP Address: NOT INCLUDED
                    MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)

0000  5e 00 00 02 00 07 5e 00 00 06 00 07 08 00 45 c0   ^.....^.......E.
0010  00 a4 72 be 00 00 40 06 eb 12 c0 a8 4d 67 c0 a8   ..r...@.....Mg..
0020  4d 0b c2 17 00 b3 31 66 99 80 93 a2 6c e5 80 18   M.....1f....l...
0030  19 34 39 0a 00 00 01 01 08 0a 00 3d e2 fe 00 3d   .49........=...=
0040  f7 f8 ff ff ff ff ff ff ff ff ff ff ff ff ff ff   ................
0050  ff ff 00 70 02 00 00 00 59 40 01 01 00 40 02 00   ...p....Y@...@..
0060  40 05 04 00 00 00 64 c0 10 18 00 02 fd e8 00 00   @.....d.........
0070  27 10 01 03 c0 a8 64 17 00 00 03 0c 00 00 00 00   '.....d.........
0080  00 08 90 0e 00 2c 00 19 46 04 c0 a8 64 17 00 02   .....,..F...d...
0090  21 00 01 c0 a8 4d 67 80 09 00 00 00 00 00 00 00   !....Mg.........
00a0  00 00 00 00 00 00 00 30 10 00 00 10 ca fe 00 00   .......0........
00b0  27 10                                             '.
```

Capture 11-5: *Route Type-2*

While IP prefix route (Type-5) uses PIP as next-hop address (HEX 67 = DEC 103).

```
Frame 44: 192 bytes on wire (1536 bits), 192 bytes captured (1536 bits)
Ethernet II, Src: 5e:00:00:06:00:07 (5e:00:00:06:00:07), Dst: 5e:00:00:02:00:07
(5e:00:00:02:00:07)
Internet Protocol Version 4, Src: 192.168.77.103, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 49687, Dst Port: 179, Seq: 381, Ack:
409, Len: 126
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 126
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 103
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: 64577
        Path Attribute - MULTI_EXIT_DISC: 0
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
```

```
            Length: 24
            Carried extended communities: (3 communities)
                Route Target: 65000:10077
                Encapsulation: VXLAN Encapsulation
                Unknown subtype 0x03: 0x5e00 0x0006 0x0007
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive
            Type Code: MP_REACH_NLRI (14)
            Length: 45
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (36 bytes)
                EVPN NLRI: IP Prefix route
                    Route Type: IP Prefix route (5)
                    Length: 34
                    Route Distinguisher: 192.168.77.103:3
                    ESI: 00 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    IP prefix length: 24
                    IPv4 address: 172.16.77.0
                    IPv4 Gateway address: 0.0.0.0
                    MPLS Label Stack: 629 (bottom)

0000  5e 00 00 02 00 07 5e 00 00 06 00 07 08 00 45 c0   ^.....^.......E.
0010  00 b2 72 ca 00 00 40 06 ea f8 c0 a8 4d 67 c0 a8   ..r...@.....Mg..
0020  4d 0b c2 17 00 b3 31 66 9a 4f 93 a2 6d c2 80 18   M.....1f.O..m...
0030  19 34 aa a6 00 00 01 01 08 0a 00 3d f6 32 00 3e   .4.........=.2.>
0040  09 ec ff ff ff ff ff ff ff ff ff ff ff ff ff ff   ................
0050  ff ff 00 7e 02 00 00 00 67 40 01 01 00 40 02 06   ...~....g@...@..
0060  02 01 00 00 fc 41 80 04 04 00 00 00 00 40 05 04   .....A.......@..
0070  00 00 00 64 c0 10 18 00 02 fd e8 00 00 27 5d 03   ...d.........'].
0080  0c 00 00 00 00 00 08 06 03 5e 00 00 06 00 07 90   .........^......
0090  0e 00 2d 00 19 46 04 c0 a8 64 67 00 05 22 00 01   ..-..F...dg.."..
00a0  c0 a8 4d 67 00 03 00 00 00 00 00 00 00 00 00 00   ..Mg............
00b0  00 00 00 00 18 ac 10 4d 00 00 00 00 00 00 27 5d   .......M......']
```

Capture 11-6: *Route Type-5*

This can be verified also from the BGP table of Leaf-101. Host Cafe related MAC and MAC/IP has 192.168.100.23 (VIP) as a next-hop, while external network 172.16.77.0/24 has 192.168.100.103 (PIP) as a next-hop address.

```
Leaf-101# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 346, Local Router ID is 192.168.77.101
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup

   Network            Next Hop          Metric     LocPrf   Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>l[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
                      192.168.100.101                   100    32768 i
* i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                      192.168.100.23                    100        0 i
*>i                   192.168.100.23                    100        0 i
*>l[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272
                      192.168.100.101                   100    32768 i
* i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
```

```
                     192.168.100.23                     100        0 i
*>i                  192.168.100.23                     100        0 i

Route Distinguisher: 192.168.77.102:3
*>i[5]:[0]:[0]:[24]:[192.168.11.0]:[0.0.0.0]/224
                     192.168.100.102                    100        0 i

Route Distinguisher: 192.168.77.102:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.23                     100        0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
                     192.168.100.23                     100        0 i

Route Distinguisher: 192.168.77.103:3
*>i[5]:[0]:[0]:[24]:[172.16.77.0]:[0.0.0.0]/224
                     192.168.100.103          0         100        0 64577 i
*>i[5]:[0]:[0]:[24]:[192.168.11.0]:[0.0.0.0]/224
                     192.168.100.103                    100        0 i

Route Distinguisher: 192.168.77.103:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.23                     100        0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
                     192.168.100.23                     100        0 i

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
* i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
                     192.168.100.23                     100        0 i
*>i                  192.168.100.23                     100        0 i
*>i[5]:[0]:[0]:[24]:[172.16.77.0]:[0.0.0.0]/224
                     192.168.100.103          0         100        0 64577 i
* i[5]:[0]:[0]:[24]:[192.168.11.0]:[0.0.0.0]/224
                     192.168.100.103                    100        0 i
*>i                  192.168.100.102                    100        0 i
```

Example 11-20: *Advertise-pip and advertise virtual-rmac*

The BGP table of TENANT77 in Leaf-101 shows that also local prefixes are advertised by using PIP (example 11-21).

```
Leaf-101# sh ip bgp vrf TENANT77
BGP routing table information for VRF TENANT77, address family IPv4 Unicast
BGP table version is 25, Local Router ID is 192.168.11.1
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup

   Network            Next Hop            Metric   LocPrf     Weight Path
*>i172.16.77.0/24     192.168.100.103          0      100   0  64577 i
* i192.168.11.0/24    192.168.100.103                100          0 i
*>i                   192.168.100.102                100          0 i
* i192.168.11.11/32   192.168.100.23                 100          0 i
*>i                   192.168.100.23                 100          0 i
```

Example 11-11: *TENANT77 BGP table*

References:

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

NX-OS and Cisco Nexus Switching – Next-Generation Data Center Architectures
Second Edition
ISBN-10: 1-58714-304-6 – Ron Fuller, David Jansen, and Matthew McPherson

Design and Configuration Guide: Best Practices for Virtual Port Channels (vPC) on Cisco Nexus7000 Series Switches - Revised: June 2016
LIST OF VPC BEST PRACTICES - Peter Welcher
https://www.netcraftsmen.com/vpc-best-practices-checklist/

Chapter 12: Multihoming - vPC and Graceful Insertion and Removal (GIR) operation

Does it really matter if the NVE1 interface of a VTEP switch and BGP EVPN use the same Loopback interface IP address as a source or should there be a dedicated Loopback interface for BGP EVPN? What about the Loopback Interface numbering, does it really matter. This chapter discusses the difference in the BGP EVPN convergence process for both of these design options.

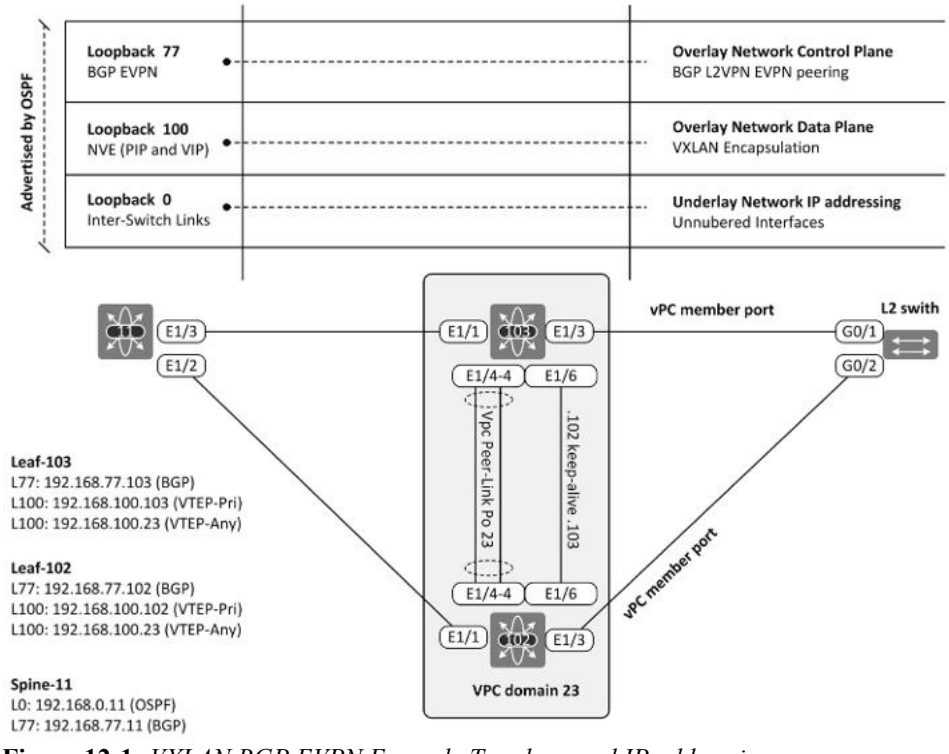


Figure 12-1: *VXLAN BGP EVPN Example Topology and IP addressing*

Loopback addressing

Figure 12-1 shows the example topology and the Loopback addresses used therein. The Loopback 0 is used in Inter-Switch links between the Spine and Leaf switches (Unnumbered physical links). The interfaces NVE1 in vPC Peer switches Leaf-102 and Leaf-103 use their Loopback 100 interface primary IP address as a Physical IP (VIP) and the secondary IP address as a Virtual/Anycast IP (VIP). BGP EVPN peering is done by using the Loopback 77 IP addresses. All of these Loopback IP addresses are advertised by OSPF.

vPC domain

Leaf-102 and Leaf-103 are vPC peer switches in vPC domain 23. vPC Peer-Link is established over PortChannel 23 and vPC Peer-Keepalive Link is Layer 3 link between switches. Both Leaf switches have one vPC Member Port belonging to PortChannel 10.

Graceful Insertion and Removal (GIR)

GIR is a method, which helps to maintain network availability while doing device-specific software or hardware maintenance tasks. In the first demonstration, BGP EVPN peering between Spine-11 and Leaf-103 is established between Loopback 77 interfaces.

Now we take the Leaf-103 out of service by using the command "***system mode maintenance***" (example 12-1).

```
Leaf-103(config)# system mode maintenance

Following configuration will be applied:

ip pim isolate
router bgp 65000
  isolate
router ospf UNDERLAY-NET
  isolate
vpc domain 23
  shutdown

NOTE: If you have vPC orphan interfaces, please ensure 'vpc orphan-port
suspend' is configured under them, before proceeding further
Do you want to continue (yes/no)? [no] yes

Generating before_maintenance snapshot before going into maintenance mode

Starting to apply commands...

Applying : ip pim isolate
Applying : router bgp 65000
Applying :   isolate
Applying : router ospf UNDERLAY-NET
Applying :   isolate
Applying : vpc domain 23
Applying :   shutdown2018 Aug 24 10:31:21 Leaf-103 %$ VDC-1 %$ %VPC-2-
VPC_SUSP_ALL_VPC: Peer-link going down, suspending all vPCs on secondary. If
vfc is bound to vPC, then only ethernet vlans of that VPC shall be down.
2018 Aug 24 10:31:21 Leaf-103 %$ VDC-1 %$ %VPC-2-VPC_SHUTDOWN: vPC shutdown
status is ON


Maintenance mode operation successful.
Leaf-103(maint-mode)(config)# 2018 Aug 24 10:31:25 Leaf-103 %$ VDC-1 %$ %MMODE-
2-MODE_CHANGED: System changed to "maintenance" mode.

Leaf-103(maint-mode)(config)# 2018 Aug 24 10:31:51 Leaf-103 %$ VDC-1 %$ %USER-
2-SYSTEM_MSG: NVE: send reinit to bring down nve1 - nve

Leaf-103(maint-mode)(config)#
```

Example 12-1: *Removing Leaf-103 by using GIR*

Figure 12-2 shows the reaction of Leaf-103.

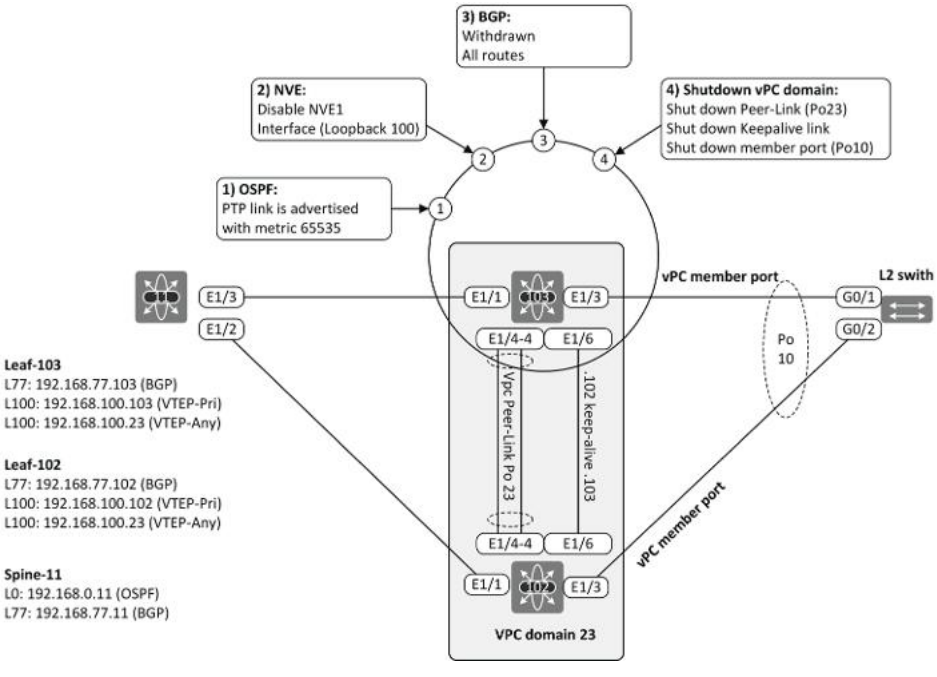


Figure 12-2: *Graceful Insertion and Remove (GIR) in Leaf-103.*

Verifications.

As can be seen from the example 12-2, vPC related PortChannels and related physical interfaces are suspended, Loopback 100 interface is disabled and the interface NVE1 is down.

```
Leaf-103(maint-mode)(config)# sh int statu | i Po10|Po23|Lo0|Lo77|Lo100|nve1
Eth1/4        ** Po23 member - v suspndByV trunk     full    auto    10g
Eth1/5        ** Po23 member - v suspndByV trunk     full    auto    10g
Po10          --                 suspndByV trunk     full    auto    --
Po23          --                 suspndByV trunk     full    auto    --
Lo0           ** RID/Underlay ** connected routed    auto    auto    --
Lo77          ** BGP peering **  connected routed    auto    auto    --
Lo100         ** VTEP/Overlay ** disabled  routed    auto    auto    --
nve1          --                 down      --        auto    auto    --
```

Example 12-2: *Interface state verification (Leaf-103)*

OSPF neighbor relations remains UP but PTP link is advertised with metric 65535.

```
Spine-11# sh ip ospf neighbors
 <snipped>
 Neighbor ID     Pri State            Up Time  Address         Interface
 Leaf-101          1 FULL/ -          04:26:45 192.168.0.101   Eth1/1
 Leaf-102          1 FULL/ -          04:25:45 192.168.0.102   Eth1/2
 Leaf-103          1 FULL/ -          04:25:46 192.168.0.103   Eth1/3


Spine-11# show ip ospf database router 192.168.0.103 detail
```

```
        OSPF Router with ID (192.168.0.11) (Process ID UNDERLAY-NET VRF
default)

                Router Link States (Area 0.0.0.0)

   LS age: 1708
   Options: 0x2 (No TOS-capability, No DC)
   LS Type: Router Links
   Link State ID: 192.168.0.103
   Advertising Router: Leaf-103
   LS Seq Number: 0x8000000d
   Checksum: 0xdd42
   Length: 60
    Number of links: 3

     Link connected to: a Stub Network
      (Link ID) Network/Subnet Number: 192.168.0.103
      (Link Data) Network Mask: 255.255.255.255
       Number of TOS metrics: 0
         TOS   0 Metric: 1

     Link connected to: a Router (point-to-point)
     (Link ID) Neighboring Router ID: 192.168.0.11
     (Link Data) Router Interface address: 0.0.0.2
       Number of TOS metrics: 0
         TOS   0 Metric: 65535

     Link connected to: a Stub Network
      (Link ID) Network/Subnet Number: 192.168.77.103
      (Link Data) Network Mask: 255.255.255.255
       Number of TOS metrics: 0
         TOS   0 Metric: 1
```

Example 12-3: *OSPF reaction to GIR*

BGP neighbor peering between Spine-11 and Leaf-103 stays UP but Leaf-103 has withdrawn all routes as we can see from the figure 12-4 (there is zero received prefix from Leaf-103).

```
Spine-11# sh bgp l2vpn evpn summary
<snipped>
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.77.101  4 65000     372     331      249    0    0 04:27:15 2
192.168.77.102  4 65000    5351    5325      249    0    0 04:26:18 4
192.168.77.103  4 65000    5347    5328      249    0    0 04:26:17 0
```

Example 12-4: *BGP reaction to GIR*

From the routing perspective, BGP and OSPF peering remains up and they just manipulate the routing updates. So the recovery is simple, OSPF and BGP just generate new routing updates. From the vPC domain perspective, all related interfaces will be brought UP. Now we are going to do the "Insertion" process by using the command "***no system mode maintenance"***, which brings Leaf-103 back to service (example 12-5).

```
Leaf-103(maint-mode)(config)# no system mode maintenance

Following configuration will be applied:

vpc domain 23
  no shutdown
router ospf UNDERLAY-NET
  no isolate
router bgp 65000
```

```
  no isolate
no ip pim isolate

Do you want to continue (yes/no)? [no] yes

Starting to apply commands...

Applying : vpc domain 23
Applying :   no shutdown2018 Aug 24 11:37:40 Leaf-103 %$ VDC-1 %$ %VPC-2-
VPC_SHUTDOWN: vPC shutdown status is OFF

Applying : router ospf UNDERLAY-NET
Applying :   no isolate
Applying : router bgp 65000
Applying :   no isolate
Applying : no ip pim isolate

Maintenance mode operation successful.

The after_maintenance snapshot will be generated in 120 seconds
After that time, please use 'show snapshots compare before_maintenance
after_maintenance' to check the health of the system
Leaf-103(config)# 2018 Aug 24 11:37:54 Leaf-103 %$ VDC-1 %$ %MMODE-2-
MODE_CHANGED: System changed to "normal" mode.
```

Example 12-5: *Bringing Leaf-103 back to service.*

Example-1 summary: BGP EVPN peering with dedicated Loopback addresses

The main point of the previous example is to show that BGP peering remains UP while removing Leaf-103 from service by using GIR. So there is no need for first bringing up the BGP peering before exchanging routing updates, which speeds up the recovery process.

Now I am going to change the BGP EVPN peering. Instead of using dedicated Loopback Interface for BGP, I am going to use the same Loopback Interface that is used by the NVE1 interface Loopback 100 (Figure 12-3).

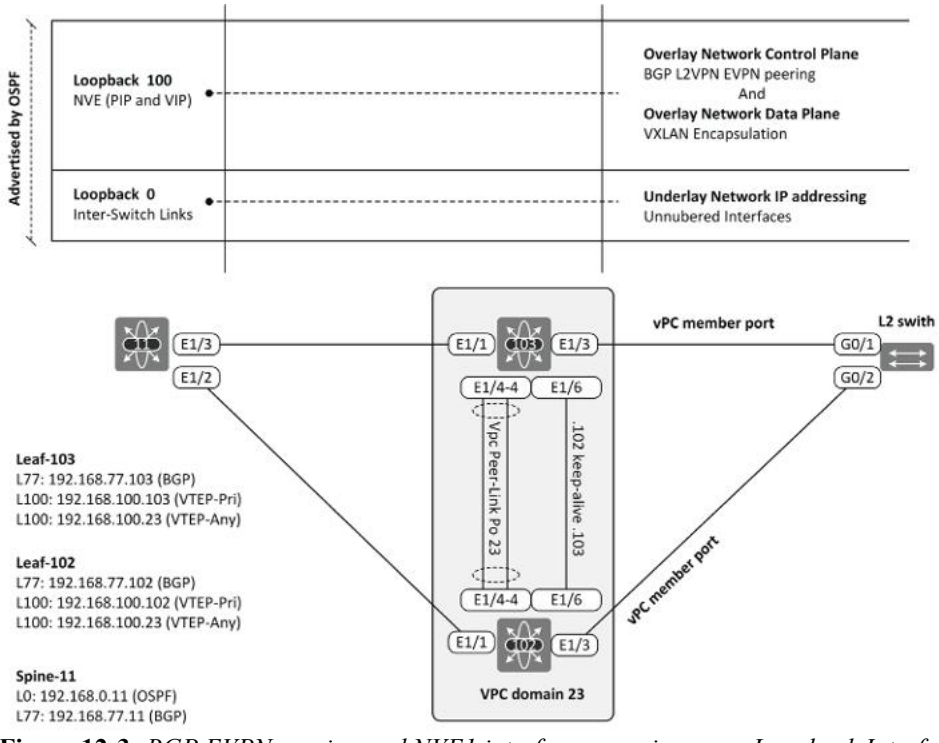


Figure 12-3: *BGP EVPN peering and NVE1 interface are using same Loopback Interface*

Example 12-6 shows the configuration of Leaf-103 related to BGP where Loopback 100 is used instead of Loopback 77.

```
router bgp 65000
  router-id 192.168.77.103
  timers bgp 3 9
  address-family ipv4 unicast
  address-family l2vpn evpn
    advertise-pip
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback100
    address-family l2vpn evpn
      send-community extended
```

Example 12-6: *BGP peering using Loopback 100.*

In Spine-11, the BGP peering is changed towards 192.168.100.103 (Loopback 100 in Leaf-103).

```
router bgp 65000
  router-id 192.168.77.111
  address-family ipv4 unicast
  address-family l2vpn evpn
  <snipped>
  neighbor 192.168.100.103
    remote-as 65000
    update-source loopback77
    address-family l2vpn evpn
      send-community
      send-community extended
      route-reflector-client
```

Example 12-7: *Configuring BGP peering using Loopback 100.*

As can be seen from output taken from Spine-11 (in example 12-8) peering is now up and there are five routes received from Leaf-103.

```
Spine-11# sh bgp l2 evpn summ
<snipped>
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.77.101  4 65000     449     398      318    0    0 05:25:15 2
192.168.77.102  4 65000    6514    6480      318    0    0 05:24:17 4
192.168.77.103  4 65000    6318    6304        0    0    0 00:09:54 Idle
192.168.100.103 4 65000     166     147      318    0    0 00:00:13 5
```

Example 12-8: *BGP peering using Loopback 100 - verification.*

Now the GIR process is repeated in Leaf-103 and verified if there are any major changes in the process.

The interface NVE1 is brought down.

```
Leaf-103(config)# system mode maintenance

Following configuration will be applied:

ip pim isolate
router bgp 65000
  isolate
router ospf UNDERLAY-NET
  isolate
vpc domain 23
  shutdown

NOTE: If you have vPC orphan interfaces, please ensure 'vpc orphan-port
suspend' is configured under them, before proceeding further
Do you want to continue (yes/no)? [no] yes

Generating before_maintenance snapshot before going into maintenance mode

Starting to apply commands...

Applying : ip pim isolate
Applying : router bgp 65000
Applying :   isolate
Applying : router ospf UNDERLAY-NET
Applying :   isolate
```

```
Applying : vpc domain 23
Applying :   shutdown2018 Aug 24 12:15:46 Leaf-103 %$ VDC-1 %$ %VPC-2-
VPC_SUSP_ALL_VPC: Peer-link going down, suspending all vPCs on secondary. If
vfc is bound to vPC, then only ethernet vlans of that VPC shall be down.
2018 Aug 24 12:15:46 Leaf-103 %$ VDC-1 %$ %VPC-2-VPC_SHUTDOWN: vPC shutdown
status is ON


Maintenance mode operation successful.
Leaf-103(maint-mode)(config)# 2018 Aug 24 12:15:50 Leaf-103 %$ VDC-1 %$ %MMODE-
2-MODE CHANGED: System changed to "maintenance" mode.
2018 Aug 24 12:16:16 Leaf-103 %$ VDC-1 %$ %USER-2-SYSTEM_MSG: NVE: send reinit
to bring down nve1 - nve
```

Example 12-9: *GIR in Leaf-103.*

And the Loopback interface 100 is disabled.

```
Leaf-103(maint-mode)(config)# sh int statu | i Lo100
Lo100         ** VTEP/Overlay ** disabled  routed    auto    auto    --
```

Example 12-10: GIR in Leaf-103.

This causes the BGP neighbor state goes to the Idle state, which means that the BGP neighbor relation between Spine-11 and Leaf-103 is down.

```
Spine-11# sh bgp l2 evpn summ | i 192.168.100.103
192.168.100.103 4 65000     262     250       0   0    0 00:05:30 Idle
```

Example 12-11: *BGP peering with the Leaf-103 change to IDLE.*

Now the BGP recovery process has to go through the BGP neighbor negotiation process first and it increases the recovery time.

Example-2 summary: BGP EVPN peering and NVE1 using the same Loopback interface.

The answer to the question presented at the beginning of the chapter: “Does it really matter if the NVE1 interface of a VTEP switch and BGP EVPN use the same Loopback interface IP address as a source or should there be a dedicated Loopback interface for BGP EVPN?”

And the answer is: By using a dedicated Loopback interface, the BGP peering remains up during the GIR process and speed up the recovery process.

One important aspect related to Loopback Interface selection. When the router boots up, it will enable Loopback Interfaces in numerical order starting from Loopback 0. If we get back to our example lab, we can see that there is one thing, which should have been done slightly different if we want to tune the convergence. To be able to speed up the BGP recovery process, the Loopback Interface number used by NVE1 should be smaller than the Loopback interface number used by BGP peering. This is because the NVE1 IP address is used as a next-hop-address in BGP EVPN Update messages sent by VTEP switches and BGP is not able to advertise routes until the next-hop (meaning the NVE1 source Loopback Interface) of the route is reachable.

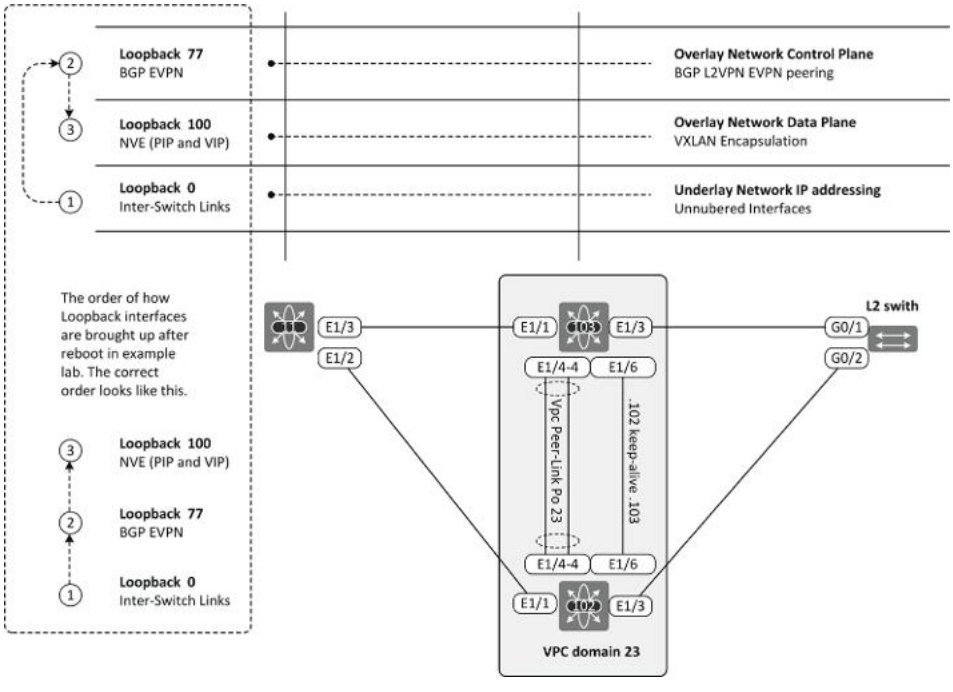


Figure 12-4: *Loopback interface "enabling" order during device boot.*

Conclusion

Even though the impact of the Loopback Interface numbering and usage to convergence time in VXLAN BGP EVPN fabric is a minor, the relationship between them is good to understand.

References:

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Nexus 9000/3000 Graceful Insertion and Removal (GIR): White Paper – SEP 2016: https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/white-paper-c11-737899.html

Chapter 13: Using vPC Peer Link as an Underlay Backup Path

This chapter explains how the VTEP switch can use a vPC peer link as a backup path to Spine switch in a situation where the Leaf switch loses connection to the Spine switch. This is recommended redundancy model when using vPC in VXLAN BGP EVPN fabric. For simplicity, there is only one spine switch used in example network.

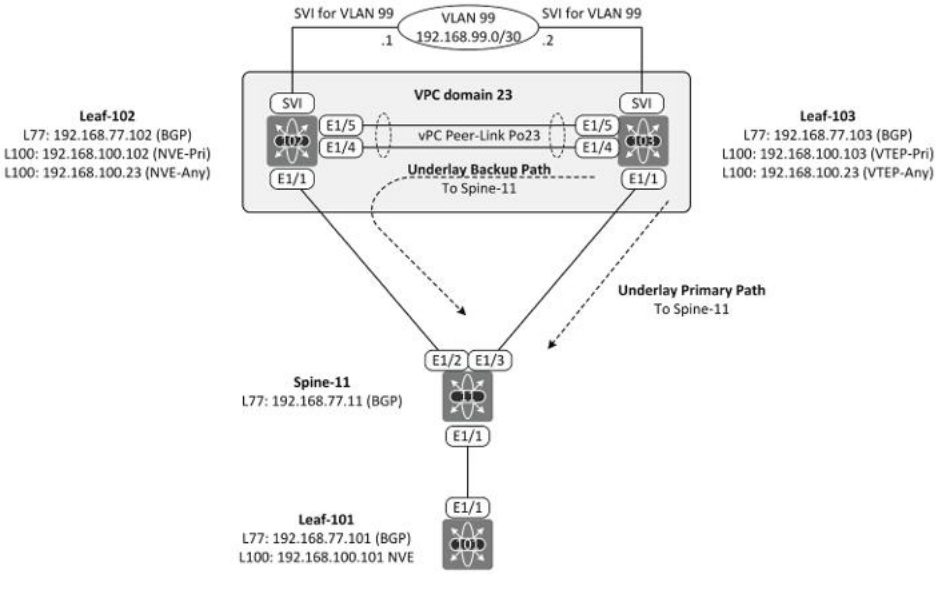


Figure 13-1: *Example Topology and IP addressing*

Example 13-1 illustrates the topology used in this chapter. Inter-Switch links use Unnumbered IP addressing (Loopback 0). Underlay unicast routing protocol is OSPF and PIM BiDir is used for Layer 2 BUM traffic. These two protocols are also needed in Inters-switch links between vPC peers in Backup Path solution.

Configuration

Step 1: Create VLAN 99

Step 2: Create Interface VLAN 99 an assign the IP address to it.

Step 3: Enable ospf and set link type to P2P in VLAN interface (we do not need DR/BDR election here)

Step 4: Enable PIM-SM on Interface VLAN 99

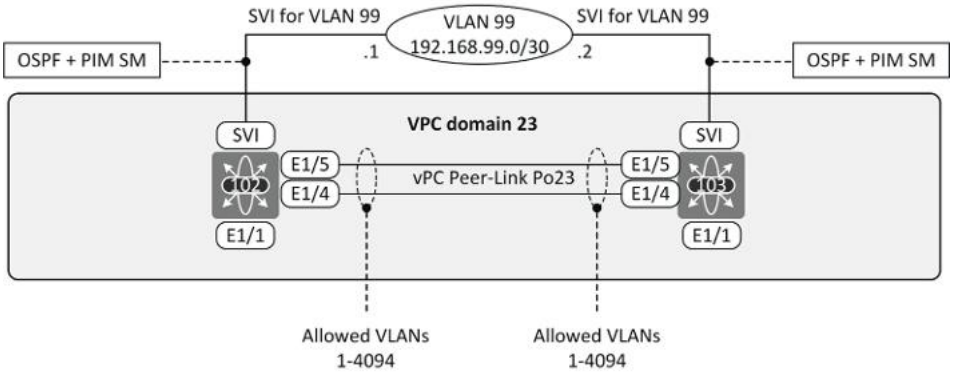


Figure 13-2: *SVI for vPC Backup.*

Note! Port-channel 23 (Trunk) is used as a vPC Peer-Link and VLANs 1-4094 are allowed in it.

```
Vlan 99
 Name Underlay-BUoVPC_Peer-Link
!
interface Vlan99
  description ** Underlay BU over vPC Peer-Link **
  no shutdown
  ip address 192.168.99.2/30
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
```

Example 13-1: *SVI for Backup over vPC Peer Link in VXLAN fabric.*

Note! VLAN 99 is used for establishing the Backup Underlay Network connection over the vPC Peer-Link. VLAN 99 is not a client VLAN (not mapped to any L2VNI) but an infra VLAN, which is why the command "***system nve infra-vlans 99***" is required when using physical Nexus switches. NX-OSv is used in this example and it does not have that command.

Verification

Example 13-2 shows that IP address used in SVI99 is reachable.

```
Leaf-102# ping 192.168.99.2
PING 192.168.99.2 (192.168.99.2): 56 data bytes
64 bytes from 192.168.99.2: icmp_seq=0 ttl=254 time=59.303 ms
64 bytes from 192.168.99.2: icmp_seq=1 ttl=254 time=47.207 ms
64 bytes from 192.168.99.2: icmp_seq=2 ttl=254 time=65.063 ms
64 bytes from 192.168.99.2: icmp_seq=3 ttl=254 time=46.248 ms
64 bytes from 192.168.99.2: icmp_seq=4 ttl=254 time=32.883 ms

--- 192.168.99.2 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 32.883/50.14/65.063 ms
```

Example 13-2: *ping test between vPC peers Leaf-102 and Leaf-103.*

Leaf-102 and Leaf-103 are OSPF and PIM neighbors.

```
Leaf-103# sh ip ospf neighbors
 OSPF Process ID UNDERLAY-NET VRF default
 Total number of neighbors: 2
 Neighbor ID     Pri State            Up Time  Address         Interface
 192.168.0.11      1 FULL/ -          03:03:07 192.168.0.11    Eth1/1
 192.168.0.102     1 FULL/ -          00:02:39 192.168.99.1    Vlan99
```

Example 13-3: *OSPF neighbors.*

```
Leaf-103# sh ip pim neighbor
PIM Neighbor Status for VRF "default"
Neighbor        Interface            Uptime    Expires   DR       Bidir-  BFD
                                                        Priority Capable State
192.168.0.11    Ethernet1/1          03:04:36  00:01:42  1   yes   n/a
192.168.99.1    Vlan99               01:20:16  00:01:36  1   yes   n/a
```

Example 13-4: *PIM neighbors.*

Figure 13-3 shows the path when an uplink from the traditional L2 switch to Leaf-102 and uplink from Leaf-103 to Spine-11 are down. After these operations, there is only one possible path from the host Cafe to the host Beef shown in Figure 13-3.

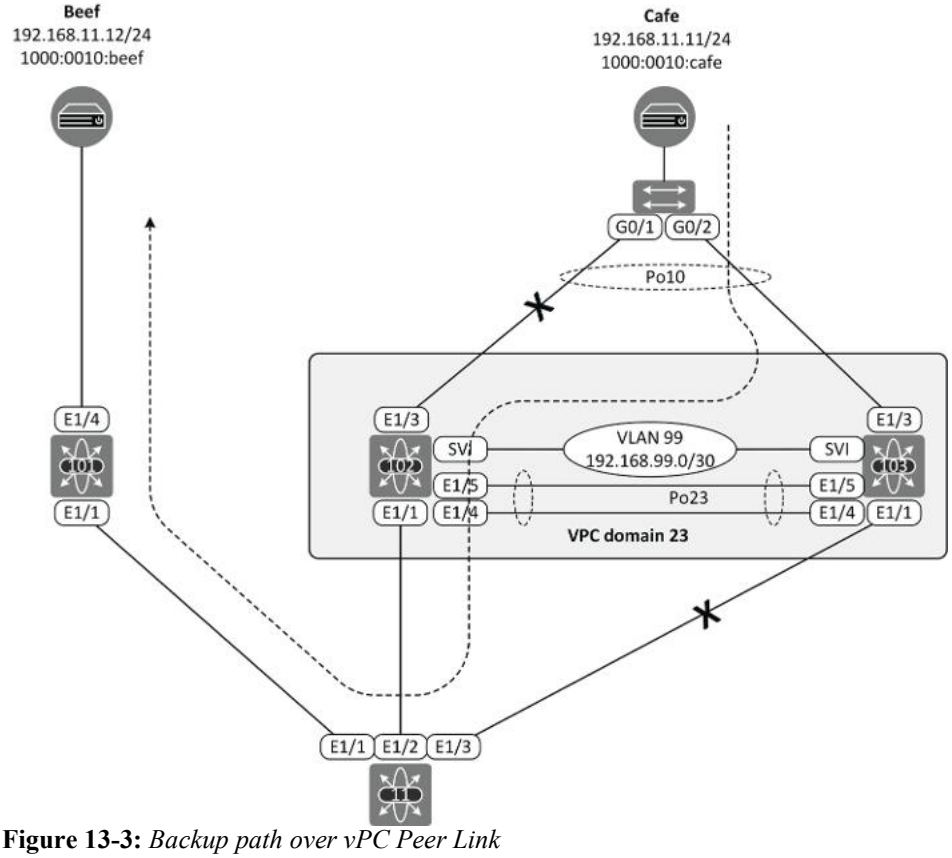


Figure 13-3: *Backup path over vPC Peer Link*

First, we are going to test Data Plane operation by pinging from host Cafe to host Beef. As we can see from the example 13-5, Data Plane is Ok.

```
Cafe#ping 192.168.11.12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.11.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/22/30 ms
```

Example 13-5: *ping from Cafe to Beef.*

Example 13-6 show the mac address-table of Leaf-103. It has learned the mac address of host Cafe through the Port-Channel 10, which leads to Ethernet Switch.

```
Leaf-103# show system internal l2fwder mac
<snipped>
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------
*    10    1000.0010.cafe   dynamic   00:04:38   F     F       Po10
```

Example 13-6: *Mac address-table of Leaf-103.*

Switch Leaf-102 in turn has learned the mac address of host Cafe via Po23, which is the vPC peer Link to Leaf-103. This is just basic mac address learning process based on flood & learn process.

```
Leaf-102#  show system internal l2fwder mac
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link,
        (T) - True, (F) - False, C - ControlPlane MAC
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------
<snipped>
+    10    1000.0010.cafe   dynamic   00:00:23   F     F       Po23
```

Example 13-7: *Mac address-table of Leaf-102.*

Since both vPC peer switches Leaf-103 and Leaf-102 has learned and installed the mac address of host Cafe to their mac address-table, they will also send a BGP update to Spine-11 (example 13-8). Note that Leaf-103 still has IP connectivity and BGP peering with Spine-11 over the vPC peer link.

```
Spine-11# sh bgp l2vpn evpn
<snipped>
Route Distinguisher: 192.168.77.102:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.23                    100         0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
                     192.168.100.23                    100         0
<snipped>
Route Distinguisher: 192.168.77.103:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.23                    100         0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
                     192.168.100.23                    100         0 i
```

Example 13-8: *Host Cafe mac and mac-ip routes in Spine-11.*

Leaf-101 has received this routing information from Spine-11. Note that the next-hop is set to vPC VIP address instead of PIP.

```
Leaf-101# sh bgp l2vpn evpn vni-id 10000
   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
<snipped>
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.23                    100        0 i
* i                  192.168.100.23                    100        0 i
```

```
*>i[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[192.168.11.11]/272
                      192.168.100.23                    100        0 i
* i                    192.168.100.23                    100        0 i
```

Example 13-9: *Host Cafe related information in Leaf-101.*

Leaf-101 installs the mac information in its L2RIB.

```
Leaf-101# show l2route mac all

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

Topology    Mac Address    Prod   Flags          Seq No     Next-Hops
----------- -------------- ------ -------------- ---------- ----------------
10          1000.0010.beef Local  L,             0          Eth1/4
10          1000.0010.cafe BGP    SplRcv         0          192.168.100.23
<snipped>
```

Example 13-10: *L2 RIBin Leaf-101.*

From L2RIB the mac address 1000.0010.cafe is stored into the mac address-table. Now all three switches have mac address 1000.0010.cafe in the mac address-table.

```
Leaf-101# show system internal l2fwder mac
<snipped>
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------
*    10    1000.0010.cafe    static   -          F     F  (0x47000001) nve-
peer1 192.168
<snipped>
```

Example 13-11: *Mac address-table Leaf-101.*

References:

VXLAN/EVPN Configuration Example (N9k / p2p) by Lukas Krattiger:
https://community.cisco.com/t5/data-center-blogs/vxlan-evpn-configuration-example-n9k-p2p/ba-p/3663830

Configuring VXLAN BGP EVPN:
https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/7-x/vxlan/configuration/guide/b_Cisco_Nexus_9000_Series_NX-OS_VXLAN_Configuration_Guide_7x/b_Cisco_Nexus_9000_Series_NX-OS_VXLAN_Configuration_Guide_7x_chapter_0100.pdf

Chapter 14: VXLAN Fabric Firewall Implementation

This chapter explains how to implement Active/Standby FW Cluster into VXLAN Fabric. Figure 14-1 shows the logical view of example setup, where there are two server networks: 192.168.30.0/24 (VLAN30 - protected) and 192.168.11.0/24 (VLAN10 - non-protected). We also have an Active/Standby FW Cluster connected to dedicated Service Leaf vPC Cluster (Leaf-102 and Leaf-103). Anycast Gateway (AGW) for the network 192.168.11.0/24 resides in the Server Leaf-101 while the Gateway for the protected network 192.168.30.0/24 resides in the Firewall (Inside Zone). Protected hosts in VLAN 30 use the VXLAN Fabric only as an L2 transport network. For simplicity, the Spine switch is not shown in the figure 14-1.

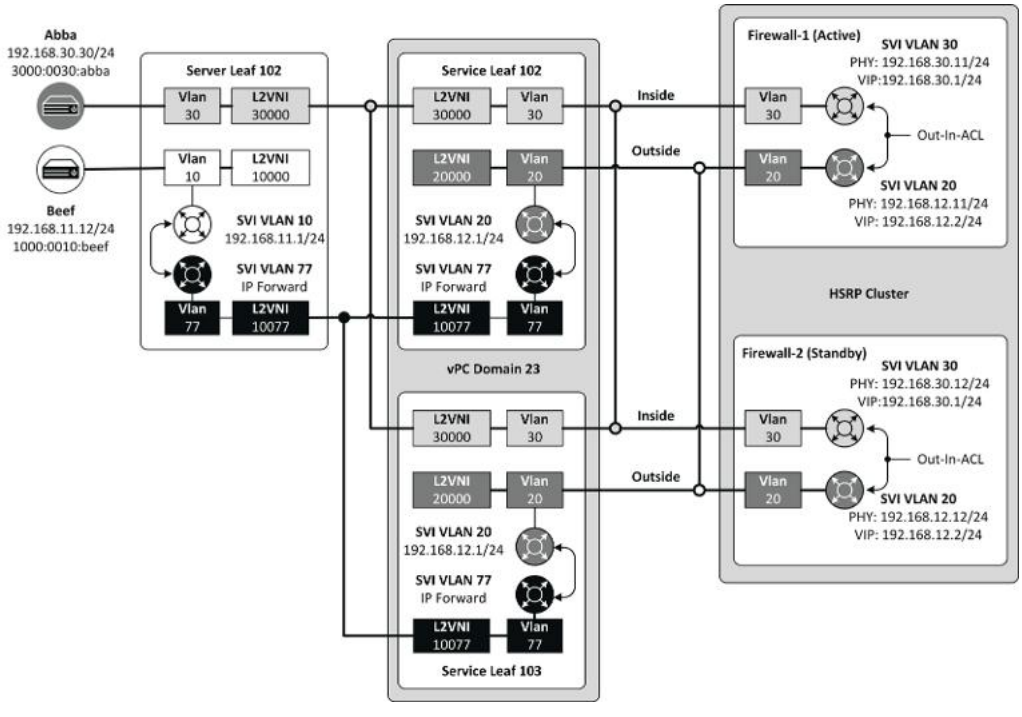


Figure 14-1: *Example Topology and IP addressing*

Note! Instead of using actual Firewalls devices in this lab, there are two Layer 3 switches used for simulating Firewall operations. HSRP is used to achieve the Active/Standby redundancy model. For the stateful filtering, reflective access-list is implemented between vlan 20 and vlan 30. There are no state synchronizations between “Firewalls”.

Protected segment

Server Leaf-102, where host Abba is connected to, has a VLAN 30 mapped to L2VNI 30000. BGP EVPN is used as a Control plane protocol to advertise host MAC/IP information. The same configuration is also added to Server Leaf-102 but as we can see in figure 14-2, neither switch has Anycast Gateway configured to VLAN 30. The Default Gateway for VLAN 30 is configured on the FW-1. This is one way to create a protected segment on VXLAN fabric.

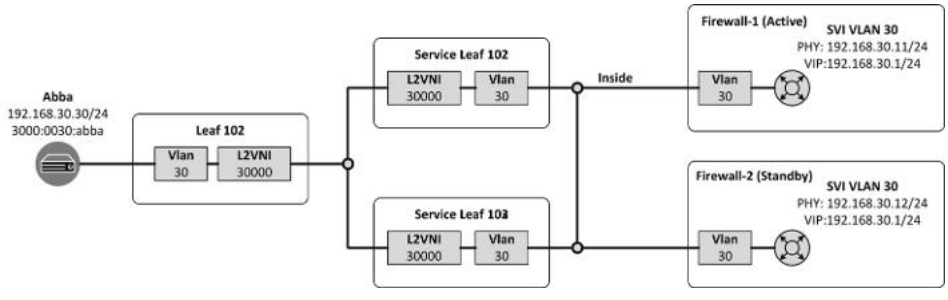


Figure 14-2: *Protected network*

The L2 configuration of Leaf switches form a protected network perspective is shown in example 14-1. First, VLAN 30 is mapped to L2VNI 30000 and then added under the EVPN configuration. Under NVE1 interface, the L2VNI specific settings such as ARP suppression and multicast group for BUM traffic are defined.

```
vlan 30
  name L2VNI-for-VLAN30
  vn-segment 30000
!
evpn
  vni 30000 l2
    rd auto
    route-target import auto
    route-target export auto
!
interface nve1
  member vni 30000
   suppress-arp
   mcast-group 238.0.0.10
!
```

Example14-1: *Leaf switches L2 configurations.*

The FW-1 configuration related to VLAN 30 can be seen in the example 14-2.

```
Vlan 30
 Name ** Inside **
!
interface Vlan30
 description ** Inside **
 ip address 192.168.30.11 255.255.255.0
 standby 30 ip 192.168.30.1
 standby 30 priority 110
 standby 30 timers 1 3
 standby 30 preempt
!
```

Example14-2: *FW-1 configurations.*

Non-Protected segment

The configuration for the non-protected segment in Server Leaf-102 is shown in example 14-3. First, VLAN 10 is created and mapped into L2VNI 10000. Anycast Gateway for the network 192.168.11.0/24 is 192.168.11.1. Then L2VNI 10000 is created under the EVPN configuration. In addition, L2VNI is added under the NVE1 interface configuration with the VNI specific options such as ARP suppression and multicast group for BUM traffic.

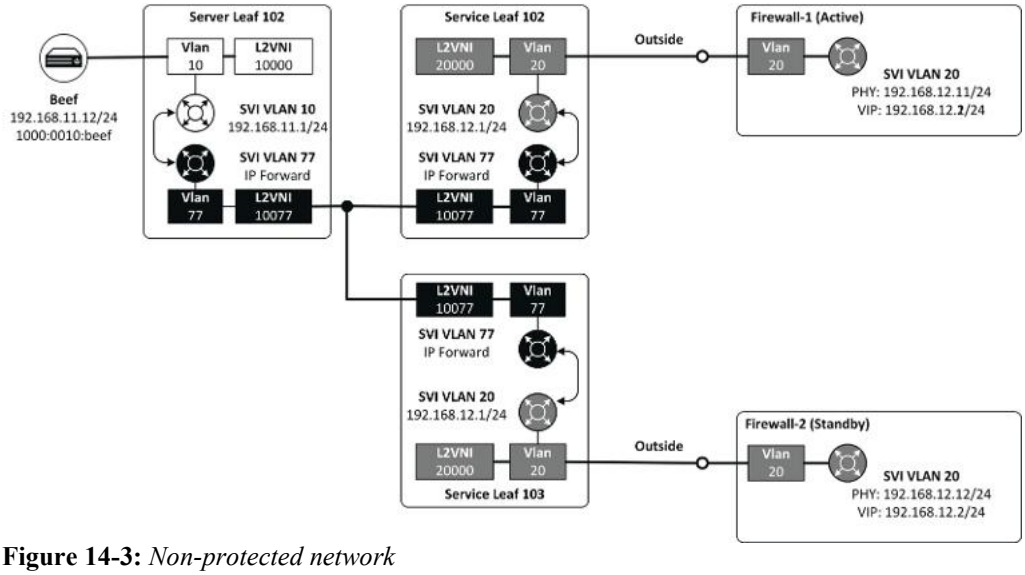


Figure 14-3: *Non-protected network*

Configuration related to the non-protected segment in Server Leaf-102 can be seen in example 14-3.

```
vlan 10
  name L2VNI-for-VLAN10
  vn-segment 10000
!
interface Vlan10
  no shutdown
  vrf member TENANT77
  ip address 192.168.11.1/24
  fabric forwarding mode anycast-gateway
!
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
!
interface nve1
  member vni 10000
    suppress-arp
    mcast-group 238.0.0.10
```

Example14-3: *Server Leaf-102 configuration for non-protected network.*

When comparing the segment between the FW-1 and the Service Leaf-102 it can be seen that it follows exactly the same design principles than non-protected segment 192.168.11.0/24 in Server Leaf-102. VLAN 20 has Anycast Gateway and it uses EVPN Control Plane and optional parameters are defined under NVE 1 interface. So from the VXLAN fabric perspective, FW-1 is the same kind of host than Abba. Example 14-4 shows the Service Leaf-102 configuration and example 14-5 shows the FW-1 configuration. Note that the configuration of the physical interfaces is excluded from both examples.

```
vlan 20
  name L2VNI-for-VLAN20
  vn-segment 20000
!
interface Vlan20
  no shutdown
  vrf member TENANT77
  no ip redirects
  ip address 192.168.12.1/24
  no ipv6 redirects
  fabric forwarding mode anycast-gateway
!
interface nve1
  member vni 20000
    suppress-arp
    mcast-group 238.0.0.10
!
evpn
  vni 20000 l2
    rd auto
    route-target import auto
    route-target export auto
```

Example14-4: *Service Leaf-102 configuration for the FW-1 Leaf-102 segment.*

```
Vlan 20
 Name ** Outside
!
interface Vlan20
 description ** Outside **
 ip address 192.168.12.11 255.255.255.0
 standby 20 ip 192.168.12.2
 standby 20 priority 110
 standby 20 preempt
 standby 20 timers 1 3
```

Example14-5: *FW-1 configuration for the FW-1 Leaf-102 segment.*

Segment 192.168.11.0/24 is only implemented in Server Leaf-102. Subnet 192.168.12.0/24 is implemented in Service Leafs and FWs. The data path between these two segments goes through the SVI 77, which is used for inter-VN routing in vrf context TENANT77. Configuration related to inter-VN routing can be seen in example 14-6.

```
vlan 77
  name TENANT77
  vn-segment 10077
!
interface Vlan77
  no shutdown
  vrf member TENANT77
  ip forward
!
vrf context TENANT77
  vni 10077
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
!
interface nve1
  host-reachability protocol bgp
  member vni 10077 associate-vrf
!
router bgp 65000
   vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
```

Example 14-6: *inter-VNI routing configuration.*

Figure 14-4 shows the setup that is now in place. We now should have connectivity from host Abba to FW-1 Inside interface IP address 192.168.30.1. The next step is to verify connectivity by pinging from host Beef to Outside interface of FW-1 (IP address 192.168.12.2).

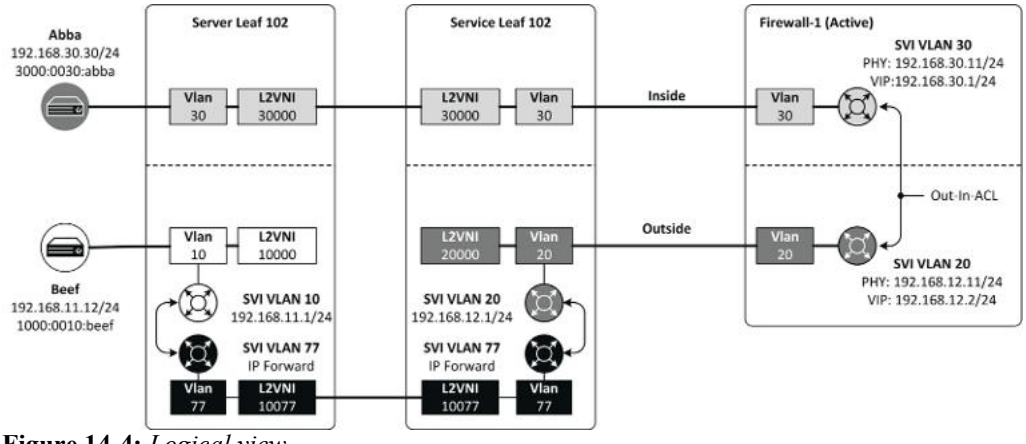


Figure 14-4: *Logical view*

.

```
Beef#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/33/66 ms
```

Example14-7: *Connectivity verification from Beef to FW-1 Outside interface.*

```
Abba#ping 192.168.30.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/19/24 ms
```

Example 14-8: *Connectivity verification from Abba to FW-1 Inside interface.*

The connection between Abba and Beef though does not work at this phase.

```
Beef#ping 192.168.30.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.30, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Example 14-9: *Connectivity verification from Beef to Abba.*

Looking at the BRIB of Server Leaf-102 shows that the route to network 192.168.30.0/0 is missing. It though has BGP route-type 2 information about MAC/IP addresses of all hosts in the segment (Abba, FW-1 virtual IP, and physical IP). This means that there is intra-VN connectivity (which was tested by pinging from Abba to inside interface of FW-1) but there is no IP connection between the network 192.168.11.0/24 and 102.168.30.0/24.

```
Server Leaf-102# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 198, Local Router ID is 192.168.77.101
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
i
njected
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>l[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
                      192.168.100.101                    100     32768 i
*>l[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[192.168.11.12]/272
                      192.168.100.101                    100     32768 i

Route Distinguisher: 192.168.77.101:32787    (L2VNI 20000)
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac14]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac14]:[32]:[192.168.12.2]/272
                      192.168.100.23                     100         0 i

Route Distinguisher: 192.168.77.101:32797    (L2VNI 30000)
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac1e]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>l[2]:[0]:[0]:[48]:[2000.0020.abba]:[0]:[0.0.0.0]/216
                      192.168.100.101                    100      32768 i
*>i[2]:[0]:[0]:[48]:[5e00.0006.801e]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac1e]:[32]:[192.168.30.1]/248
                      192.168.100.23                     100         0 i
*>l[2]:[0]:[0]:[48]:[2000.0020.abba]:[32]:[192.168.30.30]/248
                      192.168.100.101                    100     32768 i
*>i[2]:[0]:[0]:[48]:[5e00.0006.801e]:[32]:[192.168.30.11]/248
                      192.168.100.23                     100         0 i

Route Distinguisher: 192.168.77.102:3
*>i[2]:[0]:[0]:[48]:[5e00.0005.0007]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i

Route Distinguisher: 192.168.77.102:32787
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac14]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac14]:[32]:[192.168.12.2]/272
                      192.168.100.23                     100         0 i

Route Distinguisher: 192.168.77.102:32797
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac1e]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[5e00.0006.801e]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac1e]:[32]:[192.168.30.1]/248
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[5e00.0006.801e]:[32]:[192.168.30.11]/248
                      192.168.100.23                     100         0 i

Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
*>i[2]:[0]:[0]:[48]:[5e00.0005.0007]:[0]:[0.0.0.0]/216
                      192.168.100.23                     100         0 i
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac14]:[32]:[192.168.12.2]/272
                      192.168.100.23                     100         0 i
```

Example14-10: *BRIB from Leaf-101*

To fix this, the Service Leaf-102 needs to know where the network 192.168.30.0/24 is and then the information needs to be redistributed into BGP. This way Server Leaf-102 also gets the routing information. Dynamic routing can be used here but for simplicity, the static routing towards outside interface VIP address is used.

First, the static route is added under the vrf context TENANT77 on Service Leaf-102.

```
vrf context TENANT77
  vni 10077
  ip route 192.168.30.0/24 192.168.12.2
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
```

Example14-11: *static route on Service Leaf-102*

Next, the static routes are advertised to BGP on Server Leaf-102.

```
ip access-list PROTECTED_SEGMENTS
  10 permit ip 192.168.30.0/24 any
!
route-map PROTECTED_SEGMENTS permit 10
  match ip address PROTECTED_SEGMENTS
!
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
      redistribute static route-map PROTECTED_SEGMENTS
```

Example14-12: *redistribution of the static route in Server Leaf-102.*

Now Server Leaf-102 has information on how to reach the network 192.168.30.0/24 as can be seen from figure 14-13.

```
Server Leaf-102# sh bgp l2vpn evpn vni-id 10077
<snipped>
   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
*>i[2]:[0]:[0]:[48]:[5e00.0005.0007]:[0]:[0.0.0.0]/216
                      192.168.100.23                    100          0 i
*>i[2]:[0]:[0]:[48]:[0000.0c07.ac14]:[32]:[192.168.12.2]/272
                      192.168.100.23                    100          0 i
*>i[5]:[0]:[0]:[24]:[192.168.30.0]:[0.0.0.0]/224
                      192.168.100.102          0        100          0 ?
```

Example 14-13: *BRIB on Server Leaf-102*

Note! In this phase, there already is a vPC configuration in Service Leaf-102 by using a Physical IP address (PIP) as a next-hop for external networks and Virtual IP address (VIP) for internal networks. In this example Service Leaf-102 VTEP PIP is 192.168.100.102 (IP associated with NVE1) and VIP is 192.168.100.23. This is why the next-hop is different compared to FW-1 outside interface IP address 192.168.12.2 to network 192.168.30.0/24.

Now there is an IP connectivity between host Beef and Abba.

```
Beef#ping 192.168.30.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 30/35/40 ms
```

Example 14-14: *Connectivity verification from Beef to Abba*

As a final step, the filter that permits ping to Abba from any source is added between VLAN 30 and VLAN 20. Reflective ACL is used because it offers stateful look-alike ACL, whenever there is a hit for ACL entry, it is reflected/mirrored so that return traffic is allowed without static ACL. The "show ip access-list" command shows that there is reflected ACL IN-OUT-Mirror-ACL without any permit/deny statements on it since there has not been any traffic yet.

```
ip access-list extended OUT-IN-Acl
 remark ***************************
 remark ** ICMP ECHO/ECHO-REPLY  **
 permit icmp any host 192.168.30.30 echo reflect IN-OUT-Mirror-ACL
 permit icmp any host 192.168.30.30 echo-reply reflect IN-OUT-Mirror-ACL

interface Vlan30
 description ** Inside **
 ip address 192.168.30.11 255.255.255.0
 ip access-group OUT-IN-Acl out
 standby 30 ip 192.168.30.1
 standby 30 priority 110
 standby 30 preempt

FW-1#sh ip access-lists
Reflexive IP access list IN-OUT-Mirror-ACL
Extended IP access list OUT-IN-Acl
    10 permit icmp any host 192.168.30.30 echo reflect IN-OUT-Mirror-ACL
    20 permit icmp any host 192.168.30.30 echo-reply reflect IN-OUT-Mirror-ACL
```

Example 14-15: *reflective ACL on FW-1.*

After pinging from Beef to Abba there is five hits in both access-lists (example 15-17).

```
Beef#ping 192.168.30.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 37/49/62 m
```

Example 14-16: *reflective ACL on FW-1.*

```
FW-1#sh ip access-lists
Reflexive IP access list IN-OUT-Mirror-ACL
     permit icmp host 192.168.30.30 host 192.168.11.12  (5 matches) (time left
269)
Extended IP access list OUT-IN-Acl
    10 permit icmp any host 192.168.30.30 echo reflect IN-OUT-Mirror-ACL (5
matches)
    20 permit icmp any host 192.168.30.30 echo-reply reflect IN-OUT-Mirror-ACL
```

Example 14-17: *reflective ACL on FW-1.*

At this phase, the setup has no redundancy. Figure 14-5 shows the complete, redundant setup where there are both Active FW-1 and Passive FW-2 connected to Service Leaf switches Leaf-102 and Leaf 103 by using HSRP as an FW redundancy and vPC for Leaf switch redundancy. Physical connection is made over Port-channels between FWs and vPC Leaf switches (figure 14-6 shows physical topology).

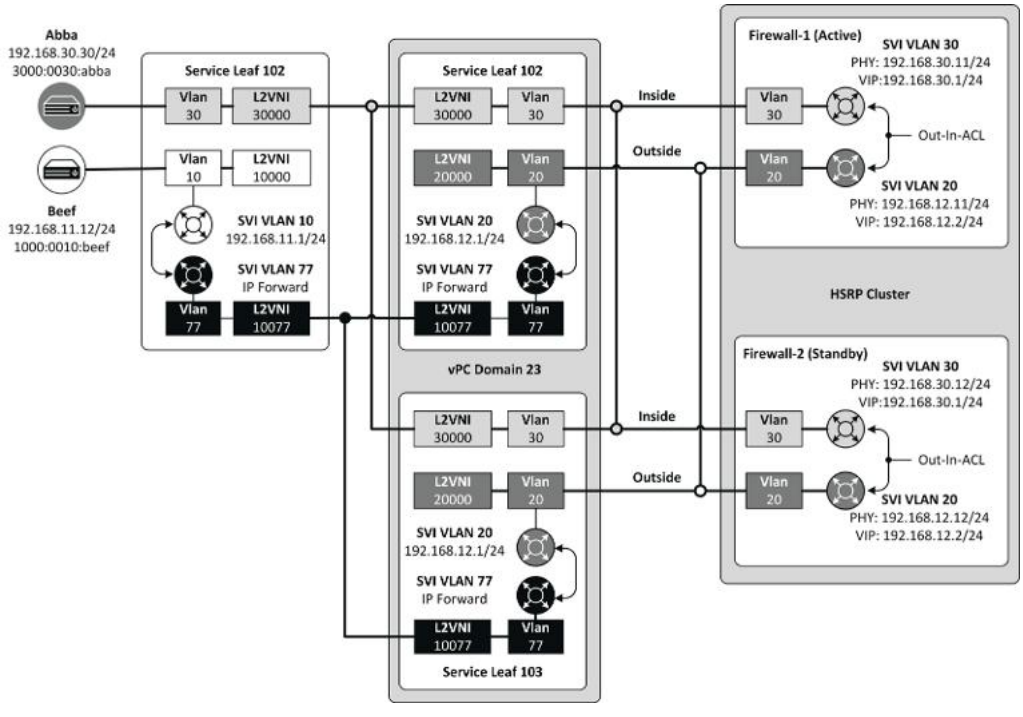


Figure 14-5: *Complete logical view*

Figure 14-6 shows the physical structure of example topology. Complete configuration of both Server Leaf switches and Firewalls can be found at the end of the chapter.

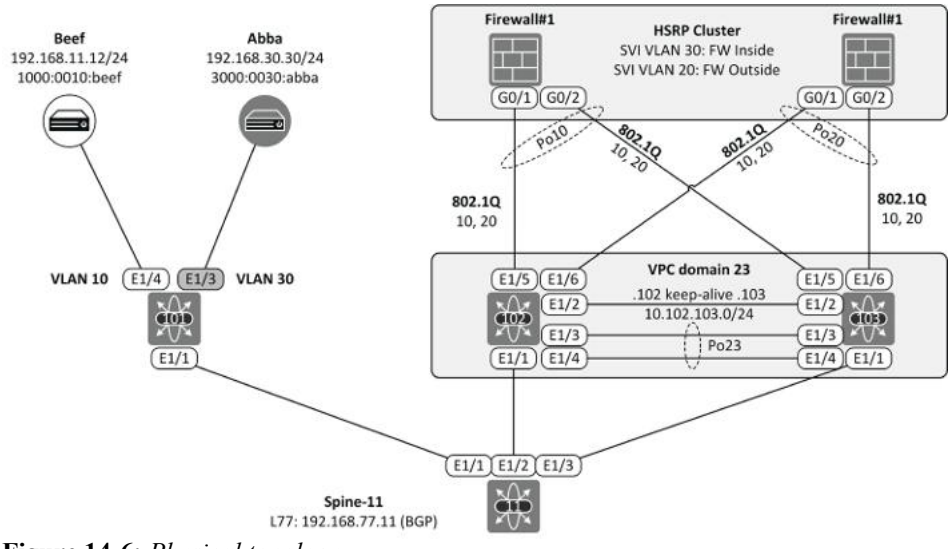


Figure 14-6: *Physical topology*

Figure 14-7 shows a simplified topology view.

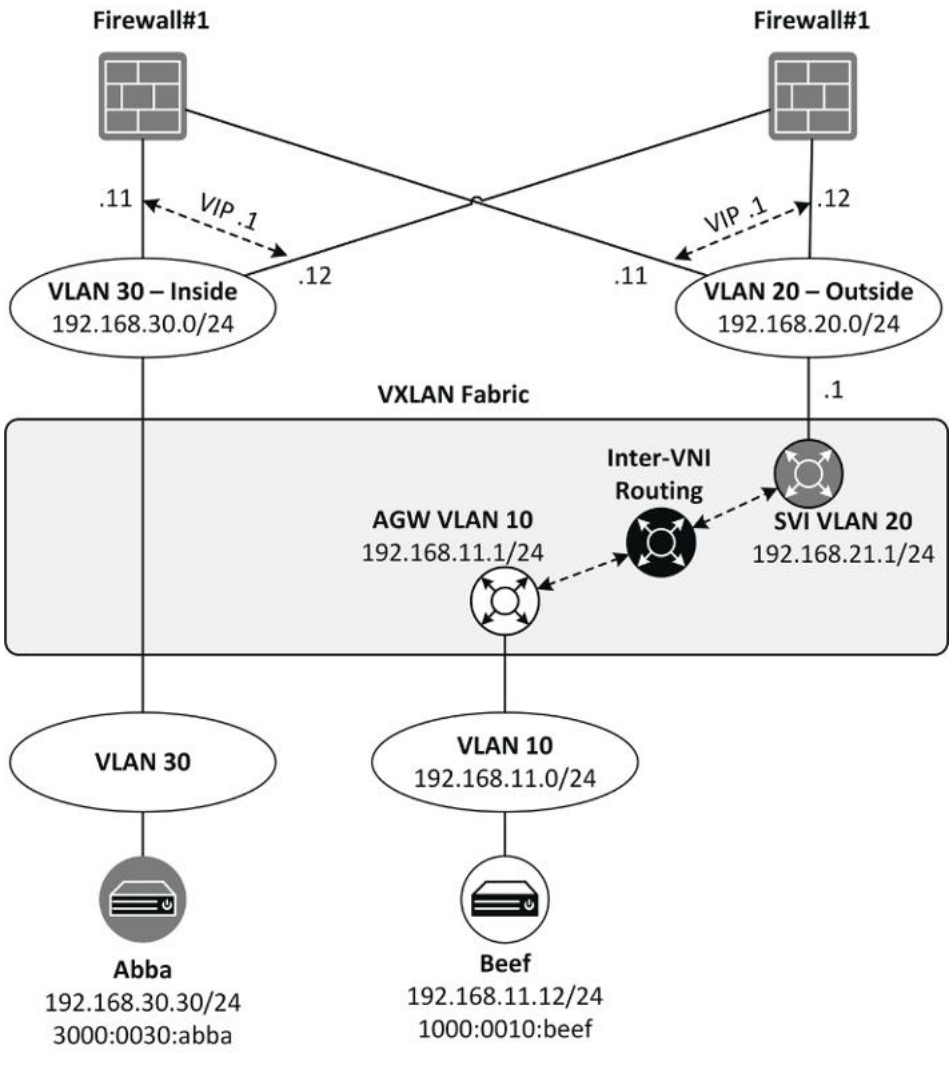


Figure 14-7: *Simplified Topology*

Connectivity Testing

Simple redundancy tests are made by pinging from Beef to Abba and shutting down the interfaces. As a first test (1), the link between FW-1 and Leaf-102 is set to down. Then FW-1 is completely restricted from network by shutting down the uplink between to Leaf-103 (2). This change generates the HSRP state change from Standby to Active in Leaf-103. As the last test, the link between FW-2 and Leaf-102 is brought down.

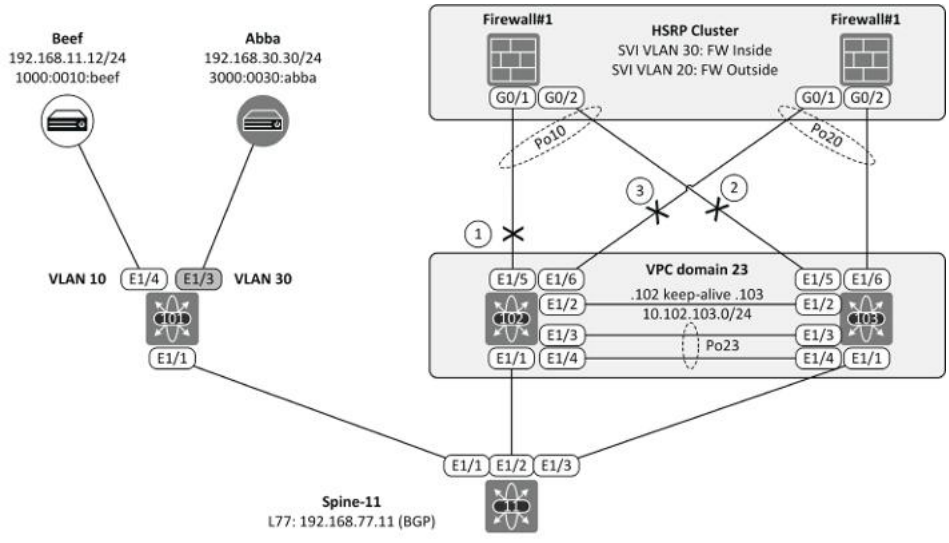


Figure 14-8: *Test scenarios*

Test 1. Link FW-1 to Leaf-102 down.

Traffic ICMP request/replies will be forwarded over the Leaf-103.

```
Beef#ping 192.168.30.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.30, timeout is 2 seconds:
!!!!!
```

Example 14-18: *Test 1- Link between FW-1 and Leaf-102 down.*

Test 2. Link FW-1 to Leaf-103 down.

```
Beef#ping 192.168.30.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.30, timeout is 2 seconds:
!!!!!
```

Example 14-19: *Test 1- Link between FW-1 and Leaf-103 down.*

```
FW-2#
* %HSRP-5-STATECHANGE: Vlan30 Grp 30 state Standby -> Active
* %HSRP-5-STATECHANGE: Vlan20 Grp 20 state Standby -> Active
```

Example 14-20: *HSRP state change in FW-2 when connection to FW-1 is down*

Test 3. Link FW-2 to Leaf-102 down.

The last redundant link between Leaf-102 and the rightmost Firewall is down

```
Beef#ping 192.168.30.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.30, timeout is 2 seconds:
!!!!!
```

Example 15-21: *Test 3- Link between FW-1 and Leaf-103 down.*
Test conclusion: Network react as expected.

References:

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Deploy Firewalls in Cisco Programmable Fabric
https://www.cisco.com/c/dam/en/us/products/collateral/switches/nexus-7000-series-switches/white-paper-c11-736585.pdf

Chapter 15: EVPN ESI Multihoming

This chapter explains the standard based VXLAN EVPN Multi-homing solution, implemented in NX-OS. It introduces the EVPN NLRIs route-type 1 Ethernet Auto-Discovery (Ethernet A-D) route and route-type 4 Ethernet Segment route. The first one is mainly used for reducing the convergence time while the second one is used for preventing forwarding loops by selecting Designated Forwarder (DF) per Ethernet Segment (ES). In addition, this chapter discusses the concept of Aliasing, which is used as a load-balancing method for Unicast traffic. Figure 15-1 illustrates the topology and addressing schemes used in this chapter.

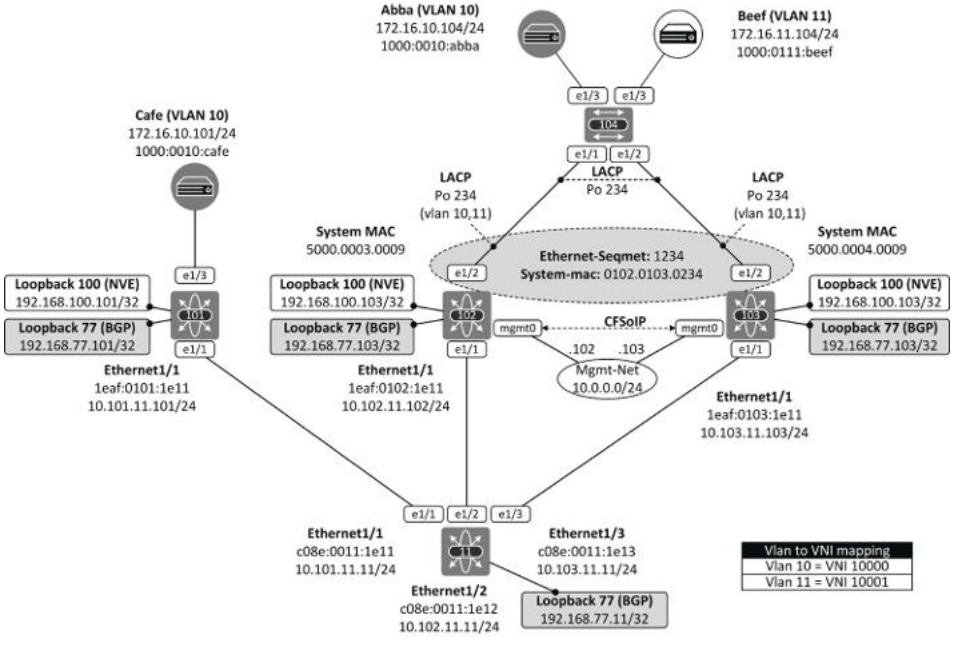


Figure 15-1: *The VXLAN EVPN Multi-homing topology and addressing scheme.*

Introduction

In the above figure 15-1, ASW-104 is connected to Leaf-102 and Leaf-103 via logical port-channel 234 that is bundled from interfaces e1/1 - 2 by using Link Aggregation Control Protocol (LACP). Leaf-102 and Leaf-103 are both connected to ASW-104 via interface e1/2, which are defined to be part of the port-channel 234. However, Leaf-102 and Leaf-103 are standalone switches without Multi-chassis Ether-Channel Trunk (MCT) between them. To be able to introduce themselves to ASW-104 as a single switch, Leaf -102 and Leaf-103 have to *first*, know that they belong to the same redundancy group and *second*, introduce the same system-MAC address to AWS-104 so it is able to bundle uplinks to port-channel. Also, leaf switches have to decide which one is allowed to forward BUM traffic (per VLAN) to and from

the ES. In addition, the Spanning-Tree root has to be in Leaf switches. To protection against packet loss caused by an uplink failure on either leaf switches (AWS-104 does not recognize these failure events), also *Core Link Tracking* should be enabled on uplink ports of leaf switches. To protect against VLAN misconfiguration on leaf switches, *Cisco Fabric Service over IP (CFSoIP)* should be implemented used on leaf switches that share the ES. In order for leaf and spine switches to do Equal Cost Multi-Pathing (ECMP) for VXLAN encapsulated frames, the maximum-paths for iBGP has to be adjusted.

Ethernet Segment Identifier (ESI) and Port-Channel

EVPN ESI multi-homing is enabled using evpn ***"esi multihoming global"*** configuration command. Interfaces g0/1 and g0/2 in ASW-104 are bundled to the port-channel 234 while interface e1/2 in both Leaf-102 and Leaf-103 participate in the port-channel 234 though they are not bundled because leaf switches are stand-alone devices. From the ASW-104 perspective, the port-channel 234 using LACP for link-aggregation and 802.1Q for trunking is just a regular port-channel. From the Leaf-102 and Leaf-103 perspective, the port-channel represents an EVPN *Ethernet Segment* (ES). The segment is activated using "***ethernet-segment"*** command with "***system-mac***" sub-command under the interface port-channel 234. Even though it looks like that the ***"ethernet segment"*** -command defines the *Ethernet Segment Identifier (ESI)*, it only defines part of it called *ES Local Discriminator (ES LD)*. The actual ESI consists of three parts; the first octet defines the type of the ESI, which in case of Cisco NX-OS is MAC-based ESI value (0x03). The next six octets are taken from the system-MAC configuration. The last three octet includes the ES LD value defined under the interface port-channel. Thus, the Ethernet Segment Identifier in this example scenario is 03.01.02.0103.02.34. In addition to using the system-MAC as a part of the ESI value it is also used in LACP messages *Actor System* field to represent local system-MAC. Since both Leaf-102 and Leaf-103 uses the same system-MAC, ASW-104 sees them as a one switch and is able to bring up the port-channel interface. Example 15-1 shows the configuration used in both Leaf-102 and Leaf-103. Figure 15-2 illustrates the physical topology and addressing scheme as well as LACP message exchanges between switches. Note that the source and destination MAC addresses used in Ethernet header are the real system MAC addresses.

```
evpn esi multihoming
!
interface port-channel234
  switchport mode trunk
  switchport trunk allowed vlan 10-11
  ethernet-segment 1234
    system-mac 0102.0103.0234
!
Interface Ethernet1/2
  Switchport mode trunk
  Switchport trunk allowed vlan 10,11
  Channel-group 10 mode active
```

Example 15-1: *Enabling EVPN multi-homing on Leaf switches*

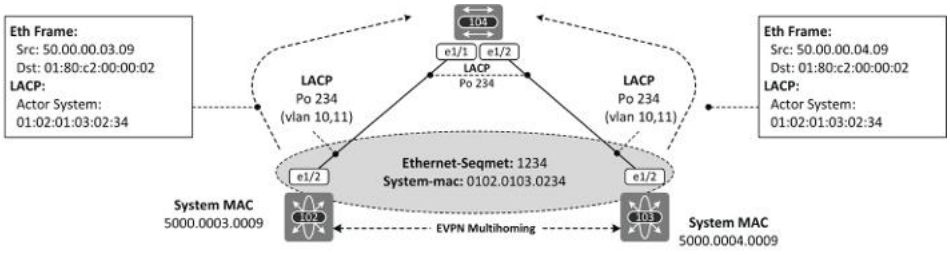


Figure 15-2: *EVPN ESI Multihoming:*

Example 15-2 shows that both interface g0/1 and g0/2 on switch ASW-104 are participating in Port-Channel 234.

```
ASW-104# show port-channel summary | b Group
Group Port-       Type     Protocol  Member Ports
      Channel
------------------------------------------------------------------------------
234   Po234(SU)   Eth      LACP      Eth1/1(P)    Eth1/2(P)
```

Example 15-2: *Port-channel 234 state on ASW-104.*

Designated Forwarder (DF)

Example 15-3 illustrates the BGP table of Leaf-102 after implementing the EVPN multi-homing. The first highlighted entry (with its attached Path Attributes) is generated and advertised by Leaf-102 (Figure 15-3 explains the NLRI values). The System-MAC address and ES configured under the port-channel 234 defines *Ethernet Segment Identifier (ESI)*. The second highlighted line describes the carried Extended Communities. The Route-Target Extended Community is auto-derived from the system-MAC address. Since both switches Leaf-102 and Leaf-103 use the same System-MAC, they also generate the same RT value for export/import policy. This means that Leaf-103 imports the NLRI advertised by Leaf-102 and another way around. This way seswitch know the existence of each other. The last two highlighted parts show the BGP Adj-RIB-In and Loc-RIB tables concerning BGP Update received from Leaf-103.

```
Leaf-102# sh bgp l2vpn evpn route-type 4
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.102:27233    (ES [0301.0201.0302.3400.04d2 0])
BGP routing table entry for
[4]:[0301.0201.0302.3400.04d2]:[32]:[192.168.100.102]/136, version 4
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    192.168.100.102 (metric 0) from 0.0.0.0 (192.168.77.102)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: ENCAP:8 RT:0102.0103.0234
```

```
  Path-id 1 advertised to peers:
    192.168.77.11
BGP routing table entry for
[4]:[0301.0201.0302.3400.04d2]:[32]:[192.168.100.103]/136, version 18
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported from
192.168.77.103:27233:[4]:[0301.0201.0302.3400.04d2]:[32]:[192.168.100.103]/136
  AS-Path: NONE, path sourced internal to AS
    192.168.100.103 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: ENCAP:8 RT:0102.0103.0234
      Originator: 192.168.77.103 Cluster list: 192.168.77.111

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.103:27233
BGP routing table entry for
[4]:[0301.0201.0302.3400.04d2]:[32]:[192.168.100.103]/136, version 17
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.103 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: ENCAP:8 RT:0102.0103.0234
      Originator: 192.168.77.103 Cluster list: 192.168.77.111

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

Example 15-3: *BGP L2VPN EVPN Ethernet Segment Route (Type 4) sent by Leaf-102.*

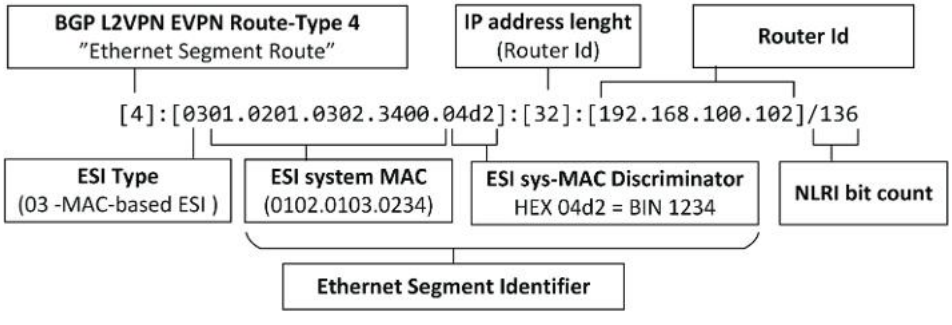


Figure 15-3: *The VXLAN EVPN Multi-homing topology and addressing scheme.*

Capture 15-1 illustrates the BGP Update message sent by Leaf-102.

```
Frame 208: 160 bytes on wire (1280 bits), 160 bytes captured (1280 bits) on
interface 0
Ethernet II, Src: 1e:af:01:02:1e:11, Dst: c0:8e:00:11:1e:12
Internet Protocol Version 4, Src: 192.168.77.102, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 179, Dst Port: 29824, Seq: 153, Ack:
153, Len: 94
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 94
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 71
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 16
            Carried extended communities: (2 communities)
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
                    Type: Transitive Opaque (0x03)
                    Subtype (Opaque): Encapsulation (0x0c)
                    Tunnel type: VXLAN Encapsulation (8)
                ES Import: RT: 01:02:01:03:02:34 [Transitive EVPN]
                    Type: Transitive EVPN (0x06)
                    Subtype (EVPN): ES Import (0x02)
                    ES-Import Route Target: 01:02:01:03:02:34
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive
            Type Code: MP_REACH_NLRI (14)
            Length: 34
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (25 bytes)
                EVPN NLRI: Ethernet Segment Route
                    Route Type: Ethernet Segment Route (4)
                    Length: 23
                    Route Distinguisher: 192.168.77.102:27233
                    ESI: 01:02:01:03:02:34, Discriminator: 00 04
                        ESI Type: ESI MAC address defined (3)
                        ESI system MAC: 01:02:01:03:02:34
                        ESI system mac discriminator: 00 04
                        Remaining bytes: d2
                    IP Address Length: 32
                    IPv4 address: 192.168.100.102
```

Capture 15-1: *BGP Update concerning ESI sent by Leaf-102*

Data Plane testing using ping shows that the channel is also operational. Example 15-4 shows that host Abba (172.16.10.104) in VLAN 10 can ping its Anycast Gateway IP address 172.16.10.1. In addition, host Abba is able to ping host Beef (172.16.11.104) in VLAN 11. Also, there is an IP connectivity between host Abba 172.16.10.101 connected to Leaf-101 and host Cafe.

```
Abba#ping 172.16.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.1, timeout is 2 seconds:
!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 20/28/44 ms
Abba#ping 172.16.11.104
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.11.104, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/60/96 ms

Abba#ping 172.16.10.101
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.101, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 76/123/164 ms
```

Example 15-4: *ping from Abba to a) VLAN 10 AGW b) host Beef c) host Cafe.*

Designated Forwarder

Switches Leaf-102 and Leaf-103 are seen by ASW-104 as a one switch that is connected through the Port-Channel 234. Leaf switches choose Designated Forwarder (DF) among themselves to forward BUM (Broadcast, Unknown Unicast and Multicast) traffic to and from ES. If ES has more than one VLAN, the DF roles are load-balanced between leaf nodes, i.e. DF for VLAN 10 is Leaf-102 and DF for VLAN 11 is Leaf-104. The selection process uses formula i = V mod N, where V represents VLAN Id and N represents a number of leaf switches in redundancy group. The "i" is an ordinal of leaf switch in redundancy group. When Leaf-102 and Leaf-103 exchange BGP L2VPN EVPN Route-Type 4 (Ethernet Segment Route) their IP address is included in NLRI. Each switch sets these IP addresses learned from BGP Update in numerical order from lowest to highest. So in case of Leaf-102 and Leaf-103 the order is 192.168.100.102, 192.168.100.103. The lowest IP i.e. 192.168.100.102 gets ordinal zero (0) and the next one gets ordinal one (1) and so on.

Formula to calculate DF for VLAN 10 is

V mod N = i
V = 10 (VLAN Id)
N = 2 (number of leaf switches)
10 mod 2 = 0 > Leaf-102
(Remainders is zero (0) when 10 is divided by 2)

Formula to calculate DF for VLAN 11 is

V mod N = i
V = 11 (VLAN Id)
N = 2 (number of leaf switches)
11 mod 2 = 0 > Leaf-103

(Remainders is one (1) when 11 is divided by 2)

Example 15-5 shows that Leaf-102 DF list includes IP address 192.168.100.102 (Leaf-102) and 192.168.100.103 (Leaf-103). It also shows that there are two active VLANs (10 and 11) in this redundancy group and Leaf-102 is DF for VLAN 10.

```
Leaf-102# sh nve ethernet-segment

ESI: 0301.0201.0302.3400.04d2
   Parent interface: port-channel234
  ES State: Up
  Port-channel state: Up
  NVE Interface: nve1
   NVE State: Up
   Host Learning Mode: control-plane
  Active Vlans: 10-11
   DF Vlans: 10
   Active VNIs: 10000-10001
  CC failed for VLANs:
  VLAN CC timer: 0
  Number of ES members: 2
  My ordinal: 0
  DF timer start time: 00:00:00
  Config State: config-applied
  DF List: 192.168.100.102 192.168.100.103
  ES route added to L2RIB: True
  EAD/ES routes added to L2RIB: True
  EAD/EVI route timer age: not running
--------------------------------------
Leaf-102#
```

Example 15-5: *show nve Ethernet-segment on Leaf-102.*

Example 15-6 shows that Leaf-103 is DF for VLAN 11.

```
Leaf-103# sh nve ethernet-segment

ESI: 0301.0201.0302.3400.04d2
   Parent interface: port-channel234
  ES State: Up
  Port-channel state: Up
  NVE Interface: nve1
   NVE State: Up
   Host Learning Mode: control-plane
  Active Vlans: 10-11
   DF Vlans: 11
   Active VNIs: 10000-10001
  CC failed for VLANs:
  VLAN CC timer: 0
  Number of ES members: 2
  My ordinal: 1
  DF timer start time: 00:00:00
  Config State: config-applied
  DF List: 192.168.100.102 192.168.100.103
  ES route added to L2RIB: True
  EAD/ES routes added to L2RIB: True
  EAD/EVI route timer age: not running
--------------------------------------
```

Example 15-6: *show nve Ethernet-segment on Leaf-103.*

References:

[RFC 7432] A. Sajassi et al., “BGP MPLS-Based Ethernet VPN”, RFC 7432, February 2015.

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

Cisco Programmable Fabric with VXLAN BGP EVPN Configuration Guide
https://www.cisco.com/c/en/us/td/docs/switches/datacenter/pf/configuration/guide/b-pf-configuration/IP-Fabric-Underlay-Options.html

Chapter 16: EVPN ESI Multihoming - Fast Convergence and Load Balancing

This chapter introduces the BGP EVPN Route Type1- Ethernet Auto-Discovery (Ethernet A-D) routes. The first section explains the Ethernet A-D per Ethernet Segment (ES) routes, which is mainly used for *Fast Convergence*. The second section discusses Ethernet A-D per EVI/ES route, which in turn is used for Load Balancing (also called *Aliasing/Backup Path)*.

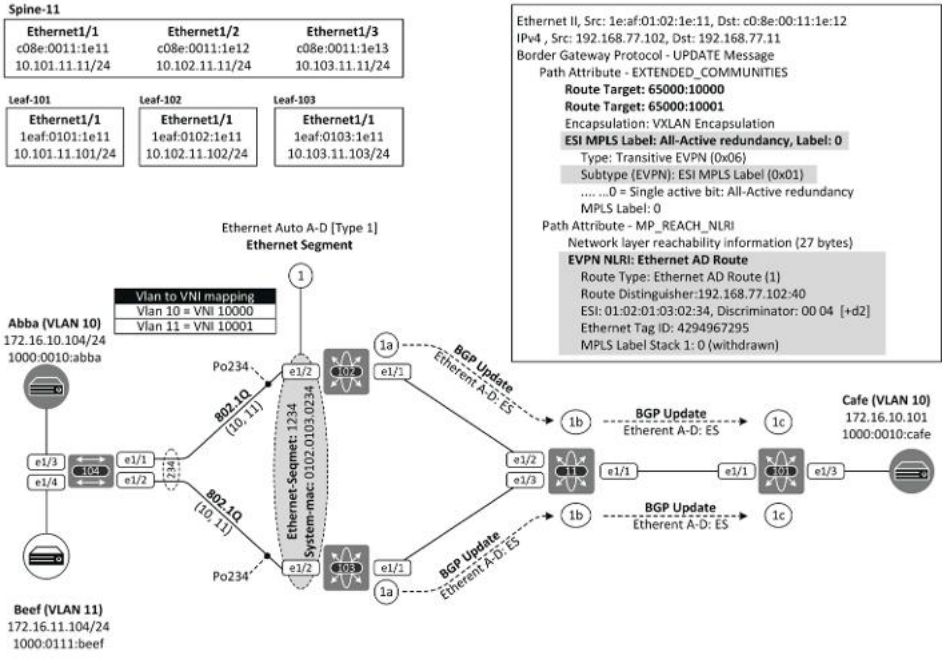


Figure 16-1: *Ethernet A-D per Ethernet Segment (ES) route.*

Ethernet A-D per ES route - Fast Convergence in the all-Active mode

Leaf-102 and Leaf-103 in figure 16-1 belong to the same redundancy group sharing the Ethernet Segment (ES) identified by ES Identifier (ESI) 01.02.01.03.00.02.34.04.d2 (Hex:04.2d=Bin:1234) via interface E1/2 assigned to Port-Channel234. Both interfaces are in the forwarding state (all-Active mode). ASW-104 is connected to them via Port-Channel 234. In failure event, where either Leaf -102 or Leaf-103 loose connection to ES, it has to be signaled to the remote switch Leaf-101. For this purpose, EVPN ESI Multihoming solution uses Ethernet A-D per ES routes BGP Updates.

At the very moment, when Leaf-102 and Leaf-103 joined to the ES, they generate a BGP EVPN *Route-Type 1 (Ethernet A-D)* route, which they advertise to Spine-11. Figure 16-1 shows some of the BGP Path Attributes (BGP PAs) carried with NLRI advertisements.

First, RT:65000:10000 and RT:65000:10001 are the same RT values that are used with MAC/IP NLRIs (BGP EVPN Route-Type 2) concerning VNI 10000 (VLAN 10) and VNI 10001 (VLAN 11). These BGP PAs are carried within the update message because both VLANs are activated in Po234. Because both VNIs 10000 and VNI 10001 are used also in Leaf-101, it imports these NLRIs into the BGP table.

Second, EVPN ESI Multihoming uses either *Single-Active* mode (only one of the links connected to ES is active at a time) or *all-Active* mode (all ES links are active at the same time). Leaf-102 and Leaf-103 are using all-Active mode, which they described to remote peers by setting the *Single-Active* bit to zero in *ESI MPLS Label* Extended Community.

Third, The *EVPN NLRI Ethernet A-D route* describes the Ethernet Segment Identifier (ESI) that is formed from the shared System-MAC and ES value defined under the Port-Channel 234 configuration (configuration can be found from the previous chapter). Note that the *Ethernet Tag Id* must be set maximum value 4294967295 (= HEX: ffff:ffff) and the *MPLS label* must be set to zero (RFC 7432 - section 8.2.1).

Spine-11 is BGP Route-Reflector and it forwards the BGP Update messages to Leaf-101. Example 16-1 illustrates the Leaf-101 BGP table. Highlighted entries are BGP EVPN Ethernet A-D per ES routes sent by Leaf-102 and Leaf-103. When these routes are imported from the BGP Adj-RIB-In into Loc-RIB, Leaf-101 changes the RD to its' own RD 192.168.77.101:65534. Also, notice that the Ethernet A-D per ES is not L2VNI specific update and that is why it is shown as L2VNI 0.

```
Leaf-101# sh bgp l2vpn evpn
<snipped>
   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*|i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.103                  100          0 i
*>i                   192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.101:65534    (L2VNI 0)
*|i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                      192.168.100.103                  100          0 i
*>i                   192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.102:40
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                      192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.102:32777
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.103:40
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                      192.168.100.103                  100          0 i

Route Distinguisher: 192.168.77.103:32777
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.103                  100          0 i
```

Example 16-1: *Leaf-101 BGP table.*

Now the host Abba (IP: 172.16.10.104/MAC: 1000.0010.abba) joins the network (figure 16-2). Both Leaf-102 and Leaf-103 learns the MAC address information from incoming traffic. They both install information into MAC VRF where they exported information into the BGP process and advertises it to BGP EVPN peer Spine-11. The EVPN NLRI MAC Advertisement route (route-type 2) includes the ESI value and *Ethernet Tag Id* among the RD and MAC/IP information. The *ESI type-3* indicates that this is a MAC-based ESI value constructed from the system-MAC and the Local Discriminator value. The Ethernet Tag Id for VLAN-based Service Interface (EVPN Instance is single VLAN) is set to zero (RFC 7432 – section 6.1).

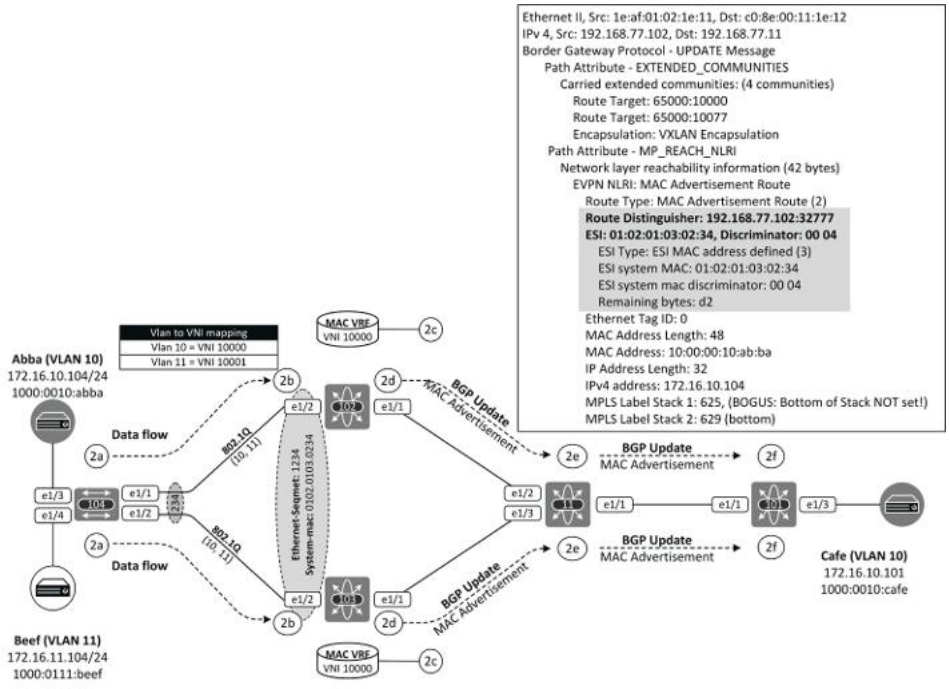


Figure 16-2: *BGP EVPN Route-Type 2 MAC/IP advertisement.*

Examples 16-2 and 16-3 shows that Leaf-101 have received and imported both updates into its BGP table.

```
Leaf-101# sh bgp l2vpn evpn rd 192.168.77.102:32777
<snipped>
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272, version 105
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 2 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
```

```
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5000.0003.0007
      Originator: 192.168.77.102 Cluster list: 192.168.77.111
      ESI: 0301.0201.0302.3400.04d2
```

Example 16-2: *Leaf-101 BGP table – MAC Advertisement originated by Leaf-102.*

```
Leaf-101# sh bgp l2vpn evpn rd 192.168.77.103:32777
<snipped>
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272, version 21
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 2 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.103 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000 10077
      Extcommunity: RT:65000:10000 RT:65000:10077 ENCAP:8 Router
MAC:5000.0004.0007
      Originator: 192.168.77.103 Cluster list: 192.168.77.111
      ESI: 0301.0201.0302.3400.04d2

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

Example 16-3: *Leaf-101 BGP table – MAC Advertisement originated by Leaf-103.*

At this phase, the remote switch Leaf-101 knows that Leaf-102 and Leaf-103 belong to the same redundancy group because they both have advertised the same ESI value 01.02.01.03.02.34.00.04.d2 by using the Ethernet A-D route. Leaf-101 also knows that host Abba is reachable via Leaf-102 and Leaf-103, based on the MAC advertisement route, which in addition the MAC-IP addresses information, includes the same ESI value than what was received from Leaf-102 and Leaf-103 via Ethernet A-D route advertisement. Note, the ESI value is not carried within the MAC-only advertisement route that carries only MAC information as can be seen from capture 16-1 below taken from Leaf-102.

```
Ethernet II, Src: 1e:af:01:02:1e:11, Dst: c0:8e:00:11:1e:12
Internet Protocol Version 4, Src: 192.168.77.102, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 56613, Dst Port: 179, Seq: 166, Ack:
180, Len: 112
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 112
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 89

        Path Attribute - MP_REACH_NLRI
            Type Code: MP_REACH_NLRI (14)
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
```

```
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
             EVPN NLRI: MAC Advertisement Route
               Route Type: MAC Advertisement Route (2)
               Length: 33
               Route Distinguisher: 192.168.77.102:32777
               ESI: 00 00 00 00 00 00 00 00 00 00
               Ethernet Tag ID: 0
               MAC Address Length: 48
               MAC Address: 10:00:00:10:ab:ba
               IP Address Length: 0
               IP Address: NOT INCLUDED
               MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)
```

Capture 16-1: *MAC only Advertisement originated by Leaf-102.*

Now we generate data flow from Abba to Cafe by using ping.

```
Abba#ping 172.16.10.101
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.101, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/53/64 ms
```

Example 16-4: *ping from Abba to Cafe.*

Example 16-5 shows that Leaf-101 has learned the MAC address of Abba from Leaf-102. This is because the LACP hashing algorithm has chosen the interface E1/1 (to Leaf-102) for dataflow between MAC address 1000.0010.abba and 1000.0010.cafe.

```
Leaf-101# show l2route evpn mac evi 10

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

Topology    Mac Address    Prod   Flags      Seq No     Next-Hops
----------- -------------- ------ ---------- ---------- ----------------
10          1000.0010.abba BGP    SplRcv     12         192.168.100.102
10          1000.0010.cafe Local  L,         0          Eth1/3
```

Example 16-5: *Leaf-101 L2RIB.*

Now we generate another dataflow between host Abba and Cafe by using Telnet.

```
Abba#telnet 172.16.10.101
Trying 172.16.10.101 ... Open
Password required, but none set
[Connection to 172.16.10.101 closed by foreign host]
```

Example 16-6: *Telnet from Abba to Cafe.*

Now Leaf-101 MAC address table points to Leaf-103. This is because the LACP hashing algorithm has now chosen the interface E1/2 for this dataflow.

```
Leaf-101# show l2route evpn mac evi 10

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

Topology    Mac Address    Prod   Flags      Seq No     Next-Hops
----------- -------------- ------ ---------- ---------- ----------------
10          1000.0010.abba BGP    SplRcv     12         192.168.100.103
10          1000.0010.cafe Local  L,         0          Eth1/3
```

Example 16-7: *Leaf-101 L2RIB.*

Example 16-8 below shows that the location of MAC address 1000.0010.abba from Leaf-101 perspective has changed 18 times.

```
Leaf-101# show bgp l2vpn evpn 1000.0010.abba
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216, version 87
Paths: (1 available, best #1)
Flags: (0x000212) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop, in rib
             Imported from
192.168.77.102:32777:[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.111)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65000:10000 ENCAP:8 MAC Mobility Sequence:00:18
      Originator: 192.168.77.102 Cluster list: 192.168.77.111
```

Example 16-8: *Leaf-101 BGP table – MAC Mobility.*

The example above illustrates that the latest Ethernet frame from the host defines its' location.

Fast Convergence

In a failure event, where leaf-103 loses its connection to the ES via Po234, it sends a BGP Update message where it withdrawn all routes related to ES. When Leaf-101 receives this message, it removes routes included in the withdrawn message and updates the next-hop addresses.

Example 16-9 illustrates the Leaf-101 BGP table before Leaf-103 generates the withdrawn message caused by link failure. Highlighted entries will be removed by Leaf-101 when it receives the withdrawn message.

```
Leaf-101# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 152, Local Router ID is 192.168.77.101
<snipped>

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.102                   100          0 i
*|i                   192.168.100.103                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.100.103                   100          0 i
*|i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                      192.168.100.103                   100          0 i
*>i                   192.168.100.102                   100          0 i

Route Distinguisher: 192.168.77.101:65534    (L2VNI 0)
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                      192.168.100.102                   100          0 i
*|i                   192.168.100.103                   100          0 i

Route Distinguisher: 192.168.77.102:40
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                      192.168.100.102                   100          0 i

Route Distinguisher: 192.168.77.102:32777
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.102                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                      192.168.100.102                   100          0 i

Route Distinguisher: 192.168.77.102:32778
*>i[2]:[0]:[0]:[48]:[1000.0111.beef]:[32]:[172.16.11.104]/272
                      192.168.100.102                   100          0 i

Route Distinguisher: 192.168.77.103:40
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                      192.168.100.103                   100          0 i

Route Distinguisher: 192.168.77.103:32777
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.103                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.100.103                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                      192.168.100.103                   100          0 i

Route Distinguisher: 192.168.77.103:32778
*>i[2]:[0]:[0]:[48]:[1000.0111.beef]:[32]:[172.16.11.104]/272
                      192.168.100.103                   100          0 i
```

Example 16-9: *BGP table on Leaf-101 before Leaf-103 withdrawn message.*

Capture 16-2 shows the Unreachable NLRIs withdrawn by Leaf-103 after a link failure (Po234).

```
Ethernet II, Src: c0:8e:00:11:1e:11 (c0:8e:00:11:1e:11), Dst: 1e:af:01:01:1e:11
(1e:af:01:01:1e:11)
Internet Protocol Version 4, Src: 192.168.77.11, Dst: 192.168.77.101
Transmission Control Protocol, Src Port: 179, Dst Port: 56294, Seq: 56, Ack:
20, Len: 111
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 111
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 88
    Path attributes
        Path Attribute - MP_UNREACH_NLRI
            Type Code: MP_UNREACH_NLRI (15)
            Length: 84
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Withdrawn routes (81 bytes)
                EVPN NLRI: Ethernet AD Route
                    Route Type: Ethernet AD Route (1)
                    Length: 25
                    Route Distinguisher: 192.168.77.103:40
                    ESI: 01:02:01:03:02:34, Discriminator: 00 04
                    Ethernet Tag ID: 4294967295
                    MPLS Label Stack 1: 0 (withdrawn)
                EVPN NLRI: Ethernet AD Route
                    Route Type: Ethernet AD Route (1)
                    Length: 25
                    Route Distinguisher: 192.168.77.103:32777
                    ESI: 01:02:01:03:02:34, Discriminator: 00 04
                    Ethernet Tag ID: 0
                    MPLS Label Stack 1: 0 (withdrawn)
                EVPN NLRI: Ethernet AD Route
                    Route Type: Ethernet AD Route (1)
                    Length: 25
                    Route Distinguisher: 192.168.77.103:32778
                    ESI: 01:02:01:03:02:34, Discriminator: 00 04
                    Ethernet Tag ID: 0
                    MPLS Label Stack 1: 0 (withdrawn)
```

Capture 16-2: *BGP Update message (withdrawn) sent by Leaf-103.*

After receiving the BGP withdrawn message from Leaf-103, Leaf removes all the withdrawn routes and updates the next-hop addresses.

```
Leaf-101# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
<snipped>

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                      192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                      192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.101:65534    (L2VNI 0)
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
```

```
                     192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.102:40
*>i[1]:[0301.0201.0302.3400.04d2]:[0xffffffff]/152
                     192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.102:32777
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                     192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                     192.168.100.102                  100          0 i

Route Distinguisher: 192.168.77.102:32778
*>i[2]:[0]:[0]:[48]:[1000.0111.beef]:[32]:[172.16.11.104]/272
                     192.168.100.102                  100          0 i
```

Example 16-10: *BGP table on Leaf-101 after Leaf-103 withdrawn message.*

Example 16-11 shows the L2RIB of Leaf-101 before changes and example 16-12 after changes. Even though there has not been any MAC address moving event, the MAC address table information is updated. Leaf-101 knows that even host Abba is reachable via Leaf-102 because the MAC advertisement of host Abba has the same ESI than what was previously received from Leaf-102.

```
Leaf-101# sh system internal l2fwder mac | i abba
*    10    1000.0010.abba    static   -  F  F  nve-peer1 192.168.100.103
```

Example 16-11: *Leaf-101 MAC address table before withdrawn.*

```
Leaf-101# sh system internal l2fwder mac | i abba
*    10    1000.0010.abba    static   -  F  F  nve-peer1 192.168.100.102
```

Example 16-12: *Leaf-101 MAC address table after withdrawn.*

Load Balancing (Aliasing)

Figure 16-3 illustrates the situation where Leaf-102 and Leaf 103 are already sent the *BGP EVPN Ethernet A-D ES route* where they describe the local redundancy mode (all-Active) used with Ethernet Segment to remote peer Leaf-101. This way Leaf-101 knows that both Leaf-102 and Leaf-103 are able to forward data to clients behind the Ethernet Segment (ES). However, Leaf-101 does not know which VNIs or clients are reachable through particular ES. For VNI information, Leaf-102 and Leaf-103 originate *BGP EVPN Ethernet A-D EVI/ES routes,* where they tell that RD 192.168.77.102/103:32777 (used with MAC advertisement route for VNI 10000) can be found behind ES 0301.0201.0302.3400.004d2. Leaf-101 imports these routes based on the RT 65000:10000 (used with VNI10000).

Based on these two Ethernet A-D routes (ES + EVI/ES) Leaf-101 know that MAC/IP routes advertised with RD 192.168.102/103:32777 are reachable via both Leaf-102 and Leaf-103. Next, host Abba joins the network and sends a GARP message. The message reaches ASW-104, which LACP hashing algorithm selects interface E1/1. This means that local MAC learning is done only by Leaf-102. It installs the route to MAC VRF and exports it into BGP process where it is sent as BGP EVPN MAC advertisement route update. Leaf-101 learns the host Abba MAC address only from the Leaf-102, but still based on Ethernet A-D ES and

EVI/ES messages it knows that MAC 1000.0010.abba is reachable through the Leaf-102 and Leaf-103. The partial output in example 16-13 shows that the MAC/IP address of host Abba is reachable through the Leaf-102 and Leaf-103 and data towards Abba can be load balanced.

```
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                     192.168.100.102                  100          0 i
*|i                  192.168.100.103                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.100.103                  100          0 i
*|i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                     192.168.100.103                  100          0 i
*>i                  192.168.100.102                  100          0 i
```

Example 16-13: *Partial BGP table of Leaf-101.*

This method, when using the all-Active mode, is also called *Aliasing.* If Single-Active mode is used the term *Backup Path* is used.

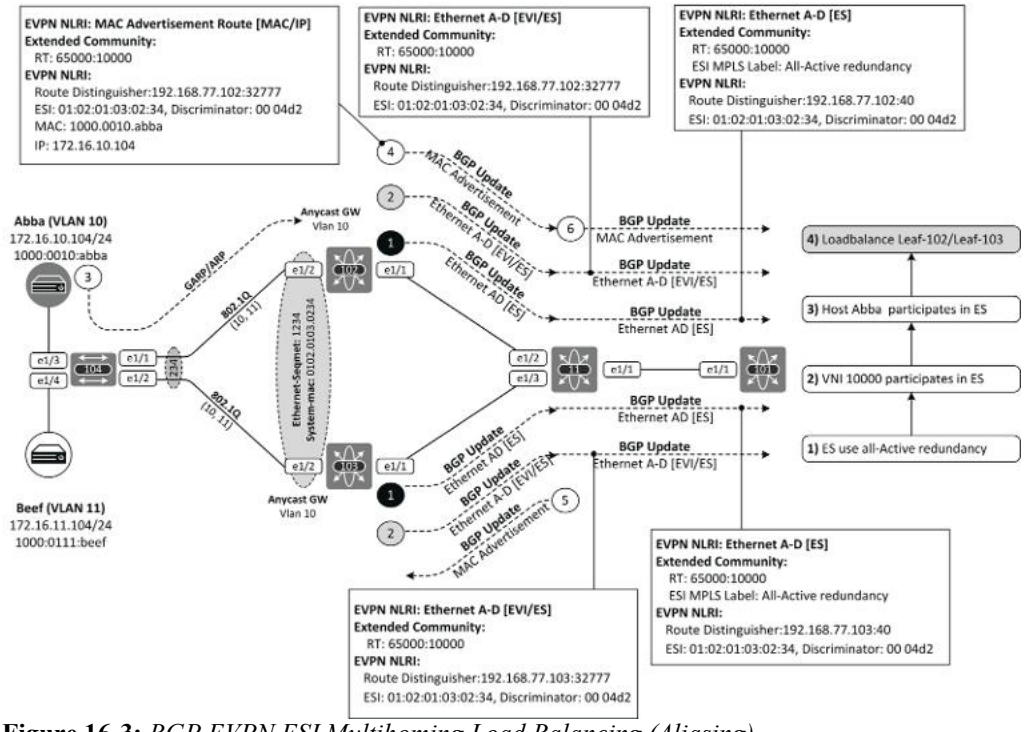


Figure 16-3: *BGP EVPN ESI Multihoming Load Balancing (Aliasing).*

Summary

This chapter describes the Fast Convergence mechanism used in BGP EVPN ESI Multihoming solution by using Ethernet A-D ES route. In addition, this chapter introduces the Load Balancing method, which relies on the information received via Ethernet A-D EVI/ES and Ethernet A-D ES route together with the MAC Advertisement route.

References:

[RFC 7432] A. Sajassi et al., “BGP MPLS-Based Ethernet VPN”, RFC 7432, February 2015.

Chapter 17: EVPN ESI Multihoming - Data Flows and link failures

This chapter explains the EVPN ESI Multihoming data flows. The first section explains the Intra-VNI flows (L2VNI) Unicast traffic and the Second section introduces BUM traffic. Figure 17-1 shows the topology and addressing schemes used in this chapter.

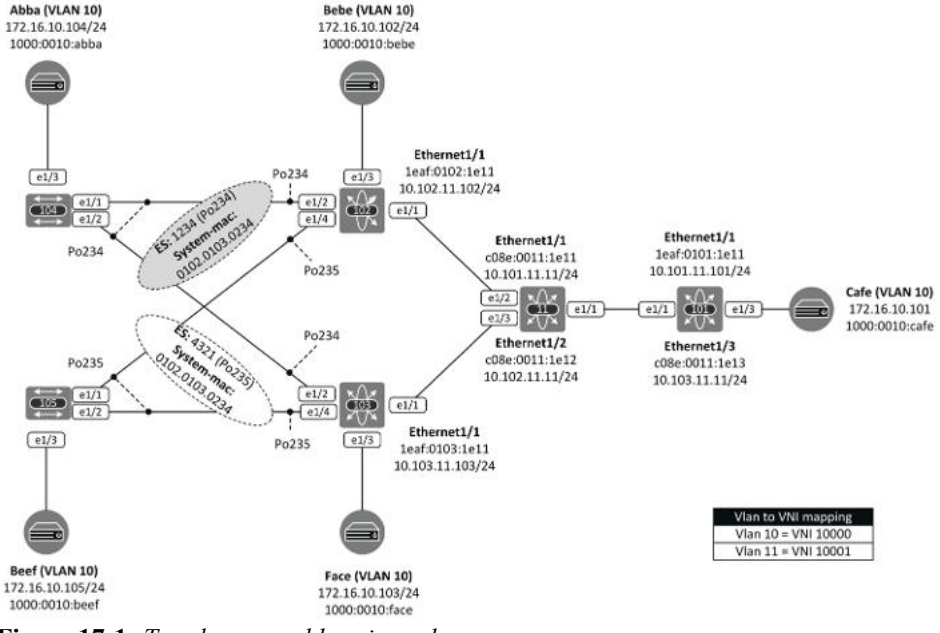


Figure 17-1: *Topology an addressing scheme.*

Introduction

Examples 17-1, 17-2 and 17-3 show the MAC address tables of leaf switches in a stable situation where all inter-switch links are up. In order to generate data flows to network, Hosts Abba and Beef sends ICMP requests to all hosts (Abba/Beef/Bebe/Face/Cafe) and to VLAN 10 AGW address in every five seconds. Note that the MAC address table is updated based on current data flow. This means that if e.g. host Abba has only one data flow in time T1, the LACP hash algorithm might choose to use the only link to Leaf-102. This means that Leaf-103 learns the MAC address only via BGP from Spine-11.

The data path between host Abba and Beef use optimal path either via Leaf-102 or via Leaf-103 depending on an LACP hashing algorithm result in ASW-104 and ASW-105. When considering data path from Abba to orphan host Bebe, the optimal path is via Leaf-102 and sub-optimal path via Leaf-103 > Spine-11 > Leaf-102. This depends on the result of the LACP hash algorithm. The same rule applies to data paths between host Abba to Face and from Beef to either orphan hosts. Return traffic from orphan hosts will use the optimal path. Data paths from host Cafe to Abba and Beef can be load balanced/per flow.

```
Leaf-102# sh sys int l2fwder mac | i abba|beef|bebe|face|cafe
*    10    1000.0010.cafe    static   -          F     F  nve-peer2
192.168.100.101
*    10    1000.0010.beef   dynamic   02:02:15   F     F       Po235
*    10    1000.0010.abba   dynamic   02:02:17   F     F       Po234
*    10    1000.0010.bebe   dynamic   02:02:17   F     F     Eth1/3
*    10    1000.0010.face    static   -          F     F  nve-peer1
192.168.100.103
```

Example 17-1: *MAC address table of Leaf-102.*

```
Leaf-103#  sh sys int l2fwder mac | i abba|beef|bebe|face|cafe
*    10    1000.0010.cafe    static   -          F     F  nve-peer2
192.168.100.101
*    10    1000.0010.beef   dynamic   02:03:10   F     F       Po235
*    10    1000.0010.abba   dynamic   02:03:12   F     F       Po234
*    10    1000.0010.bebe    static   -          F     F  nve-peer1
192.168.100.102
*    10    1000.0010.face   dynamic   02:03:11   F     F     Eth1/3
```

Example 17-2: *MAC address table of Leaf-103.*

```
Leaf-101# sh sys int l2fwder mac | i abba|beef|bebe|face|cafe
*    10    1000.0010.cafe   dynamic   02:27:47   F     F     Eth1/3
*    10    1000.0010.beef    static   -          F     F  nve-peer2
192.168.100.102
*    10    1000.0010.abba    static   -          F     F  nve-peer1
192.168.100.103
*    10    1000.0010.bebe    static   -          F     F  nve-peer2
192.168.100.102
*    10    1000.0010.face    static   -          F     F  nve-peer1
192.168.100.103
```

Example 17-3: *MAC address table of Leaf-101.*

L2RIB of Leaf-102 shows that MAC addresses of local hosts Abba (ES 1234) and Beef (ES4321) are learned from connected Port-Channels and in addition to the redundancy group peer switch Leaf-103 via BGP.

```
Leaf-102# sh l2route mac all | i abba|beef|bebe|face|cafe
10          1000.0010.abba Local  L,Dup,PF,      39          Po234
10          1000.0010.abba BGP    Dup,PF,SplRcv 38          192.168.100.103
10          1000.0010.bebe Local  L,             0          Eth1/3
10          1000.0010.beef Local  L,Dup,PF,      38          Po235
10          1000.0010.beef BGP    Dup,PF,SplRcv 37          192.168.100.103
10          1000.0010.cafe BGP    SplRcv        0           192.168.100.101
10          1000.0010.face BGP    SplRcv        0           192.168.100.103
```

Example 17-4: *L2RIB on Leaf-102.*

Intra-VNI (L2VNI): Unicast Traffic

Scenario 1: Link e1/4 down on Leaf-102

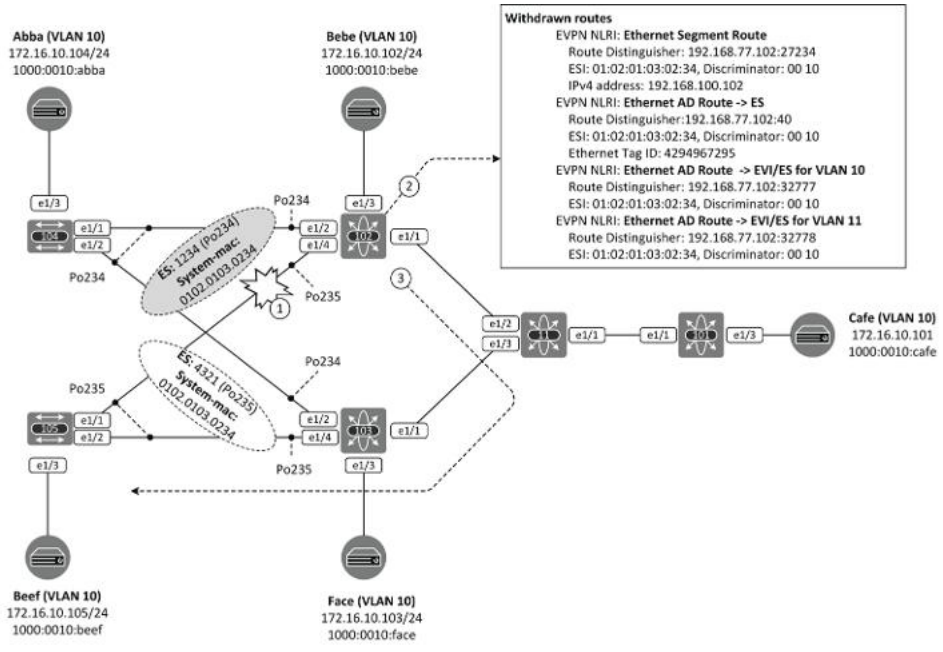


Figure 17-2: *Link failure on Leaf-102.*

When inter-switch link e1/4 connected to ASW-105 goes down on Leaf-102 (figure 17-2) it has to remove itself from the redundancy group used for ESI: 0301.0201.0302.3400.10e1 by withdrawing an Ethernet Segment Route (BGP EVPN Route-Type 4). This withdrawn message is only processed by Leaf-103 on the same redundancy group (based on Route-Target). When Leaf-103 receives the message, it updates the Ethernet Segment information. Example 17-5 shows that Leaf-103 is now Designated Forwarder (DF) for all active VLANs on ESI: 0301.0201.0302.3400.10e1 while Leaf-102 is still DF for VLAN 10 in ESI 0301.0201.0302.3400.04d2.

```
Leaf-103# sh nve ethernet-segment
ESI: 0301.0201.0302.3400.04d2
   Parent interface: port-channel234
  ES State: Up
  Port-channel state: Up
  NVE Interface: nve1
   NVE State: Up
   Host Learning Mode: control-plane
  Active Vlans: 10-11
   DF Vlans: 11
   Active VNIs: 10000-10001
  CC failed for VLANs:
  VLAN CC timer: 0
  Number of ES members: 2
  My ordinal: 1
  DF timer start time: 00:00:00
  Config State: config-applied
```

```
  DF List: 192.168.100.102 192.168.100.103
  ES route added to L2RIB: True
  EAD/ES routes added to L2RIB: True
  EAD/EVI route timer age: not running
---------------------------------------

ESI: 0301.0201.0302.3400.10e1
   Parent interface: port-channel235
  ES State: Up
  Port-channel state: Up
  NVE Interface: nve1
   NVE State: Up
   Host Learning Mode: control-plane
  Active Vlans: 10-11
   DF Vlans: 10-11
   Active VNIs: 10000-10001
  CC failed for VLANs:
  VLAN CC timer: 0
  Number of ES members: 1
  My ordinal: 0
  DF timer start time: 00:00:00
  Config State: config-applied
  DF List: 192.168.100.103
  ES route added to L2RIB: True
  EAD/ES routes added to L2RIB: True
  EAD/EVI route timer age: not running
---------------------------------------
```

Example 17-5: *ES information on Leaf-103.*

In addition, Leaf-102 has to inform remote leafs that it cannot be used for load balancing purposes concerning destination MAC addresses that are advertised with ESI: 0301.0201.0302.3400.10e1. This is done by using the Ethernet A-D ES route (BGP EVP Route-Type 1). Leaf-102 also withdrawn all the MAC addresses participating in VNI 10 (VLAN 10) and VNI 11 (VLAN 11) that have ESI: 0301.0201.0302.3400.10e1 value attaches to it by using Ethernet A-D EVI/ES route. This process is called *mass withdrawn*. As a reaction to the message, Leaf-101 updates the next-hop information.

Example 17-6 shows the BGP table of Leaf-101 before withdrawn messages and the example 17-7 after withdrawn message. ESI: 0301.0201.0302.3400.10e1 and host Beef is only advertised by Leaf-103.

```
Leaf-101# sh bgp l2vpn evpn vni-id 10000
<snipped>
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                     192.168.100.102                   100          0 i
*|i                  192.168.100.103                   100          0 i
*|i[1]:[0301.0201.0302.3400.10e1]:[0x0]/152
                     192.168.100.103                   100          0 i
*>i                  192.168.100.102                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.100.103                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.bebe]:[0]:[0.0.0.0]/216
                     192.168.100.102                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
                     192.168.100.102                   100          0 i
*>l[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.101                   100      32768 i
*>i[2]:[0]:[0]:[48]:[1000.0010.face]:[0]:[0.0.0.0]/216
                     192.168.100.103                   100          0 i
```

```
*|i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                     192.168.100.103                  100          0 i
*>i                  192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.bebe]:[32]:[172.16.10.102]/272
                     192.168.100.102                  100          0 i
*|i[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[172.16.10.105]/272
                     192.168.100.103                  100          0 i
*>i                  192.168.100.102                  100          0 i
*>l[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
                     192.168.100.101                  100      32768 i
*>i[2]:[0]:[0]:[48]:[1000.0010.face]:[32]:[172.16.10.103]/272
                     192.168.100.103
```

Example 17-6: *BGP table on Leaf-101 before withdrawn.*

```
Leaf-101# sh bgp l2vpn evpn vni-id 10000
<snipped>
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>i[1]:[0301.0201.0302.3400.04d2]:[0x0]/152
                     192.168.100.102                  100          0 i
*|i                  192.168.100.103                  100          0 i
*>i[1]:[0301.0201.0302.3400.10e1]:[0x0]/152
                     192.168.100.103                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.100.103                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.bebe]:[0]:[0.0.0.0]/216
                     192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
                     192.168.100.103                  100          0 i
*>l[2]:[0]:[0]:[48]:[1000.0010.cafe]:[0]:[0.0.0.0]/216
                     192.168.100.101                  100      32768 i
*>i[2]:[0]:[0]:[48]:[1000.0010.face]:[0]:[0.0.0.0]/216
                     192.168.100.103                  100          0 i
*|i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.104]/272
                     192.168.100.103                  100          0 i
*>i                  192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.bebe]:[32]:[172.16.10.102]/272
                     192.168.100.102                  100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.beef]:[32]:[172.16.10.105]/272
                     192.168.100.103                  100          0 i
*>l[2]:[0]:[0]:[48]:[1000.0010.cafe]:[32]:[172.16.10.101]/272
                     192.168.100.101                  100      32768 i
*>i[2]:[0]:[0]:[48]:[1000.0010.face]:[32]:[172.16.10.103]/272
                     192.168.100.103                  100          0 i
```

Example 17-7: *BGP table on Leaf-101 after withdrawn.*

As the last step, Leaf-102 updates its own MAC address table. Now host Beef is only learned from Leaf-103 via BGP.

```
Leaf-102# sh sys int l2fwd mac | i abba|beef|bebe|face|cafe
*    10    1000.0010.cafe    static   -          F     F  nve-peer2
192.168.100.101
*    10    1000.0010.beef    static   -          F     F  nve-peer1
192.168.100.103
*    10    1000.0010.abba   dynamic   03:12:26   F     F      Po234
*    10    1000.0010.bebe   dynamic   03:12:26   F     F     Eth1/3
*    10    1000.0010.face    static   -          F     F  nve-peer1
192.168.100.103
```

Example 17-8: *BGP table on Leaf-102 after link failure.*

Scenario 2: Core link down on Leaf-102.

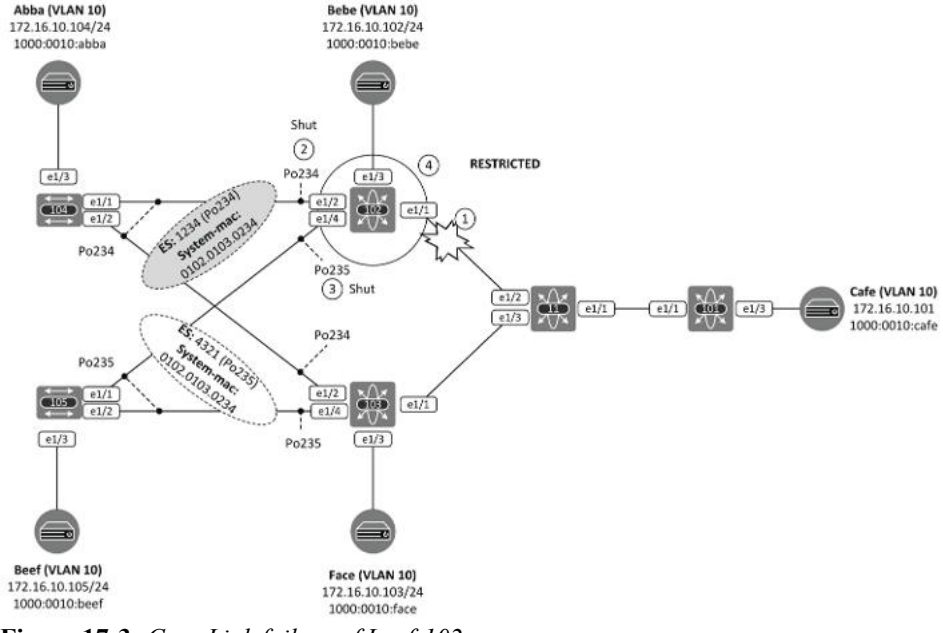


Figure 17-3: *Core Link failure of Leaf-102.*

In the case where Leaf-102 loses all of its core links, it restricts itself from the network by shutting down all the links that participate in any Ethernet Segment. The result is that also orphan host Bebe is restricted from the network even though its uplink interface e1/3 stays up. traffic between Abba, Beef, Face, and Cafe is now switched via Leaf-103.

```
Leaf-102# sh interface port-channel 234 | i 234
port-channel234 is down (NVE core link down)

Leaf-102# sh interface port-channel 235 | i 235
port-channel235 is down (NVE core link down)
```

Example 17-9: *Core link failure on Leaf-102.*

Examples 17-10 and 17-11 shows that orphan host Bebe is removed from both MAC address table and L2RIB by Leaf-103.

```
Leaf-103# sh sys int l2fwd mac | i abba|beef|bebe|face|cafe
*    10    1000.0010.cafe    static   -          F      F  nve-peer2
192.168.100.101
*    10    1000.0010.beef    dynamic  00:03:51   F      F       Po235
*    10    1000.0010.abba    dynamic  00:02:04   F      F       Po234
*    10    1000.0010.face    dynamic  03:35:35   F      F     Eth1/3
```

Example 17-10: *MAC address table of Leaf-103.*

```
Leaf-103# sh l2route evpn mac all | i abba|beef|bebe|face|cafe
10          1000.0010.abba Local  L,            0          Po234
10          1000.0010.beef Local  L,            0          Po235
10          1000.0010.cafe BGP    SplRcv        0          192.168.100.101
10          1000.0010.face Local  L,            0          Eth1/3
```

Example 17-11: *L2RIB of Leaf-103.*

Intra-VNI (L2VNI): Broadcast, Unknown Unicast and Multicast (BUM) traffic

Scenario 1: Traffic flow from Designated Forwarder

Figure 17-4 illustrates the situation where host Abba in vlan 10 sends an Ethernet frame with an unknown destination MAC address. The LACP hash algorithm of Leaf-104 forwards the frame out of the interface e1/1 to Leaf-102 (1). Leaf-102 forwards frame out of the e1/3 to the orphan host Bebe (2). Because Leaf-102 is Designated Forwarder (DR) for VLAN 10, it also forwards the frame to the Ethernet Segments with ESI: 0301.0201.0302.3400.10e1 where vlan 10 is allowed (3). For BUM traffic, all leaf switches use Multicast group 238.0.0.10 where Spine-11 is Rendezvous Point (RP). Leaf-102 encapsulates the Ethernet frame with new Ethernet/IP (238.0.0.10) /UDP/VXLAN headers and sends it to Spine-11 (4). Spine-11 receives the BUM frame with the destination IP 238.0.0.10. It forwards frames based on 238.0.0.10 Outgoing Interface List (OIL) to Leaf-101 and Leaf-103 (5). Leaf-101 forwards frames to host Cafe. Leaf-103 receives the frames and decapsulates it. Based on VXLAN header VNI 10000 value, Leaf-103 knows that the frame needs to be forwarded to vlan 10 and it forwards frames out to host Face (6). Vlan 10 is active in the Ethernet Segments with ESI: 0301.0201.0302.3400.10e1 and with ESI: 0301.0201.0302.3400.04d2, however Leaf-103 does not forward frames received from remote leaf to ES where it is not Designated Forwarder (7-8).

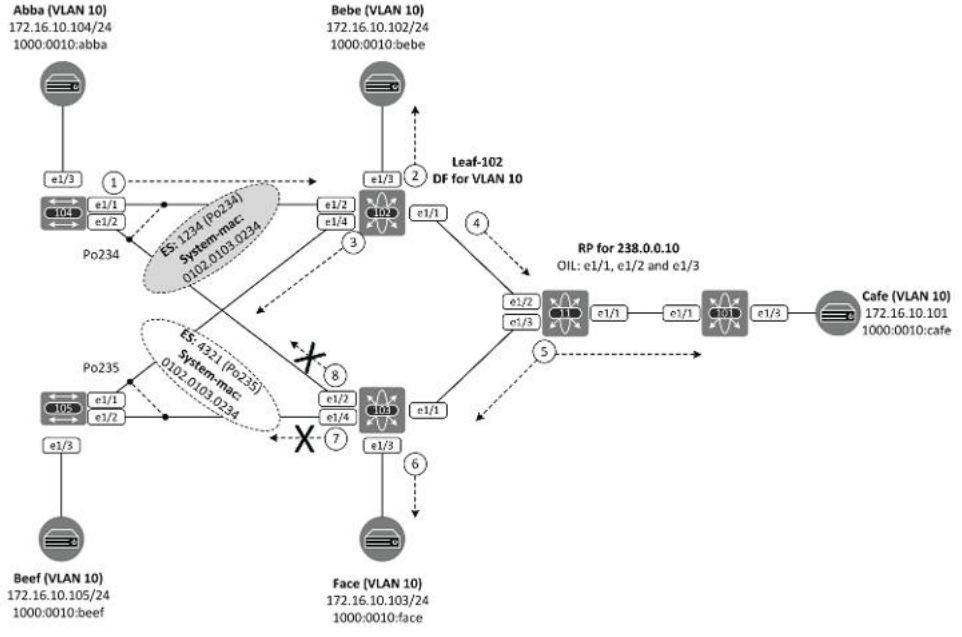


Figure 17-4: *BUM from DF perspective.*

Scenario 2: Traffic flow from non-Designated Forwarder

Figure 17-5 illustrates the situation where host Abba in vlan 10 sends an Ethernet frame with an unknown destination MAC address. The LACP hash algorithm of Leaf-104 now forwards the frame out of the interface e1/2 to Leaf-103 (1). Leaf-103 forwards frame out of the e1/3 to the orphan host Face (2). In addition, Leaf-103 forwards the frame received from local ESI: 0301.0201.0302.3400.04d2 to the local ESI: 0301.0201.0302.3400.10e1 where vlan 10 is allowed (3). Leaf-103 encapsulates the Ethernet frame with new Ethernet/IP (238.0.0.10) /UDP/VXLAN headers and sends it to Spine-11 (4). Spine-11 receives the BUM frame with the destination IP 238.0.0.10. It forwards frames based on 238.0.0.10 Outgoing Interface List (OIL) to Leaf-101 and Leaf-102 (5). Leaf-101 forwards frames to host Cafe. Leaf-102 receives the frames and decapsulates it. Based on VXLAN header VNI 10000 value, Leaf-102 knows that the frame needs to be forwarded to vlan 10 and it forwards frames out to host Bebe (6). Vlan 10 is active in the Ethernet Segments with ESI: 0301.0201.0302.3400.10e1 and with ESI: 0301.0201.0302.3400.04d2, however even though Leaf-102 is the DF for VLAN 10 it does not forward frames received from remote leaf to either local ES (7-8).

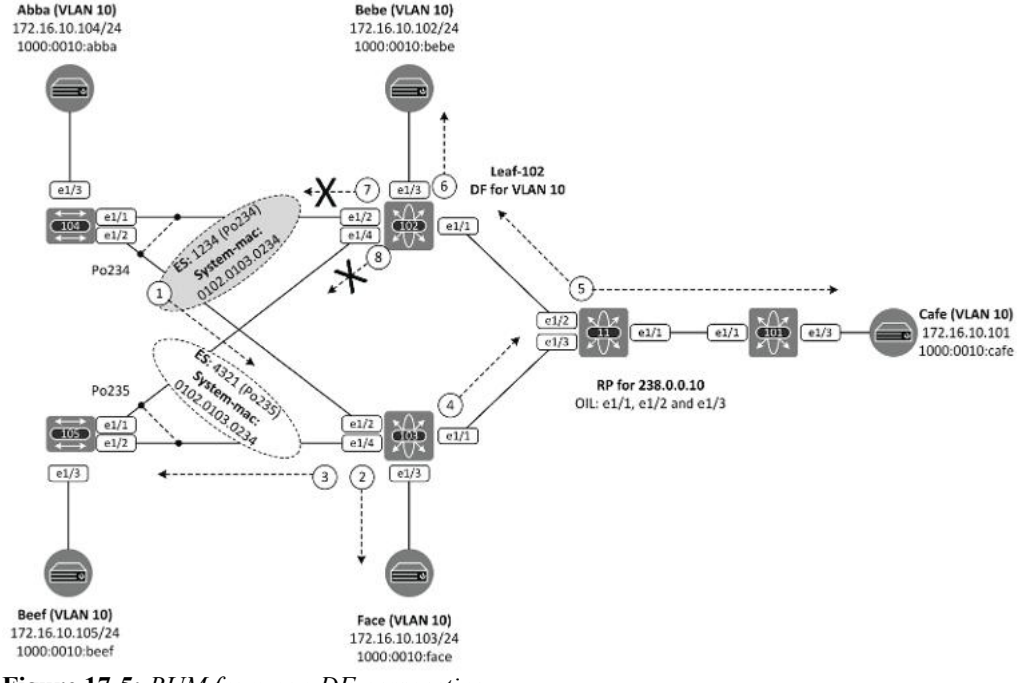


Figure 17-5: *BUM from non-DF perspective.*

Scenario 3: Traffic flow from Remote Leaf

Figure 17-6 illustrates the situation where the host Cafe in vlan 10 sends an Ethernet frame with an unknown destination MAC address. Leaf-101 encapsulates the Ethernet frame with new Ethernet/IP (238.0.0.10) /UDP/VXLAN headers and sends it to Spine-11 (1). Spine-11 receives the BUM frame with the destination IP 238.0.0.10. It forwards frames based on 238.0.0.10 Outgoing Interface List (OIL) to Leaf-102 and Leaf-103 (2). Leaf-102 and Leaf 103 receives the frame and decapsulates it and forwards frame to local orphan host Bebe and Face (3, 6). Vlan 10 is active in the Ethernet Segments with ESI: 0301.0201.0302.3400.10e1 and with ESI: 0301.0201.0302.3400.04d2, Leaf-102 is the DF for VLAN 10 and it forwards frames received from remote leaf to the ESs. Leaf-103 (non-DF for vlan 10) does not forward frames received from remote leaf to either local ES (7-8).

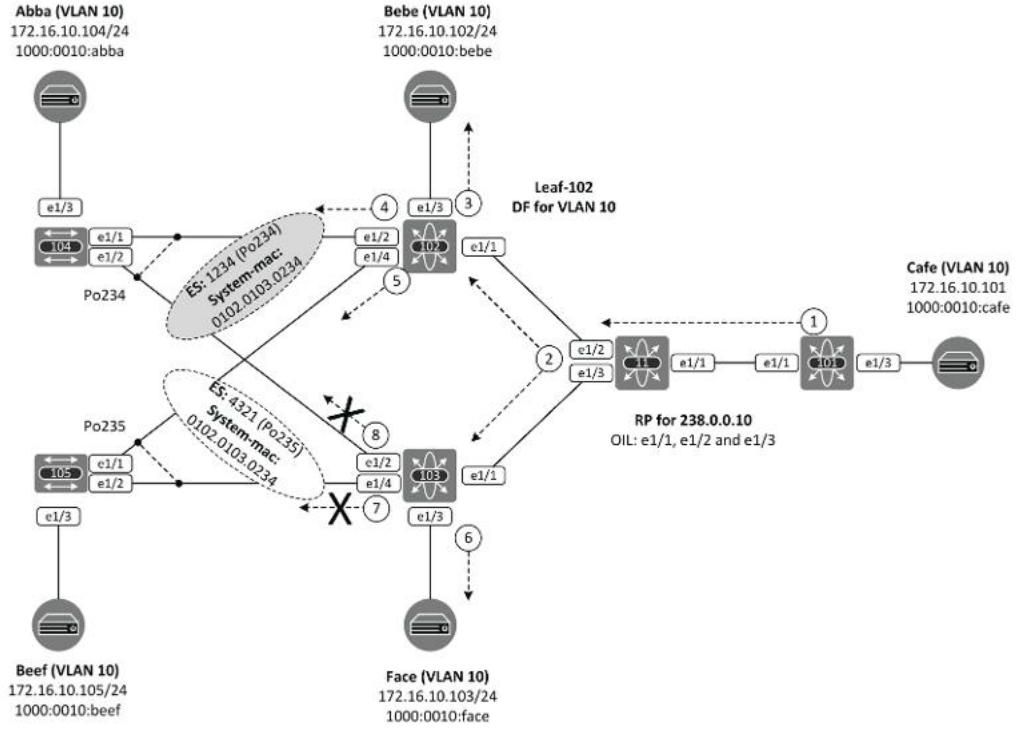


Figure 17-6: *BUM from remote leaf perspective.*

CHAPTER 18: VXLAN EVPN Multi-Site

This chapter introduces the VXLAN EVPN Multi-Site (EVPN-MS) architecture for interconnecting EVPN Domains. The first section discusses the limitations of flat VXLAN EVPN fabric and the improvements that can be achieved with EVPN-MS. The second section focuses on the technical details of EVPN-MS solutions by using various configuration examples and packet captures.

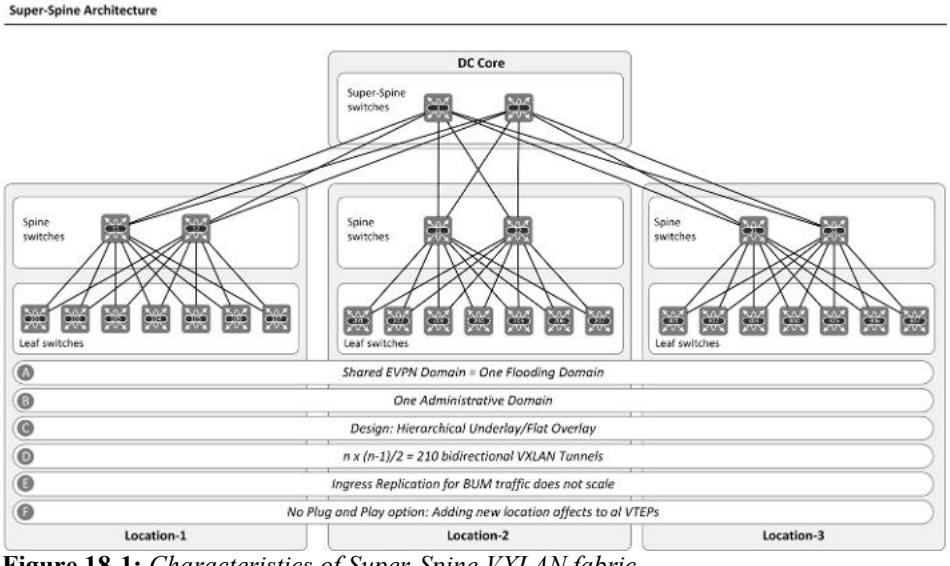


Figure 18-1: *Characteristics of Super-Spine VXLAN fabric.*

Shared EVPN domain limitations

Figure 18-1 depicts the example BGP EVPN implementation that includes three Datacenters in three different locations. Each DC has seven Leaf-switches and two Spine-switches. For the DC-interconnect, there is a pair of Super-Spine switches. All VLANs/VNIs has to be available in each Leaf switch no matter of location. This means that full mesh NVE peering between each Leaf switches is required.

Even though the physical Underlay Network in this solution is hierarchical, the Overlay Network on top of it is flat i.e. there is one shared geographically dispersed EVPN domain (one L2 flooding domain). From the Underlay Network perspective, this means that the routing design and routing protocol choice should be consistent throughout the EVPN domain, otherwise there will be a complex and hard to manage IP prefix redistribution from one protocol to another. The same design requirements apply also to multi-destination traffic, the BUM traffic forwarding has to be based on the same solution throughout the EVPN domain. The Ingress-Replication (IR) does not scale well in large scale VXLAN EVPN fabric. In this example network, there are 21 Leaf switches. Each switch has 20 NVE peers, so if IR is used for BUM traffic forwarding, the copy of the multi-destination frame/packet has to be individually sent to all NVE peers. This might lead to a situation where BUM traffic flows disturb the actual application data traffic on an uplink of sending switch. This is why the Multicast enabled Underlay-Network is preferred in large-scale solutions. In summary, a large scale VXLAN EVPN fabric can't rely on IP-only Underlay Network.

From the Overlay-Network perspective, the amount of bi-directional VXLAN tunnels on large-scale solutions also has its challenges. Even though the example here consists of only 21 Leaf switches, there are 20 NVE peering per switch and 210 bi-directional VXLAN tunnels [*n* x (*n*-1)/2)]. If the count of Leaf switches is doubled from 21 to 42 (41 NVE peers per Leaf), the bi-directional tunnel count will rise up from 210 to 861. If each switch has 41 NVE peers, it also means 41 possible next-hop addresses per Leaf switch. In the case of VM moves inside one location, every single Leaf has to update the next-hop table.

There are no real plug-and-play capabilities in this solution. When either adding devices to infrastructure or adding a whole new site, each existing Leaf switch will build an NVE peering with an added device(s). The opposite happens when devices are removed from the infrastructure, each remaining Leaf switch will tear down the tunnels.

From the administrative perspective, this solution is managed as one entity. This excludes the design where e.g. customer wants to manage one DC while the service provider manages the other DC owned by the same customer.

EVPN Multi-Site Architecture Introduction

Figure 18-2 includes the same physical topology used in the previous example with an additional pair of Border Gateways (BGWs) in each site. The one big VXLAN EVPN fabric is now divided into the set of smaller fabrics, which are connected through the BGWs into DC Core routers/switches in the shared Common EVPN domain. This brings back the hierarchy into the Overlay Network.

Each fabric forms an individual management domain that has dedicated underlay routing architecture (routing protocol, interface IP-addressing and so on). In addition, either Multicast or Ingress-Replication (IR) can be used independently, one fabric can use Multicast based solution while the other fabric can use IR. Site-local Leaf switches form a bi-directional NVE peering (VXLAN tunnels) only with an intra-site Leaf switches and with a BGW switch. In Addition, Local BGW switches form an NVE peering between each other and also between site-external BGW switches.

The intra-site Underlay Network can use any IGP protocol or BGP for routing exchange while the Overlay Network routing use BGP (L2VPN EVPN afi). The Underlay Network routing protocol in the Common EVPN domain between BGW switches and DC Core switch/router is eBGP (IPv4 unicast afi) while the eBGP (L2VPN EVPN afi) is used in the Overlay Network.
Because each site operates as an individual fabric, there are no Control Plane relationship requirements between sites. Connecting a new site to DC Core routers does not generate any major Control Plane changes from the protocol perspective (such as new NVE or routing protocol peering) in intra-site Leaf switches on remote sites. New BGWs will only establish both Underlay and Overlay network eBGP peering with DC Core routers and forms NVE peering with the existing BGW switches. After that, BGW switched can exchange routing information. In this manner, the EVPN Multi-Site solution is plug-and-play capable.
There are two BGWs per site in figure 18-2 but this is not the limitation. NX-OS 9.3.x support a maximum of six BGWs per site.

Next sections introduce the VXLAN EVPN Multi-Site solution in detail.

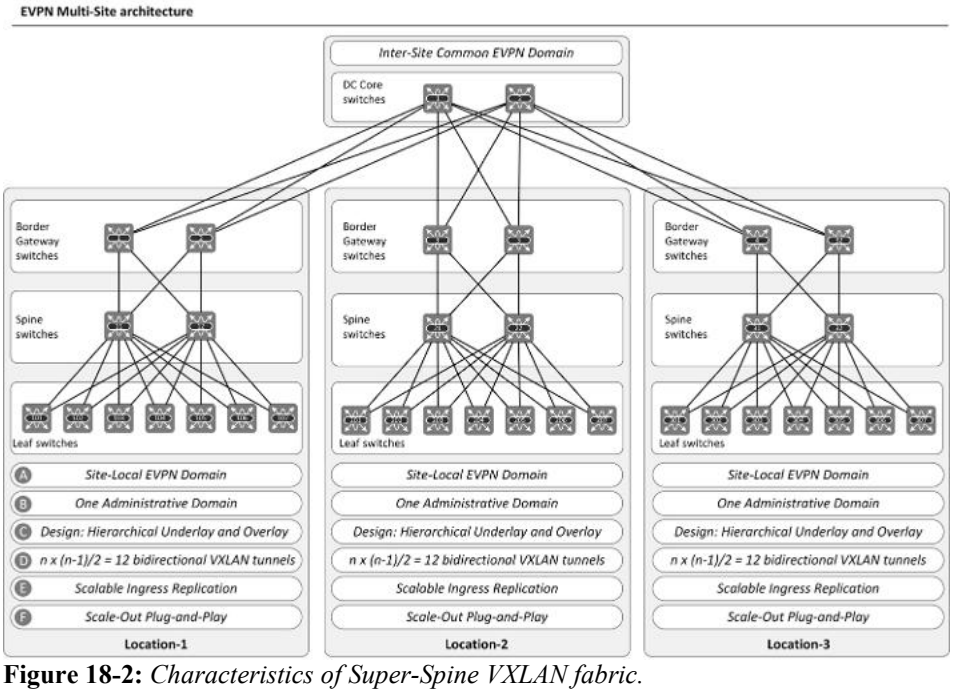


Figure 18-2: *Characteristics of Super-Spine VXLAN fabric.*

Intra-Site EVPN Domain (Fabric)

This section shortly introduces the intra-site example solution used sites 12 and 34 (figure 18-3). Both sites use the same Underlay and Overlay Network design. OSPF (RID 192.168.0.dev-number/32) is used for IP-connectivity between nodes and all Loopback address information is advertised internally. PIM BiDir is enabled on an Underlay Network and Spine switches are defined as a Pseudo Rendezvous Point. BGP L2EVPN peering is done between Loopback interface 77 (192.168.77.dev-number/32). NVE interfaces use IP address of Loopback 100 (192.168.100.dev-number/32) and NVE peering is established between these addresses. BGW-1 System MAC address is 5000.0002.0007, BGW-2 System MAC address is 5000.0003.0007 and BGW-3 System MAC address is 5000.0004.0007. The complete device configuration can be found from Appendix A at the end of this book. The left-hand site uses BGP AS 65012 and Site-Id 12. The right-hand site uses BGP AS65034 and Site-Id 34. For the sake of simplicity, the device count is minimized on this example network. Host Abba in VLAN 10 (IP:172.16.10.101/MAC: 1000.0010.abba) is connected to Leaf-101 and host beef (IP:172.16.10.102/MAC: 1000.0010.beef) is connected to Leaf-102. VLAN 10 is mapped to VNI 10000. In addition to unique switch Physical IP (PIP), intra-site BGW switches BGW-1 and BGW-2 share the same Virtual IP (VIP) that is taken from Loopback Interface 88 (192.168.88.12 in both devices).

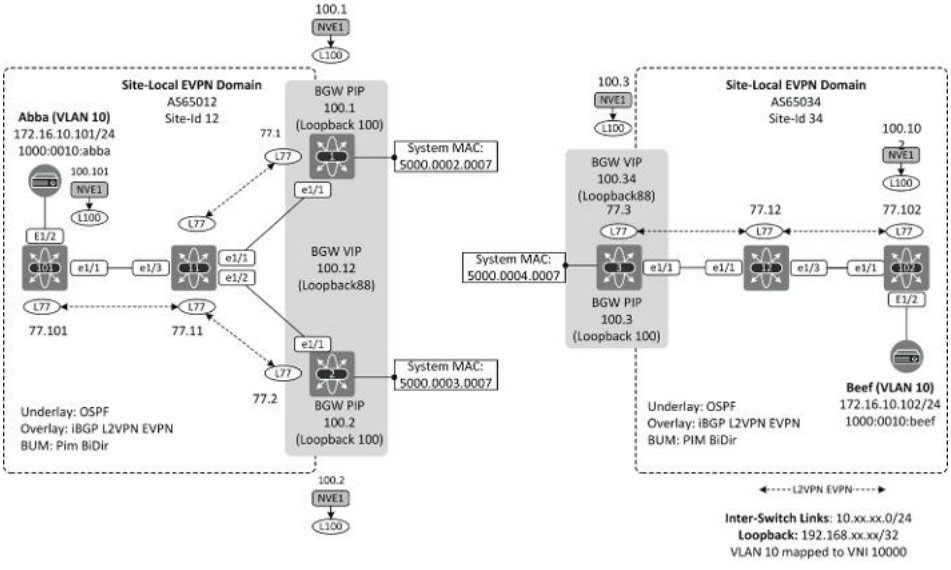


Figure 18-3: *Example EVPN Multi-Site topology.*

Intra-Site NVE peering and VXLAN tunnels

This section explains the intra-site architecture. Example 18-1 shows that Leaf-101 has three NVE peers, one with the BGW shared Virtual-IP (VIP) and two with BGW switches Physical-IP (PIP). BGW switches advertise VIP address as a next-hop address concerning all Route-Type 2 and 5 updates received from the remote BGW. A physical IP address is used for three purposes. *First*, In case the BGW switch has directly connected hosts (only routing model is supported), the host prefix is advertised with PIP as a next-hop. *Second*, If BGW switch is connected to an external network, the networks received from the external site are advertise with PIP as a next-hop. *Third*, For the BUM traffic replication, BGW switches use PIP. This means that Ingress-Replication (IR) tunnels end-point address advertised within BGP L2VPN EVPN Route-Type 3 (Inclusive Multicast Route) is PIP.

```
Leaf-101# sh nve peers detail
Details of nve Peers:
---------------------------------------
Peer-Ip: 192.168.88.12
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 01:29:16
    Router-Mac          : n/a
    Peer First VNI      : 10000
    Time since Create   : 01:29:16
    Configured VNIs     : 10000,10077
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10000
    vni assignment mode : SYMMETRIC
    Peer Location       : N/A
Peer-Ip: 192.168.100.1
    NVE Interface       : nve1
    Peer State          : Up
```

```
    Peer Uptime          : 01:07:26
    Router-Mac           : n/a
    Peer First VNI       : 10000
    Time since Create    : 01:07:26
    Configured VNIs      : 10000,10077
    Provision State      : peer-add-complete
    Learnt CP VNIs       : 10000
    vni assignment mode  : SYMMETRIC
    Peer Location        : N/A
Peer-Ip: 192.168.100.2
    NVE Interface        : nve1
    Peer State           : Up
    Peer Uptime          : 01:50:10
    Router-Mac           : n/a
    Peer First VNI       : 10000
    Time since Create    : 01:50:11
    Configured VNIs      : 10000,10077
    Provision State      : peer-add-complete
    Learnt CP VNIs       : 10000
    vni assignment mode  : SYMMETRIC
    Peer Location        : N/A
```

Example 18-1: *show nve peers detail.*

BGW switches generate the BGP L2VPN EVPN Route-Type 2 (MAC Advertisement Route) advertisements about their system-MAC address with the next-hop address of NVE Interface (PIP). Example 18-2 shows that Leaf-101 has received updates from Spine-11 concerning all three BGW switches used in this example. Note that the next-hop address towards intra-site BGW switches System MAC is a Physical IP (PIP) of BGW switch while the next-hop address towards system-MAC of inter-site BGW-3 switch is shared Virtual IP address (VIP) used between Intra-Site BGW switches BGW-1 and BGW-2. Note that both VIP and PIP have to be advertised by the Underlay Network routing protocol. Even though the route origin is not visible in the example, the BGP RID of advertising BGW can be seen from the Route-Distinguisher.

```
Leaf-101# sh bgp l2vpn evpn
<snipped>
   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.1:32777
*>i[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                     100          0 i

Route Distinguisher: 192.168.77.2:32777
*>i[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                     100          0 i

Route Distinguisher: 192.168.77.3:32777
*>i[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.88.12                     100          0 65088
65034 i
Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
*>i[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                     100          0 i
*>i[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                     100          0 i
*>i[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.88.12                     100          0 65088
65034 i
```

Example 18-2: *show bgp l2vpn evpn.*

The system-MAC attached to the NVE interface can be verified by using *show nve interface* command.

```
BGW-1# sh nve interface
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 5000.0002.0007
 Host Learning Mode: Control-Plane
 Source-Interface: loopback100 (primary: 192.168.100.1, secondary: 0.0.0.0)
```

Example 18-3: *show nve peers detail*

Example 18-4 shows the BGP table entry on Leaf-101 concerning the system-MAC address of BGW-1. The route is imported into BGP table based on Route-Target 65012:10000. The encapsulation type is VXLAN (type 8) and the advertised next-hop address is 192.168.100.1 (PIP). Based on both encapsulation type VXLAN and Next-hop IP address Leaf-101 knows that switch with IP address 192.168.100.1 has to be VXLAN tunnel end-point. Note that system-MAC address is advertised as a *sticky-MAC* address (shown in partial Capture 18-1) with *MAC-Mobility* Extended Community where the *static-flag* is set one (1) and the Sequence number is set to zero. The captured packet is shown in Capture 18-1 after example18-5.

```
Leaf-101# sh bgp l2vpn evpn 5000.0002.0007
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.1:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216, version 19
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.1 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65012:10000 SOO:192.168.77.1:512 ENCAP:8
          MAC Mobility Sequence:01:0
      Originator: 192.168.77.1 Cluster list: 192.168.77.11

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.101:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216, version 20
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported from
192.168.77.1:32777:[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
  AS-Path: NONE, path sourced internal to AS
    192.168.100.1 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
```

```
      Extcommunity: RT:65012:10000 SOO:192.168.77.1:512 ENCAP:8
         MAC Mobility Sequence:01:0
      Originator: 192.168.77.1 Cluster list: 192.168.77.11

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

Example 18-4: *show nve peers detail*

Before bringing up the tunnel, Leaf-101 has to verify that the IP address 192.168.100.1 is reachable through the Underlay Network.

```
Leaf-101# sh ip route 192.168.100.1
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.100.1/32, ubest/mbest: 1/0
    *via 10.101.11.11, Eth1/1, [110/81], 02:58:17, ospf-UNDERLAY-NET, intra
```

Example 18-5: *show ip route 192.168.100.1*

```
Internet Protocol Version 4, Src: 192.168.77.11, Dst: 192.168.77.101
Transmission Control Protocol, Src Port: 57069, Dst Port: 179, Seq: 110, Ack:
39, Len: 242
Border Gateway Protocol - UPDATE Message
<snipped>
        Path Attribute - EXTENDED_COMMUNITIES
<snipped>
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 32
            Carried extended communities: (4 communities)
                Route Target: 65012:10000 [Transitive 2-Octet AS-Specific]
                Route Origin: 192.168.77.1:512
                    Type: Transitive IPv4-Address-Specific (0x01)
                    Subtype (IPv4): Route Origin (0x03)
                    IPv4 address: 192.168.77.1
                    2-Octet AN: 512
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
                    Type: Transitive Opaque (0x03)
                    Subtype (Opaque): Encapsulation (0x0c)
                    Tunnel type: VXLAN Encapsulation (8)
                MAC Mobility: Sticky MAC [Transitive EVPN]
                    Type: Transitive EVPN (0x06)
                    Subtype (EVPN): MAC Mobility (0x00)
                    Flags: 0x01
                        .... ...1 = Sticky/Static MAC: Yes
                    Sequence number: 0
        Path Attribute - ORIGINATOR_ID: 192.168.77.1
        Path Attribute - CLUSTER_LIST: 192.168.77.11
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive,
            Type Code: MP_REACH_NLRI (14)
            Length: 44
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (35 bytes)
                EVPN NLRI: MAC Advertisement Route
                    Route Type: MAC Advertisement Route (2)
                    Length: 33
```

```
                    Route Distinguisher: 192.168.77.1:32777
                    ESI: 00 00 00 00 00 00 00 00 00
                    Ethernet Tag ID: 0
                    MAC Address Length: 48
                    MAC Address: 50:00:00:02:00:07 (50:00:00:02:00:07)
                    IP Address Length: 0
                    IP Address: NOT INCLUDED
                    MPLS Label Stack 1: 625, (BOGUS: Bottom of Stack NOT set!)
```

Capture 18-1: *show nve peers detail*

The figure 18-4 summarizes the NVE peering from the Leaf-101 perspective. BGW-1 sends the BGP L2VPN EVPN Update including its system-MAC address. This way Intra-Site Leaf-101 learns the information which is needed for NVE peering. Shared NVE Anycast-BGW address 192.168.88.12 is learned from the BGP L2VPN EVPN Mac Route Advertisement originated by BGW-3 and forwarded to both intra-site BGW switches. When DC Core switch (Route-Server) receives the Update message it changes the Route-Target (RT) AS part to its' own AS due to the rt-rewrite definition. When BGW-1 receives this update, it also modifies the RT Extended Community to its own AS and it imports the NLRI information. When sending an Update to Leaf-101, BGW-1 sets the Next-Hop to 192.168.88.12, which is the shared Anycast BGW address. Based on the RT 65012:10000 Leaf-101 is able to import BGP L2VPN EVPN MAC Advertisement Route originated by BGW-3 and learn the IP address of Intra-Site Anycast BGW from the Next-Hop field. This learning process is Control Plane learning and is also used for establishing NVE peering between Intra-Site BGW switches.

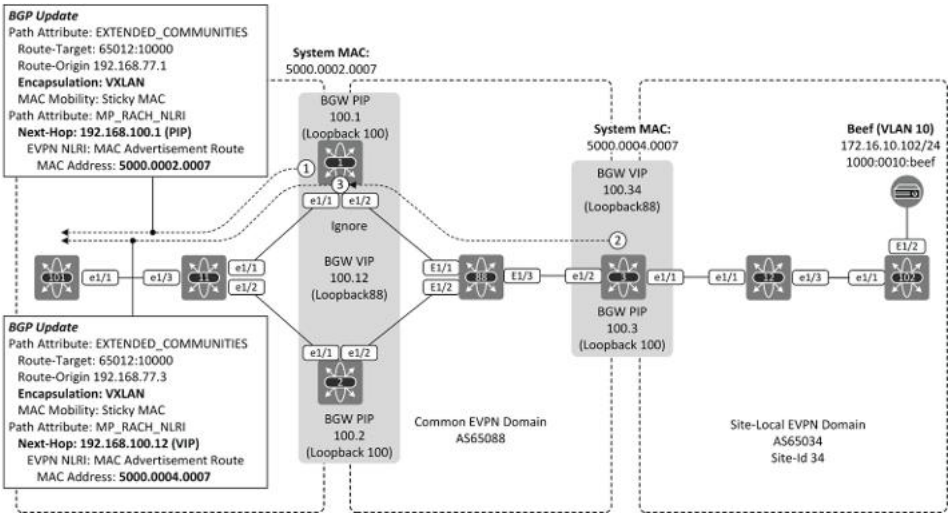


Figure 18-4: *NVE peer learning process.*

Example 18-6 shows that even though Leaf-101 has established NVE peering to BGW-1 the tunnel is still Unidirectional. BGW1 does only have NVE peering with fabric internal BGW2 and the BGW-3 but not with Leaf-101. This is because only BGWs advertises their System MAC-addresses as Route-Type 2 MAC advertisement route. The leaf is a normal VTEP switch so it does not advertise its system MAC.

```
BGW-1# sh nve peer control-plane
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- ---------------  ----- --------- -------- -----------------
nve1      192.168.100.2    Up    CP        03:05:49 n/a
nve1      192.168.100.3    Up    CP        02:44:54 n/a
```

Example 18-6: *show nve peers detail*

Now host Abba joins to the network. It pings the Anycast GW used in VLAN 10. This way Leaf-101 learns the MAC address of host Abba and sends BGP L2VPN EVPN Route-type 2 advertisement to Route-Reflector Spine-11, which in turn forward the message to BGW switches. Example 18-8 shows that after receiving the BGP Update related to host Abba, BGW1 also has established the NVE peering with Leaf-101. Now there is a bi-directional VXLAN tunnel between these two switches and data can flow over it.

```
BGW-1# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 29, Local Router ID is 192.168.77.1
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 -
best2

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.1:27001   (ES [0300.0000.0000.0c00.0309 0])
*>l[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
                      192.168.100.1                         100      32768 i
*>i[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                         100          0 i

Route Distinguisher: 192.168.77.1:32777    (L2VNI 10000)
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.100.101                       100          0 i
*>l[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                         100      32768 i
*>i[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                         100          0 i
*>e[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.100.3                                      0 65088
65034 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.101]/272
                      192.168.100.101                       100          0 i
*>l[3]:[0]:[32]:[192.168.100.1]/88
                      192.168.100.1                         100      32768 i
*>e[3]:[0]:[32]:[192.168.100.3]/88
                      192.168.100.3                                      0 65088
65034 i

Route Distinguisher: 192.168.77.2:27001
*>i[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                         100          0 i

Route Distinguisher: 192.168.77.2:32777
*>i[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                         100          0 i

Route Distinguisher: 192.168.77.3:32777
```

```
*>e[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                     192.168.100.3                                  0 65088
65034 i
*>e[3]:[0]:[32]:[192.168.100.3]/88
                     192.168.100.3                                  0 65088
65034 i

Route Distinguisher: 192.168.77.101:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.100.101                100             0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.101]/272
                     192.168.100.101                100             0 i
```

Example 18-7: *sh bgp l2vpn evpn*

```
BGW-1# sh nve peers control-plane
Interface Peer-IP          State LearnType Uptime   Router-Mac
--------- ---------------  ----- --------- -------- -----------------
nve1      192.168.100.2    Up    CP        03:14:42 n/a
nve1      192.168.100.3    Up    CP        02:53:47 n/a
nve1      192.168.100.101  Up    CP        00:01:07 n/a
```

Example 18-8: *show nve peers control-plane.*

Summary

As a conclusion, intra-site NVE peering is based on information carried within auto-generated Route-Type 2 describing system MAC address by BGW. The NVE peering from BGW-to-Leaf is based on information carried within the first Route-Type 2 MAC advertisement route that describes one of the hosts behind the Leaf switch. The result for this is the bi-directional VXLAN tunnel.

Shared Common EVPN Domain Connections

Figure 18-5 illustrates the overall topology and eBGP peering between Border Gateways and DC Core Switch. For simplicity, only one DC Core switch is used. The DC Core switch has its dedicated BGP AS65088 meaning external BGP peering is used. DC Core switch also has the role of Route Server (RS) role. In this example, the RS is in the data path but in real-life scenarios, it does not have to be. The complete configuration of RS can be found from Appendix A at the end of this book. DC Core switch and all three BGW switches belong to the Common EVPN Domain used for datacenter Interconnect (DCI). This means that each BGW belongs to both Intra-Site EVPN Domain as well as to Common EVPN Domain. All Unicast and Multicast traffic from one site to another goes through the BGW.

The eBGP IPv4 unicast afi is used for IPv4 NLRI exchange in an Underlay Network. BGW switches advertise their unique NVE interface IP address (PIP) and shared Virtual IP address (VIP) as well as the IP address of the external interface connected to DC Core switch, which in turn forwards these BGP Updates to another site. PIP addresses are used in the outer IP header destination and source IP address when BUM traffic is sent over the Ingress-Replication tunnel. VIP address, in turn, is used in the outer IP header for Unicast traffic. Physical Interface IP addresses (Underlay Network) are used for recursive route lookup to find the next hop for the PIP/VIP address.

The eBGP L2VPN EVPN afi is used for exchanging EVPN NLRI in an Overlay Network. The information includes NLRI of intra-site host MAC and MAC/IP information (Route-Type 2) and IP Prefix information (Route-Type 5). Note that BGW switches also exchanges their System-MAC addresses information by using Route-Type 2 MAC Advertisement Routes for NVE peering. In addition, BGW switches advertise *Inclusive Multicast Route* (Route-Type 3) to exchange NLRI information concerning the Ingress-Replication tunnel. BGW switches advertise also *Ethernet Segment Routes* (Route-Type 4) used for Intra-Site BGW DF election over Common EVPN Domain but those are ignored by the remote BGW switch due to unmatched route-target import policy.

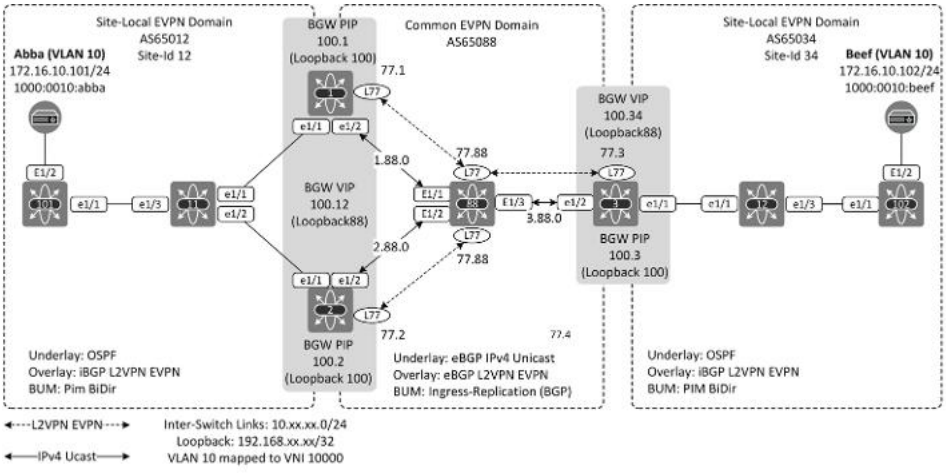


Figure 18-5: *Common EVPN Domain Underlay and Overlay eBGP peering.*

Border Gateway setup

This section explains the Border Gateway configuration.

Define Site-Id

Configure the device role as an EVPN Multi-Site Border Gateway an assign Site-Id to it. The site-Id has to be identical in all BGWs belonging to the local site. Optionally, the shared Virtual-IP address advertisement can be delayed after recovery. This way Underlay and Overlay Network Control Plane protocols of BGW switch have sufficient time to do their job such as building a BGP peering and establish both VXLAN and Ingress-Replication tunnels before introducing itself as a possible next-hop for the inter-site destination by advertising Virtual IP address.

```
evpn multisite border-gateway 12
  delay-restore time 300
```

Example 18-9: *enabling EVPN MS BGW on BGW1.*

Define source IP for VIP under NVE Interface and BUM method for DCI

NVE interface of BGW-1 uses IP address 192.168.00.1 (Loopback 100) as a Physical IP (PIP) and the IP address 192.168.88.12 (Loopback 88) as a Virtual IP address (VIP). Ingress Replication is used for Inter-Site BUM traffic for VNI 10000 while Intra-Site BUM traffic uses Multicast.

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  multisite border-gateway interface loopback88
  member vni 10000
    multisite ingress-replication
    mcast-group 238.0.0.10
  member vni 10077 associate-vrf
```

Example 18-10: *configuring NVE interface on BGW1.*

Configure BGP Peering and redistribution

Configure eBGP IPv4 Unicast afi peering for Underlay Network between physical link interface IP addresses. Advertise BGW switch Loopback interface IP addresses used by NVE interface (VIP/PIP) and BGP peering to DC Core switch. Configure eBGP L2VPN EVPN afi peering for Overlay Network between Loopback 77 IP addresses and define the fabric peer-type as external. When eBGP peering is configured between Loopback interfaces there is also a need for adjusting TTL value by using the "ebgp-multihop <value>" command. In this example scenario, the value is five.

BGP L2VPN EVPN Updates send by BGW1 will carry site site-specific Route-Target Extended Community per VNI. This community use format BGP-AS: VNI-Id. This is why there is a command "*rewrite-evpn-rt-asn*" under the L2VPN EVPN address-family. It modifies the Route-Target BGP AS-part from received number to local AS number. When BGW1 sends an eBGP L2VPN Update to DC Core switch, the original RT for VNI 10000 is 65012:10000. When DC Core switch receives the update message, it changes the RT to 65088:10000. It uses this RT value when sending the update message to BGW-3 on the other site. When BGW-3 receives the update message, it changes the RT to 65034:10000 before installing it into Adj-RIB-In. This way it is able to import NLRI information carried in update originated by remote-site BGW. Adjust also the BGP maximum path for load-balancing.

```
router bgp 65012
  router-id 192.168.77.1
  no enforce-first-as
  address-family ipv4 unicast
    redistribute direct route-map REDIST-TO-SITE-EXT-DCI
  address-family l2vpn evpn
    maximum-paths 2
    maximum-paths ibgp 2
  neighbor 10.1.88.88
    remote-as 65088
    update-source Ethernet1/2
    address-family ipv4 unicast
  neighbor 192.168.77.11
    remote-as 65012
    description ** Spine-11 BGP-RR **
    update-source loopback77
```

```
    address-family l2vpn evpn
      send-community extended
  neighbor 192.168.77.88
    remote-as 65088
    update-source loopback77
    ebgp-multihop 5
    peer-type fabric-external
    address-family l2vpn evpn
      send-community
      send-community extended
      rewrite-evpn-rt-asn
!
route-map REDIST-TO-SITE-EXT-DCI permit 10
  match tag 1234
!
interface loopback88
  description ** VIP for DCI-Inter-connect **
  ip address 192.168.88.12/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
```

Example 18-11: *configuring eBGP IPv4 Unicast and L2VPN EVPN peering route redistribution on BGW1.*

Configure DCI and Fabric Interface Tracking

BGW is in the borderline of intra-site EVPN Domain (Fabric EVPN) and Common EVPN Domain (DCI). All inter-site traffic goes through the BGW switches, so it is extremely important to have a mechanism for tracking the state of both Fabric and DCI interfaces. The configuration is shown in the example below. Link failure events are discussed in detail in "Failure Scenario" section.

```
interface Ethernet1/1
  description **Fabric Internal **
  no switchport
  mtu 9216
  mac-address b063.0001.1e11
  medium p2p
  ip address 10.1.11.1/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  evpn multisite fabric-tracking
  no shutdown
!
interface Ethernet1/2
  description ** DCI Interface **
  no switchport
  mtu 9216
  mac-address b063.0001.1e12
  medium p2p
  ip address 10.1.88.1/24 tag 1234
  ip pim sparse-mode
  evpn multisite dci-tracking
  no shutdown
```

Example 18-12: *DCI and fabric interface tracking on BGW1.*

These are the basic EVPN Multi-Site related configuration. Complete configuration of all BGP switches and DC Core switch can be found from Appendix A at the end of this chapter.

BGP peering Verification on BGW

Example 18-13 shows that BGW-1 has established an iBGP L2VPN EVPN session with 192.168.77.11 (Spine-11) and it has received three Route-Type 2 (MAC Advertisement Route) and one Route-Type 4 (Ethernet Segment Route). It also has established an eBGP L2VPN EVPN session with 192.168.77.88 (DC Core switch) from where it has received one Route-Type 2 (MAC Advertisement Route) and one Route-Type 3 (Inclusive Multicast Ethernet Tag Route).

```
BGW-1# sh bgp l2vpn evpn summary
BGP summary information for VRF default, address family L2VPN EVPN
BGP router identifier 192.168.77.1, local AS number 65012
BGP table version is 251, L2VPN EVPN config peers 2, capable peers 2
15 network entries and 15 paths using 2616 bytes of memory
BGP attribute entries [11/1804], BGP AS path entries [1/10]
BGP community entries [0/0], BGP clusterlist entries [2/8]

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.77.11   4 65012     609     486      251    0    0 07:14:11 4
192.168.77.88   4 65088     487     531      251    0    0 07:12:39 2

Neighbor        T    AS PfxRcd     Type-2     Type-3     Type-4     Type-5
192.168.77.11   I 65012 4          3          0          1          0
192.168.77.88   E 65088 2          1          1          0          0
```

Example 18-13: *sh bgp l2vpn evpn summary on BGW1.*

NVE peering Verification on BGW1

Example 18-14 shows that BGW-1 has established an NVE peering between the intra-site BGW-2 and Leaf-101. The Peer-Location shown in output verifies that these are fabric peers. In addition, BGW-1 has established NVE peering with BGW-3 which location is described as DCI. The NVE peering process between the inter-site BGW switches uses the same mechanism than NVE peering between intra-site BGW switches. The trigger for the NVE peer learning process is auto-generated system MAC-address advertisement (Route-Type 2 – MAC Advertisement Route).

```
BGW-1# sh nve peers detail
Details of nve Peers:
----------------------------------------
Peer-Ip: 192.168.100.2
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 08:13:09
    Router-Mac          : n/a
    Peer First VNI      : 10000
    Time since Create   : 08:13:09
    Configured VNIs     : 10000,10077
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10000
    vni assignment mode : SYMMETRIC
    Peer Location       : FABRIC
Peer-Ip: 192.168.100.3
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 07:52:14
    Router-Mac          : n/a
```

```
    Peer First VNI      : 10000
    Time since Create   : 07:52:14
    Configured VNIs     : 10000,10077
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10000
    vni assignment mode : SYMMETRIC
    Peer Location       : DCI
Peer-Ip: 192.168.100.101
    NVE Interface       : nve1
    Peer State          : Up
    Peer Uptime         : 04:59:34
    Router-Mac          : n/a
    Peer First VNI      : 10000
    Time since Create   : 04:59:34
    Configured VNIs     : 10000,10077
    Provision State     : peer-add-complete
    Learnt CP VNIs      : 10000
    vni assignment mode : SYMMETRIC
    Peer Location       : FABRIC
```

Example 18-14: *sh bgp l2vpn evpn summary on BGW1.*

Example 18-15 taken from BGW-1 shows the BGW NVE related information. The output shows among the other things the NVE source interface that is a Physical IP address (PIP) and the shared Virtual IP address (VIP) used as a next-hop for ingress and egress inter-site traffic. Note that the operational state for the VIP interface (Loopback88) is "down". This is because the output is taken when there was neither IP connectivity nor NVE peering between BGW-1 and BGW-2 (Spine-11 was turned off).

```
BGW-1# sh nve interface nve1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 5000.0002.0007
 Host Learning Mode: Control-Plane
 Source-Interface: loopback100 (primary: 192.168.100.1, secondary: 0.0.0.0)
 Source Interface State: Up
 Virtual RMAC Advertisement: No
 NVE Flags:
 Interface Handle: 0x49000001
 Source Interface hold-down-time: 180
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: N/A
 Virtual Router MAC Re-origination: 0200.c0a8.580c
 Interface state: nve-intf-add-complete
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 22 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Down)
 Multisite bgw-if oper down reason:
```

Example 18-15: *sh nve interface nve1 detail on BGW1.*

When Spine-11 boots up and the IP connectivity and NVE peering is established between BGW-1 and BGW-2 the operational state for Loopback88 interface on BGW-1 changes to UP-state.

```
BGW-1# sh nve interface nve1 detail | i bgw-if
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Up)
 Multisite bgw-if oper down reason:
```

Example 18-16: *sh nve interface nve1 detail on BGW1.*

BGP NLRI information verification.

Host Abba connected to Leaf-101 joins the network. It pings the Anycast-GW IP address 172.16.10.1 (SVI for VLAN 10). Leaf-101 learns the MAC address information from the ingress frame. It stores the MAC information into the MAC address table and L2RIB of MAC-VRF from where the MAC address information is exported into BGP Loc-RIB and sends it through the Adj-RIB-Out to Spine-11. BGP Route-Reflector Spine-11 forwards the BGP Update to both BGW-1 and BGW-2. BGW switches forwards BGP Update to DC Core switch after local processing. The example below shows that DC Core switch has learned the MAC address 1000.0010.abba of the host from both BGW-1 and BGW-2 with the same Next-Hop address 192.168.88.12 (VIP).

```
Route Distinguisher: 192.168.77.101:32777
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.88.12         2000               0 65012 i
* e                   192.168.88.12         2000               0 65012 i
  e[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.101]/248
                     192.168.88.12         2000               0 65012 i
  e                  192.168.88.12         2000               0 65012 i
```

Example 18-17: *sh bgp l2vpn evpn summary on DC Core switch (RouteServer).*

The example below shows that the DC Core switch has installed a route to 192.168.88.12 into RIB from BGP Loc-RIB with two equal next-hop IP addresses (Underlay Network addresses) and will use both of these for ECMP load-balancing toward the destination.

```
RouteServer-1# sh ip route 192.168.88.12
<snipped>
192.168.88.12/32, ubest/mbest: 2/0
    *via 10.1.88.1, [20/0], 00:00:04, bgp-65088, external, tag 65012
    *via 10.2.88.2, [20/0], 00:00:04, bgp-65088, external, tag 65012
```

Example 18-18: *sh ip route 192.168.88.12 on DC Core switch.*

The example below shows that BGW-3 has received the BGP Update about the MAC addresses information of host Abba from DC Core switch. BGW-3 has changed the Route-Target AS-part to its BGP AS before importing the route from Adj-RIB-In (pre) into the Adj-RIB-In (post). From the Adj-RIB-In (post) route is imported into the Loc-RIB.

```
BGW-3# show bgp l2vpn evpn 1000.0010.abba
BGP routing table information for VRF default, address family L2VPN EVPN
!----------------< COMMENT: This entry is in BGP Loc-RIB >-------------
Route Distinguisher: 192.168.77.3:32777    (L2VNI 10000)
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216, version 289
Paths: (1 available, best #1)
Flags: (0x000212) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
```

```
  Path type: external, path is valid, is best path, no labeled nexthop, in rib
             Imported from
192.168.77.101:32777:[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
  AS-Path: 65088 65012 , path sourced external to AS
    192.168.88.12 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65034:10000 ENCAP:8

  Path-id 1 not advertised to any peer
!----------------< COMMENT: This entry is in BGP Adj-RIB-In >-------------
Route Distinguisher: 192.168.77.101:32777
BGP routing table entry for
[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216, version 288
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: 65088 65012 , path sourced external to AS
    192.168.88.12 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10000
      Extcommunity: RT:65034:10000 ENCAP:8

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

Example 18-19: *sh bgp l2vpn evpn summary on BGW1.*

L2RIB Verification on remote BGW

BGW-3 has installed MAC address information from the BGP Loc-RIB into L2RIB.

```
BGW-3# show l2route mac all

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

Topology    Mac Address    Prod   Flags          Seq No     Next-Hops
----------- -------------- ------ -------------- ---------- ----------------
10          1000.0010.abba BGP    Rcv            0          192.168.88.12
10          5000.0004.0007 VXLAN  Stt,Nho,       0          192.168.100.3
```

Example 18-20: *show l2route mac all on BGW3.*

MAC Address-Table Verification on remote BGW switch

BGW-3 has also installed MAC information into the MAC address-table. The information stored in both L2RIB and MAC Address-Table incudes almost identical information. The difference compared to these two tables relies on usage. The Data Plane uses MAC address-Table for switching while the Control Plane uses the L2RIB for exporting/importing information to and from BGP processes.

```
BGW-3# show system internal l2fwder mac
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link,
        (T) - True, (F) - False, C - ControlPlane MAC
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------
G     -    b063:0003:1e12    static   -          F     F   sup-eth1(R)
G     -    b063:0003:1e11    static   -          F     F   sup-eth1(R)
*    10    1000.0010.abba    static   -          F     F  nve-peer2
192.168.88.12
G     -    b063:0003:1e14    static   -          F     F   sup-eth1(R)
G     -    b063:0003:1e13    static   -          F     F   sup-eth1(R)
G     -    0200:c0a8:5822    static   -          F     F   sup-eth1(R)
    1           1         -00:01:00:01:00:01         -             1
```

Example 18-21: *show system internal l2fwder mac on BGW3.*

BGP NLRI Next-Hop verification on remote Leafs

When BGW-3 forwards a BGP Updates message received from the Common EVPN Domain into intra-site devices, it changes the next-hop IP address to VIP address (even though it is the only BGW on-site).

```
Leaf-102# sh bgp l2vpn evpn
Route Distinguisher: 192.168.77.101:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.88.34                     100          0 65088
65012 i

Route Distinguisher: 192.168.77.102:32777    (L2VNI 10000)
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                     192.168.88.34                     100          0 65088
65012 i
Leaf-102#
```

Example 18-22: *show bgp l2vpn evpn on Leaf-102.*

The same process is done by BGW-1 and BGW-2. They both changes the next-hop address to a shared VIP address when sending BGP L2VPN EVPN BGP Updates received from Common EVPN Domain to intra-site devices.

```
Leaf-101# sh bgp l2vpn evpn
<snipped>
Route Distinguisher: 192.168.77.102:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.beef]:[0]:[0.0.0.0]/216
                     192.168.88.12                     100          0 65088
65034 i

Leaf-101#
```

Example 18-23: *show bgp l2vpn evpn on Leaf-101.*

A simple ping test verifies that there is a connection between host Beef connected to Leaf-102 and host Abba connected to Leaf-101.

```
Beef#ping 172.16.10.101
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.101, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 96/137/224 ms
```

Example 18-24: *ping from host Beef to host Abba within VLAN 10.*

Multi-Destination traffic forwarding

There is two important consideration related to inter-site BUM traffic. *First*, if there is more than one intra-site BGW switches, the role of Designated Forwarder (DF) per VLAN/VNI is selected randomly among all Intra-Site BGW switches. DF is the switch that is responsible for inter-site ingress/egress BUM traffic forwarding. *Second*, when the DF election is done, BGW switches need to know whom to forward inter-site BUM traffic over Common EVPN Domain. This means that BGWs with each location needs to build a Multi-Destination Tree between themselves. The next two sections explains the DF election process by using BGP L2VPN EVPN Route-Type 4 (Ethernet Segment Route) and Multi-Destination Tree building process by using BGP L2VPN EVPN Route-Type 3 (Inclusive Multicast Route) for finding Ingress-Replication peers.

Designated Forwarder

BGW switches send a BGP L2VPN EVPN Route-Type 4 (Ethernet Segment Route) update to all of their BGP L2VPN EVPN peers. Switches use this information for selecting DF per VLAN/VNI. The first part of the NLRI update message [4] describes the EVPN Route-Type. The second part [0300.0000.0000.0c00.0309] includes information about; ESI Type (03 = MAC-based ESI), ESI system MAC 0000.0000.000c (formed from Site-Id 12 = HEX 0c). It also contains the auto-generated ESI local discriminator 000309. Value [32] describes the length of the following IP address that describes the sender IP address [192.168.100.1] which is used for DF election process. In addition, the Update message carries an *ES Import Route-Target* Extended Community BGP Path Attribute that is generated automatically based on the local Site-Id (0c = 12). Only the Intra-Site BGW switches later import these Updates.

All BGP L2VPN EVPN peers will receive the Route-Type 4 BGP Update, also Leaf-101 (reflected by Spine-11) and BGW-3 will receive the BGP Update though they ignore it because they do not have matching import clause for the RT.

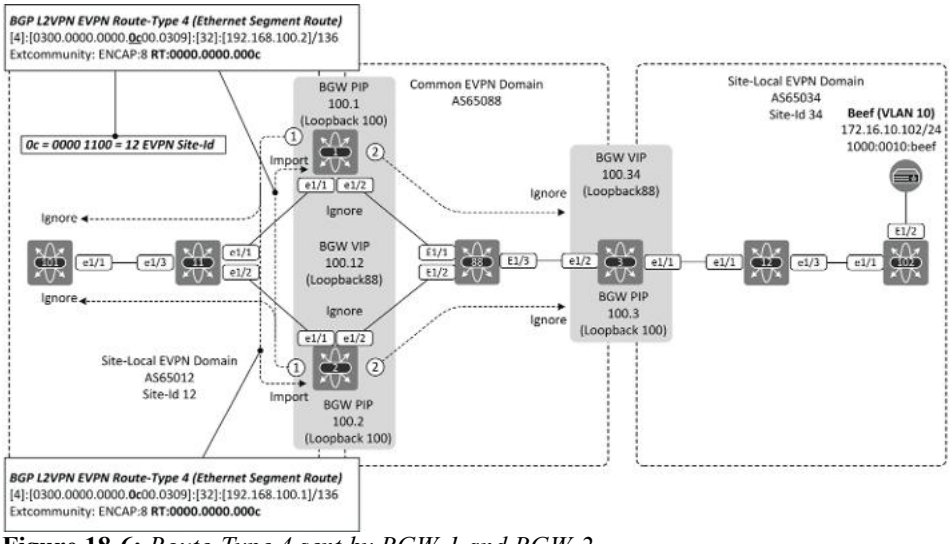


Figure 18-6: *Route-Type 4 sent by BGW-1 and BGW-2.*

Examples 18-25 shows that BGW-1 has installed Ethernet Segment Route learned from the peer site-local BGW-2 switch into the BGP table. The Route-Target Extended Path Attribute is based on Site-Id meaning that only intra-site BGW switches are able to import Ethernet Segment Routes between each other.

```
BGW-1# sh bgp l2vpn evpn route-type 4
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.1:27001    (ES [0300.0000.0000.0c00.0309 0])
BGP routing table entry for
[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136, version 7
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    192.168.100.1 (metric 0) from 0.0.0.0 (192.168.77.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: ENCAP:8 RT:0000.0000.000c

  Path-id 1 advertised to peers:
    192.168.77.11      192.168.77.88
BGP routing table entry for
[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136, version 9
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
```

```
            Imported from
192.168.77.2:27001:[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
  AS-Path: NONE, path sourced internal to AS
    192.168.100.2 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: ENCAP:8 RT:0000.0000.000c
      Originator: 192.168.77.2 Cluster list: 192.168.77.11

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.2:27001
BGP routing table entry for
[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136, version 8
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
            Imported to 1 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.2 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: ENCAP:8 RT:0000.0000.000c
      Originator: 192.168.77.2 Cluster list: 192.168.77.11

  Path-id 1 advertised to peers:
    192.168.77.88
```

Example 18-25: *BGP L2VPN EVPN Ethernet Segment Route in BGW-1 BGP table.*

Examples 18-26 shows that BGW-2 has installed Ethernet Segment Route learned from the site-local peer BGW-1 switch into BGP table.

```
BGW-2# sh bgp l2vpn evpn route-type 4
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.1:27001
BGP routing table entry for
[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136, version 8
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
            Imported to 1 destination(s)
  AS-Path: NONE, path sourced internal to AS
    192.168.100.1 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: ENCAP:8 RT:0000.0000.000c
      Originator: 192.168.77.1 Cluster list: 192.168.77.11

  Path-id 1 advertised to peers:
    192.168.77.88

Route Distinguisher: 192.168.77.2:27001   (ES [0300.0000.0000.0c00.0309 0])
BGP routing table entry for
[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136, version 9
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
```

```
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported from
192.168.77.1:27001:[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
  AS-Path: NONE, path sourced internal to AS
    192.168.100.1 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: ENCAP:8 RT:0000.0000.000c
      Originator: 192.168.77.1 Cluster list: 192.168.77.11

  Path-id 1 not advertised to any peer
BGP routing table entry for
[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136, version 7
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    192.168.100.2 (metric 0) from 0.0.0.0 (192.168.77.2)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: ENCAP:8 RT:0000.0000.000c

  Path-id 1 advertised to peers:
    192.168.77.11      192.168.77.88
```

Example 18-26: *BGP L2VPN EVPN Ethernet Segment Route in BGW-2 BGP table.*

The capture below shows the BGP L2VPN EVPN Route-Type 4 (Ethernet Segment Route) sent by BGW-1.

```
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 93
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 70
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: 65012
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 16
            Carried extended communities: (2 communities)
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
                    Type: Transitive Opaque (0x03)
                    Subtype (Opaque): Encapsulation (0x0c)
                    Tunnel type: VXLAN Encapsulation (8)
                ES Import: RT: 00:00:00:00:00:0c [Transitive EVPN]
                    Type: Transitive EVPN (0x06)
                    Subtype (EVPN): ES Import (0x02)
                    ES-Import Route Target: 00:00:00_00:00:0c
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive
            Type Code: MP_REACH_NLRI (14)
            Length: 34
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
```

```
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (25 bytes)
                EVPN NLRI: Ethernet Segment Route
                    Route Type: Ethernet Segment Route (4)
                    Length: 23
                    Route Distinguisher: 192.168.77.1:27001
                    ESI: 00:00:00:00:00:0c, Discriminator: 00 03
                        ESI Type: ESI MAC address defined (3)
                        ESI system MAC: 00:00:00_00:00:0c
                        ESI system mac discriminator: 00 03
                        Remaining bytes: 09
                    IP Address Length: 32
                    IPv4 address: 192.168.100.1
```

Capture 18-2: *BGP L2VPN EVPN Route-Type 4*

BGW switches choose Designated Forwarder (DF) among themselves to forward BUM (Broadcast, Unknown Unicast and Multicast) traffic to and from intra-site EVPN Domain. If intra-site has more than one VLAN, the DF roles are load-balanced between BGW nodes, i.e. DF for VLAN 10 is BGW-1 and DF for VLAN 1 and 77 is BGW-2. The selection process uses the formula "*i = V mod N"*, where V represents VLAN Id and N represents a number of BGW switches in the redundancy group. The "i" is an ordinal of a leaf switch in the redundancy group. When BGW-1 and BGW-2 exchanges BGP L2VPN EVPN Route-Type 4 routes (Ethernet Segment Route) their IP address is included in NLRI. Each switch sets these IP addresses learned from BGP Update in numerical order from lowest to highest. In case of BGW-1 and BGW-2, the order is 192.168.100.1, 192.168.100.2. The lowest IP i.e. 192.168.100.1 gets ordinal zero (0) and the next one gets ordinal one (1) and so on.

Formula to calculate DF for VLAN 10 is

V mod N = i
V = 10 (VLAN Id)
N = 2 (number of leaf switches)
10 mod 2 = 0 > Leaf-102
(Remainders is zero (0) when 10 is divided by 2)
Ordinal zero is used by BGW-1, so it will be the DF for VLAN 10.

Formula to calculate DF for VLAN 77 is

V mod N = i
V = 77 (VLAN Id)
N = 2 (number of leaf switches)
77 mod 2 = 01 > BGW-1
(Remainders is one (1) when 77 is divided by 2)
Ordinal one is used by BGW-2, so it will be the DF for VLAN 77.

This procedure is the same that what was introduced in "EVPN ESI Multihoming- Part I: EVPN Ethernet Segment (ES) DF election section".

Examples 18-27 shows that BGW-1 is DF for VLAN 10 and example 18-28 shows that BGW-2 is DF for VLAN 1 and 77.

```
BGW-1# sh nve ethernet-segment

ESI: 0300.0000.0000.0c00.0309
   Parent interface: nve1
  ES State: Up
  Port-channel state: N/A
  NVE Interface: nve1
   NVE State: Up
   Host Learning Mode: control-plane
  Active Vlans: 1,10,77
   DF Vlans: 10
   Active VNIs: 10000
  CC failed for VLANs:
  VLAN CC timer: 0
  Number of ES members: 2
  My ordinal: 0
  DF timer start time: 00:00:00
  Config State: N/A
  DF List: 192.168.100.1 192.168.100.2
  ES route added to L2RIB: True
  EAD/ES routes added to L2RIB: False
  EAD/EVI route timer age: not running
```

Example 18-27: *DF election verification on BGW-1.*

```
BGW-2# sh nve ethernet-segment

ESI: 0300.0000.0000.0c00.0309
   Parent interface: nve1
  ES State: Up
  Port-channel state: N/A
  NVE Interface: nve1
   NVE State: Up
   Host Learning Mode: control-plane
  Active Vlans: 1,10,77
   DF Vlans: 1,77
   Active VNIs: 10000
  CC failed for VLANs:
  VLAN CC timer: 0
  Number of ES members: 2
  My ordinal: 1
  DF timer start time: 00:00:00
  Config State: N/A
  DF List: 192.168.100.1 192.168.100.2
  ES route added to L2RIB: True
  EAD/ES routes added to L2RIB: False
  EAD/EVI route timer age: not running
```

Example 18-28: *DF election verification on BGW-2.*

Ingress-Replication

In order to forward Inter-Site Multi-Destination traffic, BGW switches form a Multi-destination tree between remote-site BGW switches. Switches use BGP L2VPN EVPN Route-Type 3 (Inclusive Multicast Route) to describe their Tunnel-Id used with VNI and tunnel type, which is Ingress-Replication. By using this information, switches are able to form the Multi-Destination tree over the Unicast-Only Underlay Network.

In figure 18-6, BGW-1 sends a BGP L2VPN EVPN Update to BGW-3. EVPN NLRI describes the Route-Type (Inclusive Multicast Route) and sender IP (192.168.100.1). PMSI Tunnel Attribute describes the tunnel type (Ingress-Replication) the VNI which BUM traffic should be sent over the tunnel and Tunnel-Id used by BGW-1. This attribute is discussed in a later section. Route-Target Extended Community is set based on local values (65012:10000) which receiving switches changes to correspond their own AS: VNI. The same process applies to BGW-2 and BGW-3.

Note that DC Core SW does not forward BGP L2VPN EVPN Inclusive Multicast Route sent by BGW-1 to BGW-2 due to the same AS number. This is a normal BGP Loop Prevention mechanism. Also, BGP L2VPN EVPN Inclusive Multicast Route is only sent out from the DCI interface and receiving BGW switch does not forward it to local BGP speakers.

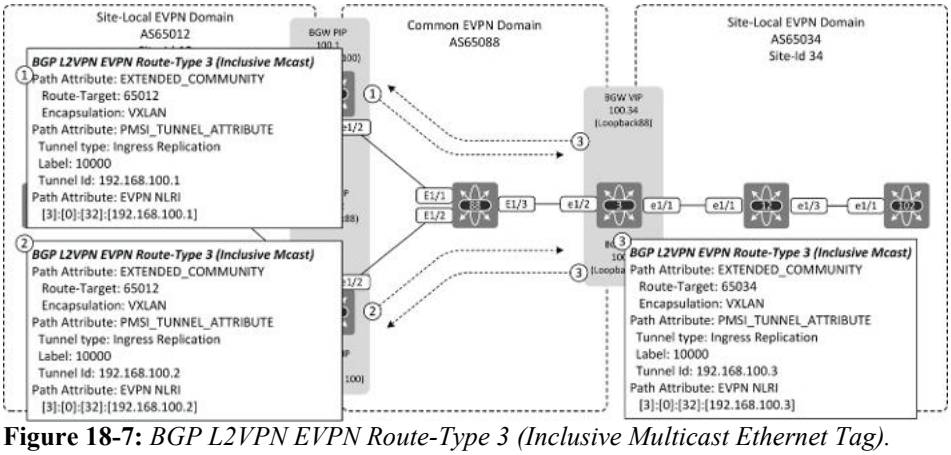


Figure 18-7: *BGP L2VPN EVPN Route-Type 3 (Inclusive Multicast Ethernet Tag).*

Examples 18-29 shows that BGW-1 received the BGP L2VPN EVPN Route-Type 3 NLRI information originated by BGW-3.

```
BGW-1# sh bgp l2vpn evpn route-type 3
!--< Comment: This is the local information advertise to BGW-3 >---------
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.1:32777    (L2VNI 10000)
BGP routing table entry for [3]:[0]:[32]:[192.168.100.1]/88, version 3
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
```

```
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    192.168.100.1 (metric 0) from 0.0.0.0 (192.168.77.1)
      Origin IGP, MED not set, localpref 100, weight 32768
      Origin flag 0x2
      Extcommunity: RT:65012:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.1

  Path-id 1 advertised to peers:
    192.168.77.88
!----< Comment: This is the information installed into BGP Loc-RIB ---->
BGP routing table entry for [3]:[0]:[32]:[192.168.100.3]/88, version 26
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported from 192.168.77.3:32777:[3]:[0]:[32]:[192.168.100.3]/88
  AS-Path: 65088 65034 , path sourced external to AS
    192.168.100.3 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65012:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.3

  Path-id 1 not advertised to any peer

!---< Comment: This is the information installed into Adj-RIB-In >-------
Route Distinguisher: 192.168.77.3:32777
BGP routing table entry for [3]:[0]:[32]:[192.168.100.3]/88, version 24
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: 65088 65034 , path sourced external to AS
    192.168.100.3 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65012:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.3

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

Example 18-29: *sh bgp l2vpn evpn route-type 3.*

Examples 18-30 shows that also BGW-2 has received the BGP L2VPN EVPN Route-Type 3 NLRI information originated by BGW-3.

```
BGW-2# sh bgp l2vpn evpn route-type 3
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.2:32777    (L2VNI 10000)
BGP routing table entry for [3]:[0]:[32]:[192.168.100.2]/88, version 3
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    192.168.100.2 (metric 0) from 0.0.0.0 (192.168.77.2)
      Origin IGP, MED not set, localpref 100, weight 32768
      Origin flag 0x2
      Extcommunity: RT:65012:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.2

  Path-id 1 advertised to peers:
    192.168.77.88
BGP routing table entry for [3]:[0]:[32]:[192.168.100.3]/88, version 26
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported from 192.168.77.3:32777:[3]:[0]:[32]:[192.168.100.3]/88
  AS-Path: 65088 65034 , path sourced external to AS
    192.168.100.3 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65012:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.3

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.3:32777
BGP routing table entry for [3]:[0]:[32]:[192.168.100.3]/88, version 24
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: 65088 65034 , path sourced external to AS
    192.168.100.3 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65012:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.3

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

Example 18-30: *sh bgp l2vpn evpn route-type 3.*

Examples 18-31 shows that BGW-3 has received the BGP L2VPN EVPN Route-Type 3 NLRI information originated by BGW-1 and BGW-2.

```
BGW-3# sh bgp l2vpn evpn route-type 3
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.77.1:32777
BGP routing table entry for [3]:[0]:[32]:[192.168.100.1]/88, version 7
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: 65088 65012 , path sourced external to AS
    192.168.100.1 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65034:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.1

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.2:32777
BGP routing table entry for [3]:[0]:[32]:[192.168.100.2]/88, version 15
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not
in HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
  AS-Path: 65088 65012 , path sourced external to AS
    192.168.100.2 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65034:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.2

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.3:32777    (L2VNI 10000)
BGP routing table entry for [3]:[0]:[32]:[192.168.100.1]/88, version 11
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported from 192.168.77.1:32777:[3]:[0]:[32]:[192.168.100.1]/88
  AS-Path: 65088 65012 , path sourced external to AS
    192.168.100.1 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65034:10000 ENCAP:8
      PMSI Tunnel Attribute:
```

```
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.1

  Path-id 1 not advertised to any peer
BGP routing table entry for [3]:[0]:[32]:[192.168.100.2]/88, version 17
Paths: (1 available, best #1)
Flags: (0x000012) (high32 00000000) on xmit-list, is in l2rib/evpn, is not in
HW
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported from 192.168.77.2:32777:[3]:[0]:[32]:[192.168.100.2]/88
  AS-Path: 65088 65012 , path sourced external to AS
    192.168.100.2 (metric 0) from 192.168.77.88 (192.168.77.88)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65034:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.2

  Path-id 1 not advertised to any peer
BGP routing table entry for [3]:[0]:[32]:[192.168.100.3]/88, version 3
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    192.168.100.3 (metric 0) from 0.0.0.0 (192.168.77.3)
      Origin IGP, MED not set, localpref 100, weight 32768
      Origin flag 0x2
      Extcommunity: RT:65034:10000 ENCAP:8
      PMSI Tunnel Attribute:
        flags: 0x00, Tunnel type: Ingress Replication
        Label: 10000, Tunnel Id: 192.168.100.3

  Path-id 1 advertised to peers:
    192.168.77.88
```

Example 18-31: *sh bgp l2vpn evpn route-type 3.*

P-Multicast Service Instance (PMSI) Path Attribute shown in capture 18-3 describes the PSMI tunnel end-point for Multi-Destination tree over a Common EVPN domain for VNI 10000. BGW that acts as a kind of PE device offers PMSI service for site-local devices, which means that the BGW switch has to be able to forward Multi-Destination traffic received form CE device, which in intra-site perspective are Leaf and Spine switches, over a Common EVPN Domain to BGW switches located on remote-site and another way around. The binary figures in front of the “MPLS label” describes the Virtual Network Identifier (VNI) for this Multi-destination tree. Binary value 0010.0111.0001 is in decimal notation 10000 (VNI used with VLAN 10).

```
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 99
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 76
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: 65012
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 16
            Carried extended communities: (2 communities)
                Route Target: 65012:10000
                Encapsulation: VXLAN Encapsulation [Transitive Opaque]
        Path Attribute - PMSI_TUNNEL_ATTRIBUTE
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: PMSI_TUNNEL_ATTRIBUTE (22)
            Length: 9
            Flags: 0
            Tunnel Type: Ingress Replication (6)
            0000 0000 0010 0111 0001 .... = MPLS Label: 625
            Tunnel ID: tunnel end point -> 192.168.100.1
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive,
            Type Code: MP_REACH_NLRI (14)
            Length: 28
            Address family identifier (AFI): Layer-2 VPN (25)
            Subsequent address family identifier (SAFI): EVPN (70)
            Next hop network address (4 bytes)
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (19 bytes)
                EVPN NLRI: Inclusive Multicast Route
                    Route Type: Inclusive Multicast Route (3)
                    Length: 17
                    Route Distinguisher: 0001c0a84d018009 (192.168.77.1:32777)
                    Ethernet Tag ID: 0
                    IP Address Length: 32
                    IPv4 address: 192.168.100.1
```

Capture 18-3: *BGP L2VPN EVPN Route-Type 3 – Inclusive Multicast Route (captured from BGW-1).*

Figure 18-8 illustrates the Multi-Destination forwarding path. PIM BiDir is used to build a Bidirectional Multicast tree in both Intra-Sites. Spine switches are defined as Pseudo Rendezvous Point (Pseudo RP) for Multicast Tree. Site-Local Leaf switches and BGW switches will join the Multicast tree. On the Common EVPN Domain side, BGW switches located in the different sites will form an Ingress-Replication path between each other. BGP L2VPN EVPN Route-Type 3 (Inclusive Multicast Route) EVPN NLRI is used for signaling.

In the case where Leaf-101 receives ingress L2 BUM frame from its connected host from VLAN 10 (VNI10000), it will check the Multicast Group attached to VNI 10000 (238.0.0.10) and sends the frame to the Spine-11 that is Pseudo-RP for group 238.0.0.10. (3) Spine-11 will forward L2 BUM frame out of the interfaces found from the *Outgoing Interface List* (OIL) for the Multicast Group 238.0.0.10. The OIL is build based on received PIM Join messages. Both BGW-1 and BGW-2 are joined to Mcast Group so Spine-11 will forward the L2 BUM frame to them. BGW-1 is selected to Designated Forwarder (DF) for VLAN 10, so it will send L2 BUM frame to BGW-3 over the Ingress-Replication tunnel formed over Common EVPN domain with VXLAN encapsulation. BGW-2 will not forward fame. The source IP address used in outer tunnel IP header is PIP of BGW-1. When BGW-3 receives the frame, it checks the VNI found from the VXLAN header and de-capsulate the frame. It forwards the frame to Mcast Group 238.0.0.10 (MG for VNI 10000 also in this site) RP Spine-12. Spine checks the OIL list and forward frame to Leaf-102.

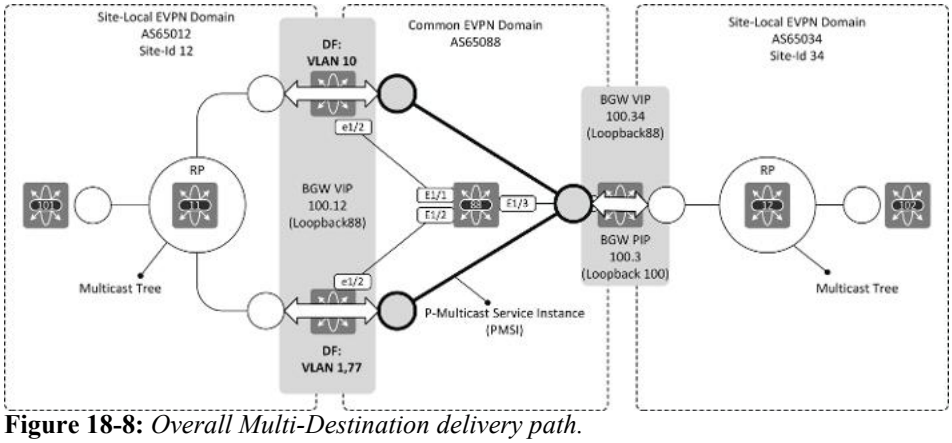


Figure 18-8: *Overall Multi-Destination delivery path.*

Fabric Link Failure

In the case where the BGW switch loses all intra-site links, it will stop advertising Shared Virtual IP (VIP) to its DCI Underlay Network BGP IPv4 Unicast peer. This way it makes sure that other switches do not consider it as a valid next for in ECMP decision process. In addition, it withdrawn all intra-site host-related MAC address information (Route-Type 2). It also stops advertising itself as an Ingress-Tunnel Endpoint by withdrawing the Inclusive Multicast Route (Route-Type 3). In addition, it withdrawn the Ethernet Segment Routes (Route-Type 4) even though they are not used outside the local site. It also withdrawn learned IP prefix routes (Route-Type 5), excluded locally connected prefixes from ether connected host or external IPv4 peer.

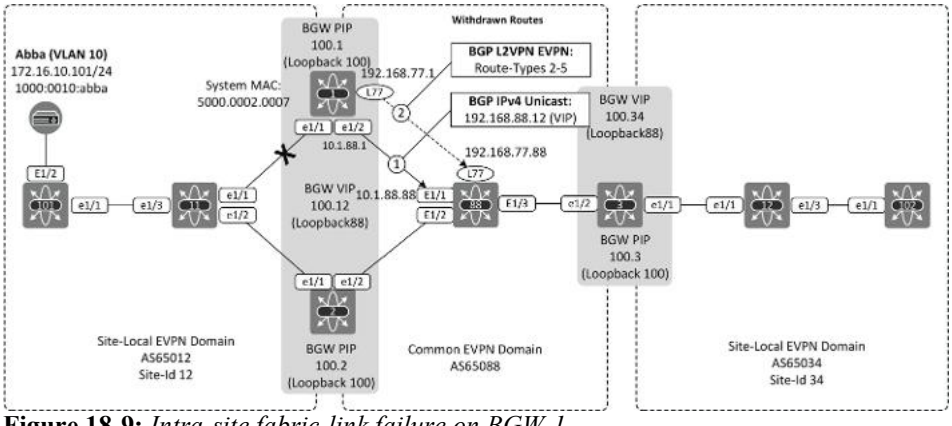


Figure 18-9: *Intra-site fabric-link failure on BGW-1.*

Normal State

Example 18-32 shows the BGP IPv4 Unicast entries installed into DC Core switch BGP table before fabric-link failure. DC Core Switch has learned the Shared VIP address used in Site-12 from its' BGP IPv4 Unicast peer switches BGW-1 and BGW-2.

```
RouteServer-1# sh ip bgp
BGP routing table information for VRF default, address family IPv4 Unicast
BGP table version is 21, Local Router ID is 192.168.77.88
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 -
best2

   Network            Next Hop            Metric     LocPrf     Weight Path
*>e10.1.88.0/24       10.1.88.1                0                  0 65012 ?
*>e10.2.88.0/24       10.2.88.2                0                  0 65012 ?
*>e10.3.88.0/24       10.3.88.3                0                  0 65034 ?
*>e10.88.1.0/24       10.1.88.1                0                  0 65012 ?
*>e10.88.2.0/24       10.2.88.2                0                  0 65012 ?
*>e10.88.3.0/24       10.3.88.3                0                  0 65034 ?
*>e192.168.0.1/32     10.1.88.1                0                  0 65012 ?
*>e192.168.0.2/32     10.2.88.2                0                  0 65012 ?
*>e192.168.0.3/32     10.3.88.3                0                  0 65034 ?
*>e192.168.77.1/32    10.1.88.1                0                  0 65012 ?
*>e192.168.77.2/32    10.2.88.2                0                  0 65012 ?
*>e192.168.77.3/32    10.3.88.3                0                  0 65034 ?
*>r192.168.77.88/32   0.0.0.0                  0        100       32768   ?
*>e192.168.88.12/32   10.1.88.1                0                  0 65012 ?
*|e                   10.2.88.2                0                  0 65012 ?
*>e192.168.88.34/32   10.3.88.3                0                  0 65034 ?
*>r192.168.88.88/32   0.0.0.0                  0        100       32768   ?
*>e192.168.100.1/32   10.1.88.1                0                  0 65012 ?
*>e192.168.100.2/32   10.2.88.2                0                  0 65012 ?
*>e192.168.100.3/32   10.3.88.3                0                  0 65034 ?
```

Example 18-32 *BGP IPv4 Unicast entries in DC Core switch.*

Example 18-33 shows the BGP L2VPN EVPN entries installed into BGW-3 BGP table before fabric-link failure. There is one Route-Type 4 entry (Ethernet Segment Route), one Route-Type 3 entry (Inclusive Multicast Route) and two Route-Type 2 entries (MAC Advertisement Route) first one for System MAC and the second one for host Abba.

```
BGW-3# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 47, Local Router ID is 192.168.77.3
<snipped>

   Network            Next Hop            Metric LocPrf  Weight Path
Route Distinguisher: 192.168.77.1:27001
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
                      192.168.100.1                          0 65088 65012 i

Route Distinguisher: 192.168.77.1:32777
*>e[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                          0 65088 65012 i
*>e[3]:[0]:[32]:[192.168.100.1]/88
                      192.168.100.1                          0 65088 65012 i

Route Distinguisher: 192.168.77.2:27001
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.77.1                           0 65088 65012 i

Route Distinguisher: 192.168.77.2:32777
*>e[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                          0 65088 65012 i
*>e[3]:[0]:[32]:[192.168.100.2]/88
                      192.168.100.2                          0 65088 65012 i

Route Distinguisher: 192.168.77.3:27001   (ES [0300.0000.0000.0c00.0309 0])
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
                      192.168.100.1                          0 65088 65012 i
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.77.1                           0 65088 65012 i
*>l[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.3]/136
                      192.168.100.3                        100      32768 i

Route Distinguisher: 192.168.77.3:32777    (L2VNI 10000)
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.88.12                          0 65088 65012 i
*>e[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                          0 65088 65012 i
*>e[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                          0 65088 65012 i
*>l[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.100.3                        100       32768 i
*>e[3]:[0]:[32]:[192.168.100.1]/88
                      192.168.100.1                          0 65088 65012 i
*>e[3]:[0]:[32]:[192.168.100.2]/88
                      192.168.100.2                          0 65088 65012 i
*>l[3]:[0]:[32]:[192.168.100.3]/88
                      192.168.100.3                        100       32768 i

Route Distinguisher: 192.168.77.101:32777
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.88.12                          0 65088 65012 i
```

Example 18-33 *BGP L2VPN EVPN entries in BGW-3.*

The Protocol, Link and Admin status of Loopback 88 (VIP) is UP on BGW-1.

```
BGW-1(config-if)# sh ip int bri

IP Interface Status for VRF "default"(1)
Interface            IP Address      Interface Status
Lo0                  192.168.0.1     protocol-up/link-up/admin-up
Lo77                 192.168.77.1    protocol-up/link-up/admin-up
Lo88                 192.168.88.12   protocol-up/link-up/admin-up
Lo100                192.168.100.1   protocol-up/link-up/admin-up
Eth1/1               10.1.11.1       protocol-up/link-up/admin-up
Eth1/2               10.1.88.1       protocol-up/link-up/admin-up
Eth1/3               10.11.1.1       protocol-up/link-up/admin-up
Eth1/4               10.88.1.1       protocol-up/link-up/admin-up
```

Example 18-34 *Interface Loopback 88 UP on BGW-1.*

Fabric-Link Failure

The fabric-link failure is simulated by shutting down the fabric-link Interface e1/1. When BGW-1 notices this, it changes the Interface Loopback 88 link-state to down.

```
BGW-1(config-if)# sh ip int bri

IP Interface Status for VRF "default"(1)
Interface            IP Address      Interface Status
Lo0                  192.168.0.1     protocol-up/link-up/admin-up
Lo77                 192.168.77.1    protocol-up/link-up/admin-up
Lo88                 192.168.88.12   protocol-down/link-down/admin-up
Lo100                192.168.100.1   protocol-up/link-up/admin-up
Eth1/1               10.1.11.1       protocol-down/link-down/admin-down
Eth1/2               10.1.88.1       protocol-up/link-up/admin-up
Eth1/3               10.11.1.1       protocol-up/link-up/admin-up
Eth1/4               10.88.1.1       protocol-up/link-up/admin-up
```

Example 18-35 *Interface Loopback 88 DOWN on BGW-1.*

Example 18-36 verifies that the Fabric-Link is also down

```
BGW-1# sh nve multisite fabric-links
Interface      State
---------      -----
Ethernet1/1    Down
```

Example 18-36 *sh nve multisite fabric-links on BGW-1.*

Capture 18-4 shows that BGW-1 sends MP_Unreach_NLRI concerning the IP address of Loopback 88 to DC Core switch over the BGP IPv4 Unicast peering.

```
Internet Protocol Version 4, Src: 10.1.88.1, Dst: 10.1.88.88
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 35
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 12
    Path attributes
        Path Attribute - MP_UNREACH_NLRI
            Flags: 0x90, Optional, Extended-Length, Non-transitive
            Type Code: MP_UNREACH_NLRI (15)
            Length: 8
            Address family identifier (AFI): IPv4 (1)
            Subsequent address family identifier (SAFI): Unicast (1)
            Withdrawn routes (5 bytes)
                192.168.88.12/32
```

Capture 18-4: *Withdrawn route 192.168.88.12/32 by BGW-1.*

As a result, the DC Core switch removes the routing entry from its BGP IPv4 table (example 18-37), and now the ip 192.168.88.12/32 is only reachable via BGW-2.

```
RouteServer-1# sh ip bgp
BGP routing table information for VRF default, address family IPv4 Unicast
BGP table version is 22, Local Router ID is 192.168.77.88
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 -
best2

   Network            Next Hop            Metric     LocPrf     Weight Path
*>e10.1.88.0/24       10.1.88.1                0                     0 65012 ?
*>e10.2.88.0/24       10.2.88.2                0                     0 65012 ?
*>e10.3.88.0/24       10.3.88.3                0                     0 65034 ?
*>e10.88.1.0/24       10.1.88.1                0                     0 65012 ?
*>e10.88.2.0/24       10.2.88.2                0                     0 65012 ?
*>e10.88.3.0/24       10.3.88.3                0                     0 65034 ?
*>e192.168.0.1/32     10.1.88.1                0                     0 65012 ?
*>e192.168.0.2/32     10.2.88.2                0                     0 65012 ?
*>e192.168.0.3/32     10.3.88.3                0                     0 65034 ?
*>e192.168.77.1/32    10.1.88.1                0                     0 65012 ?
*>e192.168.77.2/32    10.2.88.2                0                     0 65012 ?
*>e192.168.77.3/32    10.3.88.3                0                     0 65034 ?
*>r192.168.77.88/32   0.0.0.0                  0        100      32768 ?
*>e192.168.88.12/32   10.2.88.2                0                     0 65012 ?
*>e192.168.88.34/32   10.3.88.3                0                     0 65034 ?
*>r192.168.88.88/32   0.0.0.0                  0        100      32768 ?
*>e192.168.100.1/32   10.1.88.1                0                     0 65012 ?
*>e192.168.100.2/32   10.2.88.2                0                     0 65012 ?
*>e192.168.100.3/32   10.3.88.3                0                     0 65034 ?
```

Example 18-37 *Loopback88 of BGW-1 removed from the BGP IPv4 table of DC Core switch.*

BGW-1 has also withdrawn all Route-type 2-5. Example 18-38 shows that the routes to abba and looback 88 withdrawned by BGW-1 are removed from BGW-3 BGP L2VPN EVPN table.

```
BGW-3# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 87, Local Router ID is 192.168.77.3
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 -
best2

   Network            Next Hop            Metric  LocPrf   Weight Path
Route Distinguisher: 192.168.77.2:27001
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                         0 65088 65012 i

Route Distinguisher: 192.168.77.2:32777
*>e[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                         0 65088 65012 i
*>e[3]:[0]:[32]:[192.168.100.2]/88
                      192.168.100.2                         0 65088 65012 i

Route Distinguisher: 192.168.77.3:27001   (ES [0300.0000.0000.0c00.0309 0])
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                          0 65088 65012 i
*>l[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.3]/136
                      192.168.100.3                        100      32768 i

Route Distinguisher: 192.168.77.3:32777    (L2VNI 10000)
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.88.12                         0 65088 65012 i
*>e[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                         0 65088 65012 i
*>l[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.100.3                       100       32768 i
*>e[3]:[0]:[32]:[192.168.100.2]/88
                      192.168.100.2                         0 65088 65012 i
*>l[3]:[0]:[32]:[192.168.100.3]/88
                      192.168.100.3                       100       32768 i

Route Distinguisher: 192.168.77.101:32777
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.88.12                         0 65088 65012 i
```

Example 18-38 *BGP table of BGW-3 after fabric-link failure in BGW-1.*

Fabric-Link Recovery

When fabric-link is brought back up on BGW-1, the Admin state is changed to UP state while the Operational state is still kept on DOWN state. BGW-1 starts the *Delay-Restore Timer* as can be seen from the example 18-39 and 18-40.

```
BGW-1# show nve interface nve 1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 5000.0002.0007
 Host Learning Mode: Control-Plane
 Source-Interface: loopback100 (primary: 192.168.100.1, secondary: 0.0.0.0)
 Source Interface State: Up
 Virtual RMAC Advertisement: No
 NVE Flags:
 Interface Handle: 0x49000001
 Source Interface hold-down-time: 180
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: N/A
 Virtual Router MAC Re-origination: 0200.c0a8.580c
 Interface state: nve-intf-add-complete
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 236 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Down)
 Multisite bgw-if oper down reason:
```

Example 18-39 *Delay Restore Timer on BGW-1.*

```
BGW-1# show nve interface nve 1 detail | i  Multisite
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 20 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Down)
 Multisite bgw-if oper down reason:
```

Example 18-40 *Delay Restore Timer on BGW-1.*

After 300 seconds, BGW-1 changes the Operational state of Interface Loopback to UP state as shown in examples 18-41 and 18-42.

```
BGW-1# show nve interface nve 1 detail | i  Multisite
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 0 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Up)
 Multisite bgw-if oper down reason:
```

Example 18-41 *Delay Restore Timer on BGW-1.*

```
BGW-1# sh ip int bri

IP Interface Status for VRF "default"(1)
Interface            IP Address      Interface Status
Lo0                  192.168.0.1     protocol-up/link-up/admin-up
Lo77                 192.168.77.1    protocol-up/link-up/admin-up
Lo88                 192.168.88.12   protocol-up/link-up/admin-up
Lo100                192.168.100.1   protocol-up/link-up/admin-up
Eth1/1               10.1.11.1       protocol-up/link-up/admin-up
Eth1/2               10.1.88.1       protocol-up/link-up/admin-up
Eth1/3               10.11.1.1       protocol-up/link-up/admin-up
Eth1/4               10.88.1.1       protocol-up/link-up/admin-up
```

Example 18-42 *Loopback 88 status after recovery on BGW-1.*

The network has recovered as can be seen from the examples 18-43 and 18-44.

```
RouteServer-1# sh ip bgp
BGP routing table information for VRF default, address family IPv4 Unicast
BGP table version is 23, Local Router ID is 192.168.77.88
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 -
best2

   Network            Next Hop            Metric   LocPrf   Weight Path
*>e10.1.88.0/24       10.1.88.1                0                0 65012 ?
*>e10.2.88.0/24       10.2.88.2                0                0 65012
*>e10.3.88.0/24       10.3.88.3                0                0 65034 ?
*>e10.88.1.0/24       10.1.88.1                0                0 65012 ?
*>e10.88.2.0/24       10.2.88.2                0                0 65012 ?
*>e10.88.3.0/24       10.3.88.3                0                0 65034 ?
*>e192.168.0.1/32     10.1.88.1                0                0 65012 ?
*>e192.168.0.2/32     10.2.88.2                0                0 65012 ?
*>e192.168.0.3/32     10.3.88.3                0                0 65034 ?
*>e192.168.77.1/32    10.1.88.1                0                0 65012 ?
*>e192.168.77.2/32    10.2.88.2                0                0 65012 ?
*>e192.168.77.3/32    10.3.88.3                0                0 65034 ?
*>r192.168.77.88/32   0.0.0.0                  0      100  32768        ?
*|e192.168.88.12/32   10.1.88.1                0                0 65012 ?
*>e                   10.2.88.2                0                0 65012 ?
*>e192.168.88.34/32   10.3.88.3                0                0 65034 ?
*>r192.168.88.88/32   0.0.0.0                  0      100  32768        ?
*>e192.168.100.1/32   10.1.88.1                0                0 65012 ?
*>e192.168.100.2/32   10.2.88.2                0                0 65012 ?
*>e192.168.100.3/32   10.3.88.3                0                0 65034 ?
```

Example 18-43:*BGP IPv4 table on DC Core Switch after recovery.*

```
BGW-3# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 103, Local Router ID is 192.168.77.3
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-
best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist, I-
injected
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath, & - backup, 2 -
best2

   Network            Next Hop            Metric  LocPrf   Weight Path
Route Distinguisher: 192.168.77.1:27001
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
                      192.168.100.1                         0 65088 65012 i

Route Distinguisher: 192.168.77.1:32777
*>e[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                         0 65088 65012 i
*>e[3]:[0]:[32]:[192.168.100.1]/88
                      192.168.100.1                         0 65088 65012 i

Route Distinguisher: 192.168.77.2:27001
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                         0 65088 65012 i

Route Distinguisher: 192.168.77.2:32777
*>e[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                         0 65088 65012 i
```

```
*>e[3]:[0]:[32]:[192.168.100.2]/88
                      192.168.100.2                          0 65088 65012 i

Route Distinguisher: 192.168.77.3:27001   (ES [0300.0000.0000.0c00.0309 0])
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
                      192.168.100.1                          0 65088 65012 i
*>e[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                          0 65088 65012 i
*>l[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.3]/136
                      192.168.100.3                     100      32768  i

Route Distinguisher: 192.168.77.3:32777    (L2VNI 10000)
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.88.12                          0 65088 65012 i
*>e[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                          0 65088 65012 i
*>e[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                          0 65088 65012 i
*>l[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.100.3                     100      32768  i
*>e[3]:[0]:[32]:[192.168.100.1]/88
                      192.168.100.1                          0 65088 65012 i
*>e[3]:[0]:[32]:[192.168.100.2]/88
                      192.168.100.2                          0 65088 65012 i
*>l[3]:[0]:[32]:[192.168.100.3]/88
                      192.168.100.3                     100      32768  i

Route Distinguisher: 192.168.77.101:32777
*>e[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.88.12                          0 65088 65012 i
```

Example 18-44:*BGP L2VPN EVPN table on DC Core Switch after recovery.*

DCI-Link Failure

When all of the DCI links of BGW are down, it stops advertising VIP address to Intra-Site peer just like in case of previously discussed Fabric-Link failure. Naturally, it also stops advertising routes learned via DCI link due to link failure. What it still does, it continues acting as a regular Leaf switch. If it has connected hosts or external peers, it continues to advertise prefix attached/learned from those.

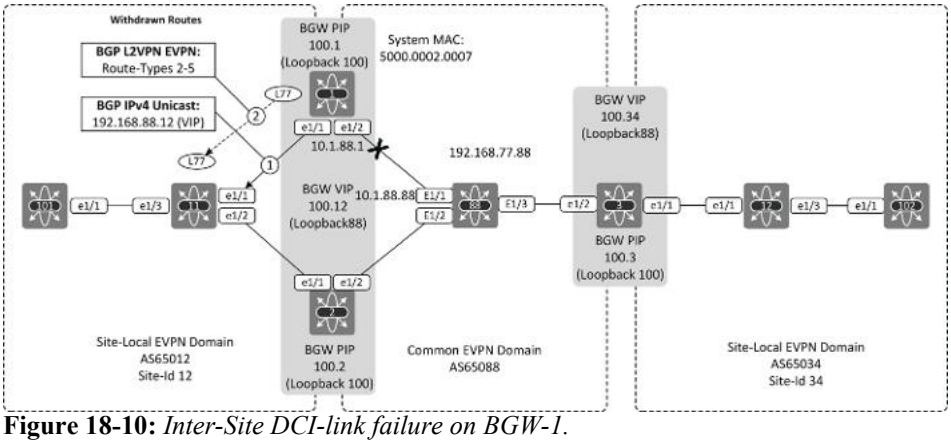


Figure 18-10: *Inter-Site DCI-link failure on BGW-1.*

Normal State

Example 18-45 shows that Spine-11 has learned Site-12 Shared VIP from both Intra-Site BGW switches via OSPF (Underlay Network).

```
Spine-11# sh ip route 192.168.88.12
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.88.12/32, ubest/mbest: 2/0
    *via 10.1.11.1, Eth1/1, [110/41], 01:28:36, ospf-UNDERLAY-NET, intra
    *via 10.2.11.2, Eth1/2, [110/41], 01:28:36, ospf-UNDERLAY-NET, intra
```

Example 18-45:*RIB on Spine-11 in normal situation.*

Example 18-46 shows that Spine-11 uses both BGW-1 and BG-2 for load sharing data to Inter-Site. Note that the MAC address 5000.0004.0007 is the System MAC address of BGW-3 on Site-34.

```
Spine-11# sh bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 122, Local Router ID is 192.168.77.11
<snipped>
   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.1:27001
*>i[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.1]/136
                      192.168.100.1                     100          0 i

Route Distinguisher: 192.168.77.1:32777
*>i[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                     100          0 i

Route Distinguisher: 192.168.77.2:27001
*>i[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                     100          0 i

Route Distinguisher: 192.168.77.2:32777
*>i[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                     100          0 i

Route Distinguisher: 192.168.77.3:32777
*>i[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.88.12                     100          0 65088
65034 i
* i                   192.168.88.12                     100          0 65088
65034 i

Route Distinguisher: 192.168.77.101:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.100.101                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.101]/272
                      192.168.100.101                   100          0 i
```

Example 18-46: *BGP L2VPN EVPN table on Spine-11 in normal situation.*

DCI Link Failure

The DCI link failure is demonstrated by shutting down the DCI interface e1/2. The state of the link is verified on example 1-47 and 1-48.

```
BGW-1# sh nve multisite dci-links
Interface      State
---------      -----
Ethernet1/2    Down
```

Example 18-47: *sh nve multisite dci-links on Spine-11.*

```
BGW-1# sh ip int bri

IP Interface Status for VRF "default"(1)
Interface            IP Address      Interface Status
Lo0                  192.168.0.1     protocol-up/link-up/admin-up
Lo77                 192.168.77.1    protocol-up/link-up/admin-up
Lo88                 192.168.88.12   protocol-down/link-down/admin-up
Lo100                192.168.100.1   protocol-up/link-up/admin-up
Eth1/1               10.1.11.1       protocol-up/link-up/admin-up
Eth1/2               10.1.88.1       protocol-down/link-down/admin-down
Eth1/3               10.11.1.1       protocol-up/link-up/admin-up
Eth1/4               10.88.1.1       protocol-up/link-up/admin-up
```

Example 18-48: *sh ip int bri on BGW-1.*

Example 18-49 below shows that reason for Down-state is “DCI Isolated”.

```
BGW-1# show nve interface nve 1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 5000.0002.0007
 Host Learning Mode: Control-Plane
 Source-Interface: loopback100 (primary: 192.168.100.1, secondary: 0.0.0.0)
 Source Interface State: Up
 Virtual RMAC Advertisement: No
 NVE Flags:
 Interface Handle: 0x49000001
 Source Interface hold-down-time: 180
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: N/A
 Virtual Router MAC Re-origination: 0200.c0a8.580c
 Interface state: nve-intf-add-complete
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 0 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Down)
 Multisite bgw-if oper down reason:  DCI isolated.
```

Example 19-49: *show nve interface nve 1 detail on Spine-11.*

BGW-1 withdrawn the VIP and now Spine-11 has only one destination to Intra-Site VIP address via BGW-2.

```
Spine-11# sh ip route 192.168.88.12
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.88.12/32, ubest/mbest: 1/0
    *via 10.2.11.2, Eth1/2, [110/41], 00:00:45, ospf-UNDERLAY-NET, intra
```

Example 18-50: *show ip route 192.168.88.12 on Spine-11.*

BGW-1 also withdrawn all Route-Type 2-5 routes received via DCI link. Now Spine-11 learns Inter-Site routes only via BGW-2.

```
Spine-11# sh bgp l2vpn evpn
<snipped>

   Network            Next Hop            Metric     LocPrf     Weight Path
Route Distinguisher: 192.168.77.1:32777
*>i[2]:[0]:[0]:[48]:[5000.0002.0007]:[0]:[0.0.0.0]/216
                      192.168.100.1                     100          0 i

Route Distinguisher: 192.168.77.2:27001
*>i[4]:[0300.0000.0000.0c00.0309]:[32]:[192.168.100.2]/136
                      192.168.100.2                     100          0 i

Route Distinguisher: 192.168.77.2:32777
*>i[2]:[0]:[0]:[48]:[5000.0003.0007]:[0]:[0.0.0.0]/216
                      192.168.100.2                     100          0 i

Route Distinguisher: 192.168.77.3:32777
*>i[2]:[0]:[0]:[48]:[5000.0004.0007]:[0]:[0.0.0.0]/216
                      192.168.88.12                     100          0 65088
65034 i

Route Distinguisher: 192.168.77.101:32777
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[0]:[0.0.0.0]/216
                      192.168.100.101                   100          0 i
*>i[2]:[0]:[0]:[48]:[1000.0010.abba]:[32]:[172.16.10.101]/272
                      192.168.100.101                   100          0 i
```

Example 18-51: *sh bgp l2vpn evpn on Spine-11.*

DCI Link Recovery

The recovery process is the same as in the case of Fabric-Link failure. BGW-1 starts a Delay-Restore timer that is set to 300 seconds.

```
BGW-1# show nve interface nve 1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 5000.0002.0007
 Host Learning Mode: Control-Plane
 Source-Interface: loopback100 (primary: 192.168.100.1, secondary: 0.0.0.0)
 Source Interface State: Up
 Virtual RMAC Advertisement: No
 NVE Flags:
 Interface Handle: 0x49000001
```

```
 Source Interface hold-down-time: 180
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: N/A
 Virtual Router MAC Re-origination: 0200.c0a8.580c
 Interface state: nve-intf-add-complete
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 295 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Down)
 Multisite bgw-if oper down reason:
```

Example 18-52: *Delay-restore timer start on BGW-1.*

BGW-1 changes the interface Loopback 88 status to UP after 300 seconds and start the normal operation.

```
BGW-1# show nve interface nve 1 detail | i Multisite
 Multisite delay-restore time: 300 seconds
 Multisite delay-restore time left: 0 seconds
 Multisite bgw-if: loopback88 (ip: 192.168.88.12, admin: Up, oper: Up)
 Multisite bgw-if oper down reason:
```

Example 18-53: *Delay-restore timer stop on BGW-1.*

Note that during failure, the BGW-2 will take over the Designated Forwarder role for all Intra-Site VNIs.

References

Building Data Center with VXLAN BGP EVPN – A Cisco NX-OS Perspective
ISBN-10: 1-58714-467-0 – Krattiger Lukas, Shyam Kapadia, and Jansen Davis

[RFC 6513] E. Rosen and R. Aggarwal, “Multicast in MPLS/BGP IP VPNs”, RFC 6513, February 2012.

[RFC 6514] R. Aggarwal et al., “BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs”, RFC 6514, February 2012.

[RFC 7432] A. Sajssil et al., “BGP MPLS-Based Ethernet VPN”, RFC 7432, February 2015.

[M-S EVPN] R. Sharma et al., “Multi-site EVPN based VXLAN using Border Gateways, July 2017.

VXLAN EVPN Multi-Site Design and Deployment:
https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/white-paper-c11-739942.html

Chapter 19: Tenant Routed Multicast in VXLAN Fabric

This chapter introduces the "*Tenant Routed Multicast" (TRM)* solution in BGP EVPN VXLAN fabric. TRM relies on standard-based BGP IPv4 MVPN Address-Family [RFC 6513] and [RFC 6514]. Figure 19-1 illustrates the basic idea of TRM operation. (1) Leaf switches establish a Multicast tunnel per tenant, which they are using for forwarding tenant-specific Intra/Inter-VN Multicast traffic. (2) When Leaf -101 starts receiving Multicast flow from host Cafe to group 239.77.77.77, it updates its tenant-specific MRIB table and generates an MVPN route-type 5 "*Source Active Auto-Discovery (SA A-D)*" route (3), where the MP-REACH-NLRI carries information about Source-Specific group (S, G). This route-type is used for discovering if there are any Multicast receivers behind remote leafs. When Leaf-102 receives the BGP Update message, it imports information into BGP table. (4) Next, host Bebe sends an IGMP join message. (5) Leaf-102 updates its MRIB and then it generates the *MVPN route-type 7 "Source-Tree Join*" route. By doing this, it informs the source that it has local receivers for Multicast group 239.77.77.77. Leaf-101 installs the route into BGP table and updates its MRIB by adding the NVE interface into group-specific OIL. Then it starts forwarding Multicast flow received from host Cafe to core over Source-Specific Multicast delivery tree which is actually tunneled over tenant-specific Multicast tunnel. In other words the destination IP address in outer IP header uses Multicast tunnel group address 238.101.102.103 and the source IP address is taken from interface NVE1. By doing this, the actual tenant-specific Inter-VNI Multicast flows are totally transparent to Spine switch.

This chapter starts by explaining how Multicast tunnels used for Intra-VN (L2), and Inter-VN (L3) are established and how MRIB is constructed. Then it introduces the configuration required for TRM. The last two sections discuss BGP MVPN Control Plane operation and Multicast data forwarding Data Plane operation.

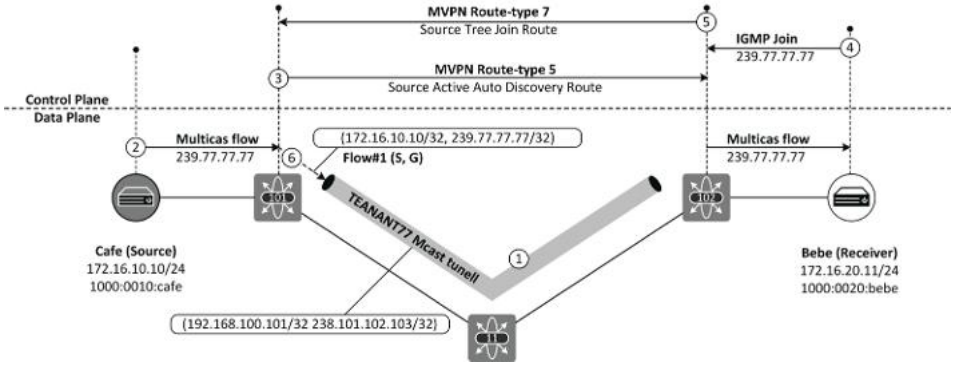


Figure-19-1: *Tenant Routed Multicast (TRM) Topology.*

Figure 19-2 shows the IP addressing scheme and figure-19-3 shows the logical structure used in this chapter.

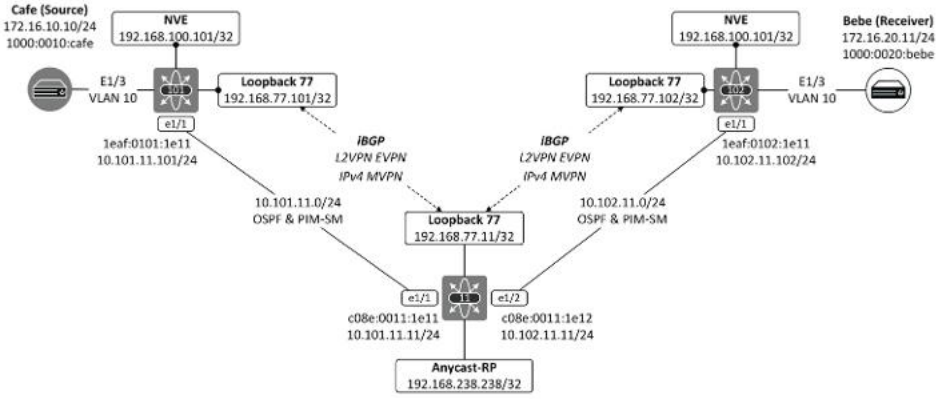


Figure-19-2: *IP addressing scheme.*

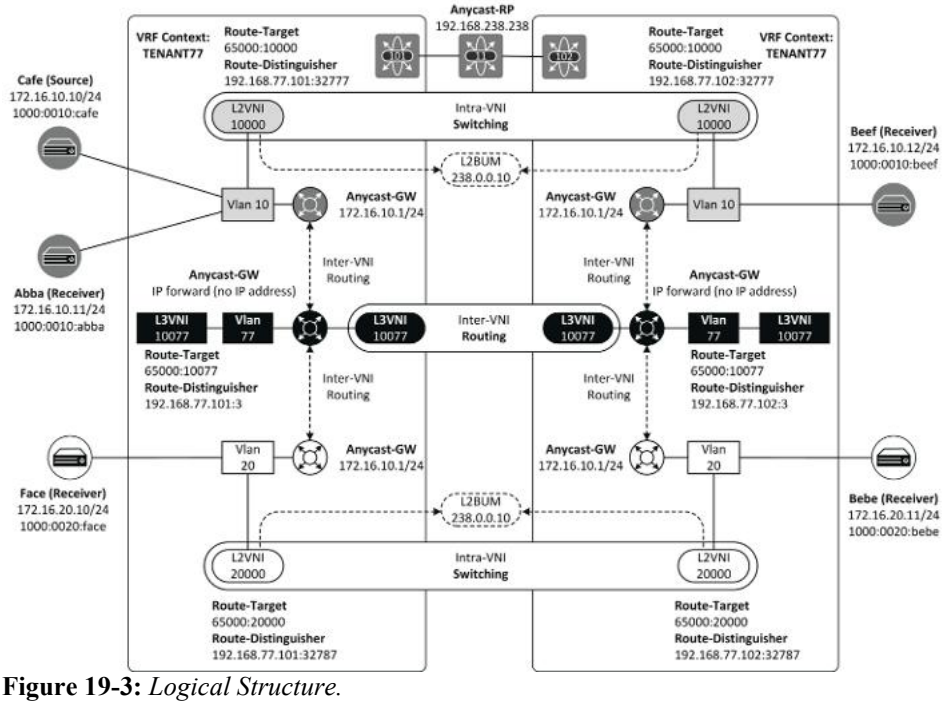


Figure 19-3: *Logical Structure.*

Underlay Multicast Routing

Tenant Routed Multicast (TRM) requires a Multicast enabled Underlay Network. This section starts by discussing a Multicast routing solution used in Underlay Network. It first discusses PIM peering and then it explains the Shared Tree and Source-Specific Tree operation and their usage from the Intra-VN (L2VNI) perspective.

PIM neighbor establishment process

In order to exchange Multicast routing information, switches first need to establish a PIM neighbor relationship. This process beginning when a switch starts sending periodic PIM hello messages to 224.0.0.13 (All-PIM-Routers) out of its PIM interface. When a switch receives a PIM hello message from one of its PIM enabled interface, it installs the source IP address and the ingress interface information in its PIM neighbor table. Figure 19-4 illustrates the PIM neighbor process.

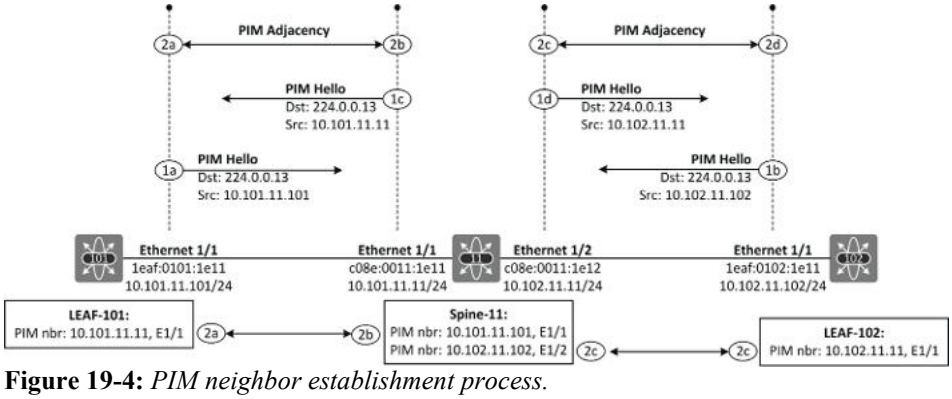


Figure 19-4: *PIM neighbor establishment process.*

```
Leaf-101# sh ip pim neighbor | i 10.101
10.101.11.11     Ethernet1/1    00:37:55  00:01:39  1  yes      n/a      no
```

Example 19-1: *PIM neighbor-table Leaf-101.*

```
Spine-11# sh ip pim neighbor | i 10.101
10.101.11.101    Ethernet1/1    00:41:33  00:01:42  1  yes      n/a      no
```

Example 19-2: *PIM neighbor-table Leaf-102.*

Shared Multicast Tree for Intra-VN

In a traditional networks, the *"Shared tree"* also called "*RP-Tree (RPT)"* is used when a host wants to receive Multicast flow from some specific Multicast group but it does not know who is a source. From the VXLAN L2VNI perspective, the Shared tree is also used for L2 Broadcast and Unknown Unicast messages. ARP-request message falls into Broadcast category and can be used as an example of Shared tree usage. Figure 19-5 illustrates the structure of the Shared tree for group 238.0.0.10 (Multicast Group for L2VNI10000/VLAN10) rooted by Spine-11. When host Cafe connected to Leaf-101 wants to send data to host Beef within the same subnet, it has to resolve the MAC address of host Beef. It generates an ARP-request by using a Broadcast MAC address FF:FF:FF:FF:FF:FF as a destination. When Leaf-101 receives the frame, it first encapsulates the original frame inside VXLAN tunnel headers where the outer destination IP address is L2VNI specific Multicast group address. Then it sends the encapsulated message to the RP. When Spine-11 receives the packet, it forwards the packet down to Shared tree-based on Outgoing Interface List of MRIB entry.

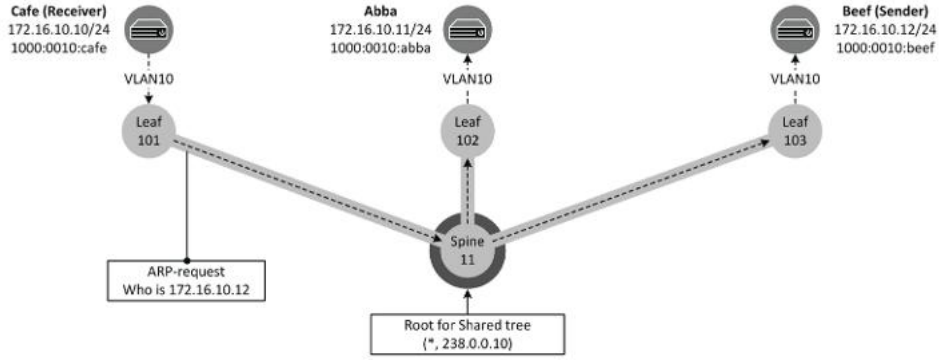


Figure 19-5: *Underlay Network Shared tree rooted by Spine-11.*

Joining to Intra-VN Shared Tree

Figure 19-6 illustrates how leaf switches join to Shared Tree. Multicast group 238.0.0.10 is used for L2VNI10000 L2BUM traffic. When the interface NVE1 comes up on Leaf-101 and Leaf-102 they send a PIM Join message using IP address 224.0.0.13 *(All-PIM-Routers)* as the destination IP address. The PIM Join message contains the information about the Multicast group whereof switch wants to receive Mcast flows (Group-1: 238.0.0.10 in capture 19-1). In addition, it describes that switch will forward Multicast flows from the source 192.168.238.238 when the flow is received from the interface where the PIM-Join is sent out ("Number of Joins" in capture 19-1). In other words, Multicast flow to group 238.0.0.10 from Spine-11 is forwarded when the flow is received from the interface E1/1. In addition, the PIM-join message describes that the switch will NOT forward Multicast flows from the source 192.168.100.101 (in case of Leaf-101) if the flow is received from the interface where the PIM-Join is sent out ("Number of Prunes" in capture 19-1). In other words, if leaf switch receives Multicast flow from the interface E1/1, it does not forward receive flow if the source is its own NVE interface IP address. Capture 19-1 shows the PIM-Join messages originated by Leaf-101. When Spine-11 receives PIM Join message, which carries group information for witch Spine-11 is RP, it updates Shared Tree (*, 238.0.0.10) MRIB entry by adding ingress interface into Outgoing Interface List (OIL).

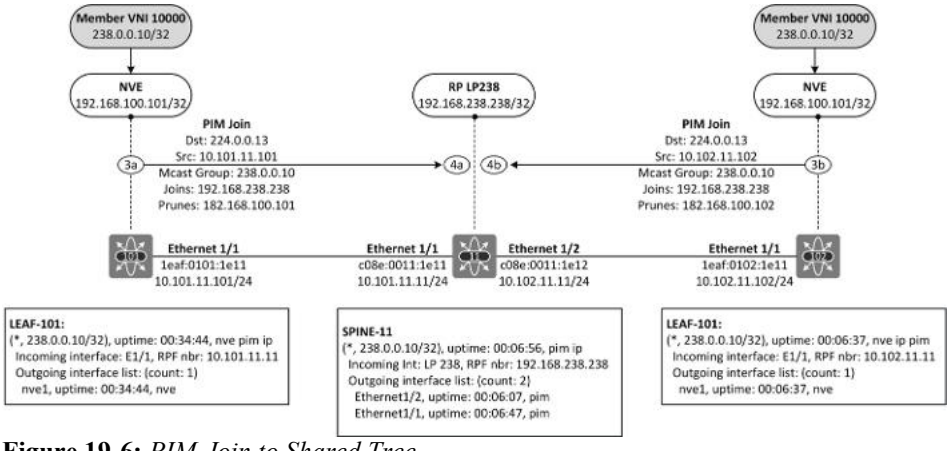


Figure 19-6: *PIM-Join to Shared Tree.*

Capture 19-1 shows the PIM-Join message sent by Leaf-101.

```
Ethernet II, Src: 1e:af:01:01:1e:11 , Dst: 01:00:5e:00:00:0d
Internet Protocol Version 4, Src: 10.101.11.101, Dst: 224.0.0.13
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0011 = Type: Join/Prune (2)
    Reserved byte(s): 00
    Checksum: 0xe602 [correct]
    [Checksum Status: Good]
    PIM Options
        Upstream-neighbor: 10.101.11.11
        Reserved byte(s): 00
        Num Groups: 2
        Holdtime: 210
        Group 1: 238.0.0.10/32
            Num Joins: 1
                IP address: 192.168.238.238/32 (SWR)
            Num Prunes: 1
                IP address: 192.168.100.101/32 (SR)
```

Capture 19-1: *PIM-Join message sent by Leaf-101.*

Joining to Intra-VN Source-Specific Tree

When a local host wants to join to some specific Multicast group, it sends an IGMP-Join message to destination 224.0.0.1 *(ALL-SYSTEM-IN-THIS-SUBNET)*. When an L2 segment gateway router hears this message, it creates a Shared tree entry (*, G) for the group in its MRIB. In addition, it generates a group-specific PIM-Join message towards RP. When RP receives the PIM-Join message, it also creates (*, G) entry in its MRIB if this a first PIM-Join for group and installs ingress interface into the Outgoing Interface List (OIL). In case where the source of the group has already started sending flow, the RP beginning to forward Multicast data out of the interface just added to OIL. When the gateway router, that asked Multicast flow by sending PIM-Join message, received the first packet of the Multicast flow, it learns the IP address of source and creates Source-Specific entry in its MRIB (S, G). It could now send a new PIM-Join message to source for joining the Source-Specific Tree and prune itself from the Shared Tree. This is part of the Multicast path-optimization process.

This section describes how leaf switches in VXLAN fabric join to Source-Specific Tree. The process starts when an NVE1 interface on Leaf-101 comes up (figure 19-7). L2VNI10000 uses Multicast group 238.0.0.10 address for L2BUM. Leaf-101 sends a VLAN10/L2VNI10000 specific "*IGMP Membership Query"* message to destination IP address 224.0.0.1 to check if there is someone who is interested in to receive a group-specific Mcast flow. The message is sent out of the local VLAN10 interfaces. In addition, the message is sent over the NVE1interface as an Intra-VN bridged traffic using VXLAN tunnel header with VN-Id 10000. The destination IP address in VXLAN tunnel header is set to Multicast group address 238.0.0.10. Leaf-101 forwards the original IGMP messages, encapsulated with VXLAN tunnel header as a *PIM-Register* message, which adds an additional tunnel header (IP and PIM) where the destination IP address is 192.168.238.238 (RP Spine-11). This happens because Leaf-101 considers this packet as the first packet for Multicast flow for group 238.0.0.10. This means that the original IGMP Membership Query is double encapsulates with VXLAN and PIM-Register messages when it is sent to RP.

When Spine-11 receives the IGMP Membership Query message it removes the Out-most IP header and PIM header before it forwards the packet based on the Shared tree (*, 238.0.0.10) OIL to leaf-102. In addition, it asks Leaf-101 to send the rest of the Mcast flow to the group using a group-specific destination address by replying to Leaf-101 with PIM Register-Stop message. In addition, Spine-11 uses the PIM Register-Stop message to inform Leaf-101 its willingness to join the Source-Specific Tree (192.168.100.101, 238.0.0.10).

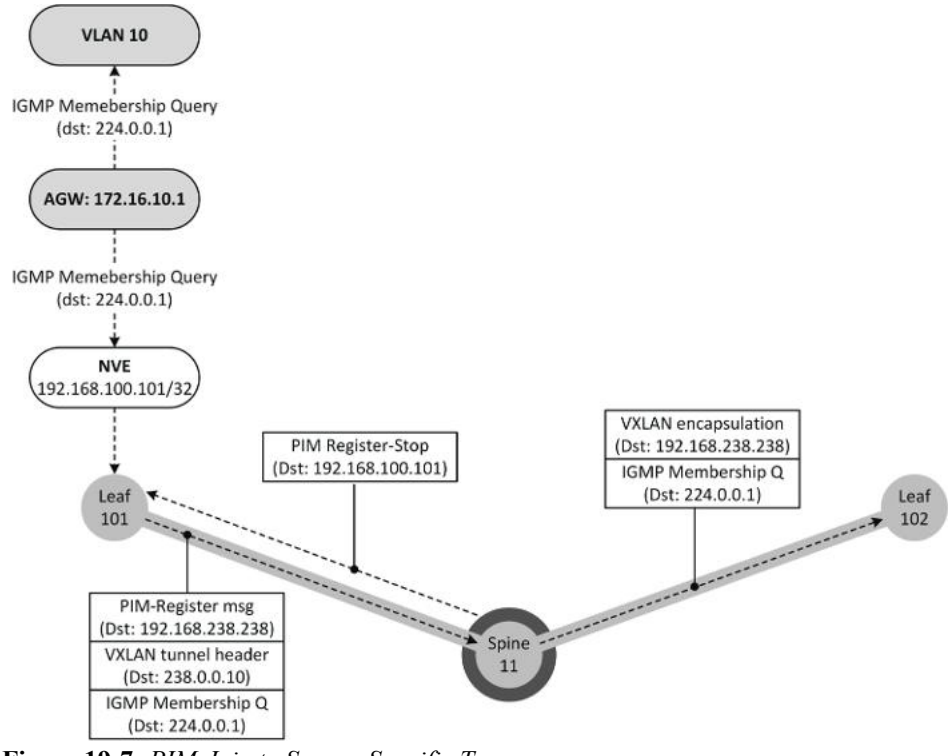


Figure 19-7: *PIM-Join to Source-Specific Tree.*

Capture 19-2 shows the IGMP Membership Query message sent by Leaf-101.

```
Ethernet II, Src: 1e:af:01:01:1e:11, Dst: c0:8e:00:11:1e:11
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 192.168.238.238
    <snipped>
    Time to live: 255
    Protocol: PIM (103)
    <snipped>
    Source: 192.168.100.101
    Destination: 192.168.238.238
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0001 = Type: Register (1)
    <snipped>
    PIM Options
        Flags: 0x00000000
            0... .... .... .... .... .... .... .... = Border: No
            .0.. .... .... .... .... .... .... .... = Null-Register: No
        0100 .... = IP Version: IPv4 (4)
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
    <snipped>
    Time to live: 253
    Protocol: UDP (17)
    <snipped>
    Source: 192.168.100.101
    Destination: 238.0.0.10
```

```
User Datagram Protocol, Src Port: 58128, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: 50:00:00:01:00:0a), Dst: 01:00:5e:00:00:01
Internet Protocol Version 4, Src: 172.16.10.1, Dst: 224.0.0.1
    <snipped>
    Time to live: 1
    Protocol: IGMP (2)
    <snipped>
    Source: 172.16.10.1
    Destination: 224.0.0.1
    <snipped>
Internet Group Management Protocol
    [IGMP Version: 2]
    Type: Membership Query (0x11)
    Max Resp Time: 10.0 sec (0x64)
    Checksum: 0xee9b [correct]
    [Checksum Status: Good]
    Multicast Address: 0.0.0.0
```

Capture 19-2: *IGMP Membership Query for L2VNI10000 message sent by Leaf-101*

Figures from 19-8 to 19-11 illustrates the Source-Specific Tree join process. Leaf-101 sends an *IGMP Membership Query* message. The original message is double encapsulated with VXLAN tunnel header and Multicast tunnel header by Leaf-101 and then it sends the message down to RP Spine-11. Spine-11 creates (S, G) entry in its MRIB. Capture 19-2 shows the original message.

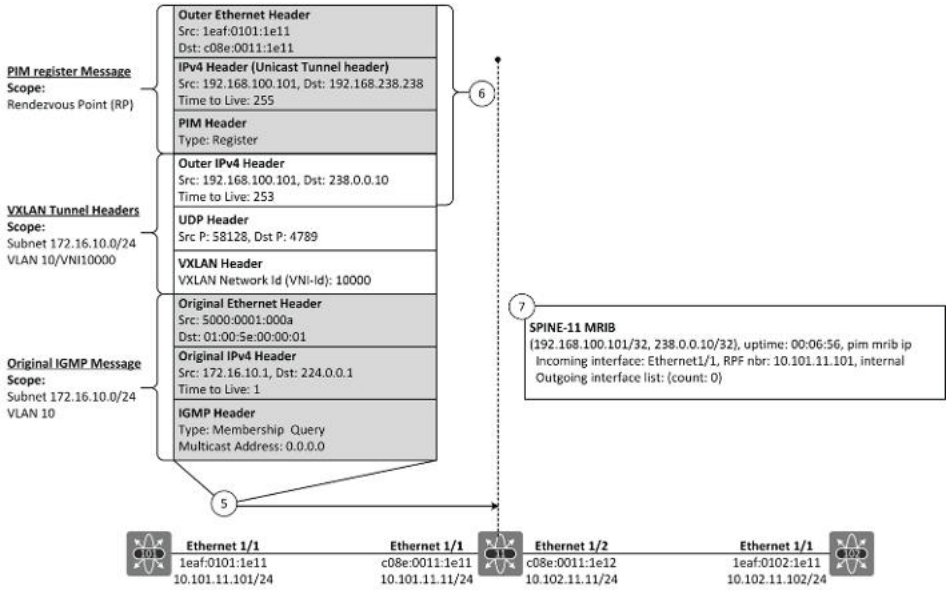


Figure 19-8: *IGMP Membership Query/PIM-Register originated by Leaf-101.*

According to standard PIM-Register/ Register-Stop procedure, Spine-11 sends the PIM Register-Stop message to Leaf-101 as a response to the PIM-Register message. The Register-Stop message is also used as PIM-Join message to Source-Specific Tree. When Leaf-101 receives the message, it adds ingress interface E1/1 into Source-Specific group OIL.

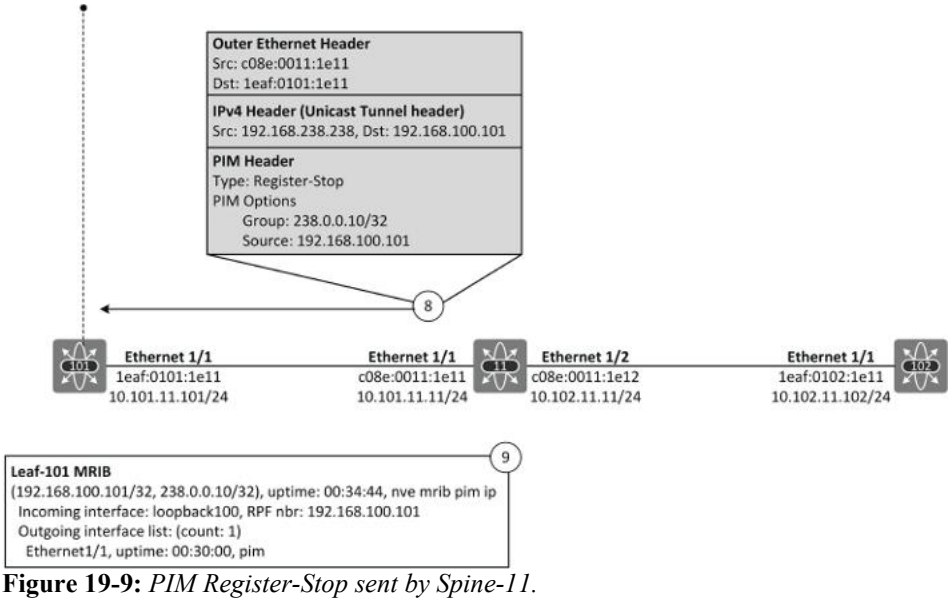


Figure 19-9: *PIM Register-Stop sent by Spine-11.*

The capture 19-3 shows the PIM Register-Stop message sent by Spine-11.

```
Ethernet II, Src: c0:8e:00:11:1e:11, Dst: 1e:af:01:01:1e:11
Internet Protocol Version 4, Src: 192.168.238.238, Dst: 192.168.100.101
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0010 = Type: Register-stop (2)
    Reserved byte(s): 00
    Checksum: 0xc8c6 [correct]
    [Checksum Status: Good]
    PIM Options
        Group: 238.0.0.10/32
        Source: 192.168.100.101
```

Capture 19-3: *PIM Register-Stop sent by Leaf-101.*

In addition, Spine-11 forwards the original IGMP Membership Query message down to previously formed Shared Tree (*. 238.0.0.10) to Leaf-102 without a PIM header.

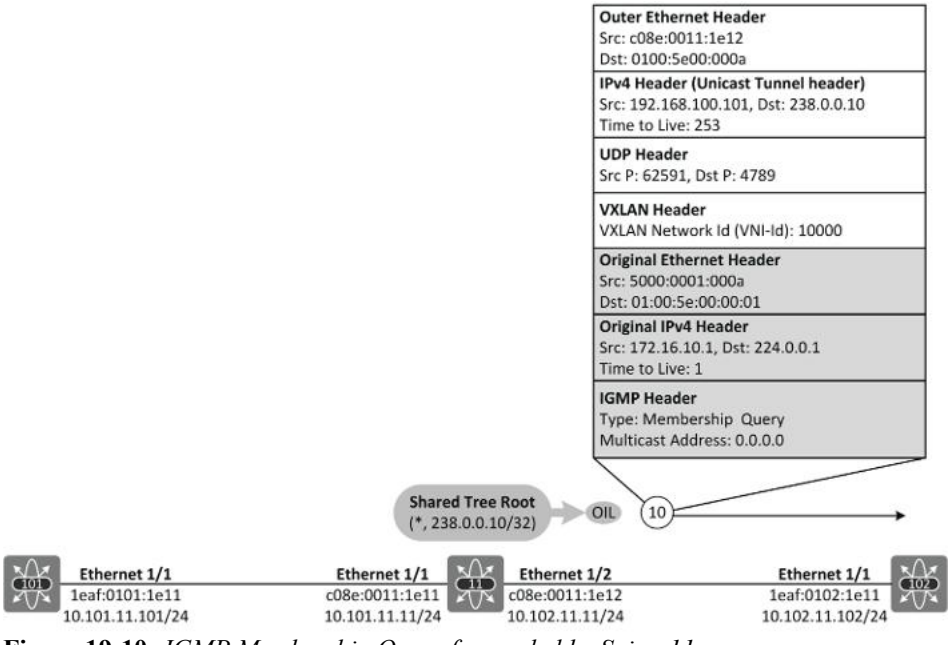


Figure 19-10: *IGMP Membership Query forwarded by Spine-11.*

Capture 19-4 shows the forwarded message.

```
Ethernet II, Src: c0:8e:00:11:1e:12, Dst: 01:00:5e:00:00:0a
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.0.0.10
    <snipped>
    Time to live: 253
    Protocol: UDP (17)
    <snipped>
    Source: 192.168.100.101
    Destination: 238.0.0.10
User Datagram Protocol, Src Port: 62591, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10000
    Reserved: 0
Ethernet II, Src: 50:00:00:01:00:0a, Dst: 01:00:5e:00:00:01
Internet Protocol Version 4, Src: 172.16.10.1, Dst: 224.0.0.1
    <snipped>
    Time to live: 1
    Protocol: IGMP (2)
    <snipped>
    Source: 172.16.10.1
    Destination: 224.0.0.1
    Options: (4 bytes), Router Alert
Internet Group Management Protocol
    [IGMP Version: 2]
    Type: Membership Query (0x11)
    Max Resp Time: 10.0 sec (0x64)
    Checksum: 0xee9b [correct]
    [Checksum Status: Good]
```

Capture 19-4: *IGMP Membership Query forwarded to Leaf-102 by Spine-11*

When Leaf-102 receives the packet, it learns the Source-Specific Tree information from the Src/Dst fields in the IP header. It starts its own join process by sending a PIM Join message to 224.0.0.13 (All-PIM-Routers). It also creates an MRIB entry for the group (192.168.100.101/32, 238.0.0.10) where the interface E1/1 is added into IIL and interface NVE1 is added into OIL. When the PIM Join message reaches Spine-11, it updates the MRIB entry by adding interface E1/2 to OIL of group (192.168.100.101/32, 238.0.0.10/32). It does not forward PIM Join message any further, not because it is an RP, but because it already has Source-Specific Multicast group entry in its MRIB.

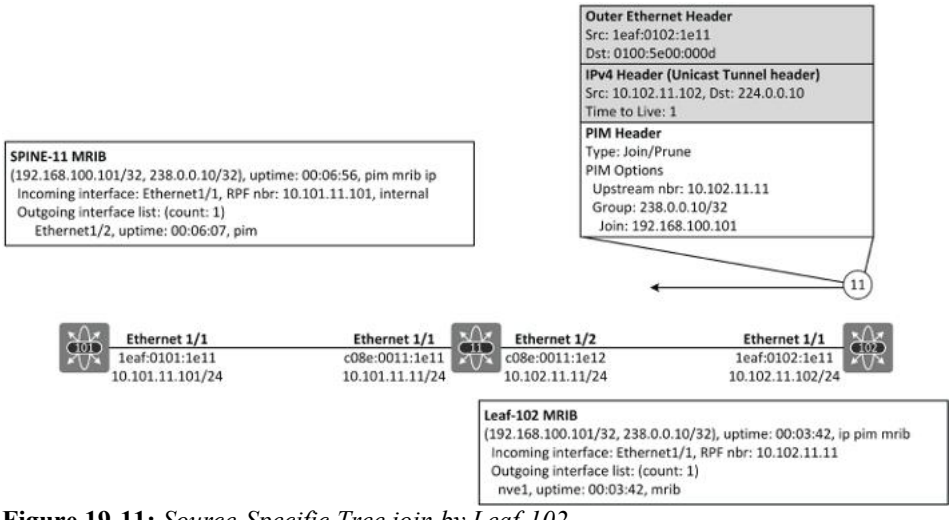


Figure 19-11: *Source-Specific Tree join by Leaf-102.*

Capture 19-5 shows the PIM-Join message originated by Leaf-102.

```
Ethernet II, Src: 1e:af:01:02:1e:11 Dst: 01:00:5e:00:00:0d
Internet Protocol Version 4, Src: 10.102.11.102, Dst: 224.0.0.13
    <snipped>
    Time to live: 1
    Protocol: PIM (103)
    <snipped>
    Source: 10.102.11.102
    Destination: 224.0.0.13
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0011 = Type: Join/Prune (3)
    Reserved byte(s): 00
    Checksum: 0xac61 [correct]
    [Checksum Status: Good]
    PIM Options
        Upstream-neighbor: 10.102.11.11
        Reserved byte(s): 00
        Num Groups: 1
        Holdtime: 210
        Group 0: 238.0.0.10/32
            Num Joins: 1
                IP address: 192.168.100.101/32 (S)
            Num Prunes: 0
```

Capture 19-5: *PIM-Join message sent by leaf-102.*

At this phase, all switches have Shared Tree entry (*, 238.0.0.10) and Source-Specific tree entry (192.168.100.101/32, 238.0.0.10/32) in their MRIB. The same process is executed also by Leaf-102 and as a result, all switches learn also Source-Specific group information (192.168.100.102/32, 238.0.0.10/32).

Tenant Routed Multicast (TRM) Configuration

As its names describe, "*Tenant Routed Multicast (TRM)"* enables Tenant/VRF specific Inter-VN Multicast routing in BGP EVPN VXLAN Fabric. This section introduces the configuration part.

Define Anycast-RP

There is three solutions for TRM Rendezvous Point (RP); Anycast-RP in each leaf switch, external RP outside VXLAN fabric and combination of these two - RP Anywhere. The first option, Anycast-RP is used throughout the remaining sections.

The same anycast-RP IP address is configured in each leaf switch. There is a dedicated interface Loopback 200 with IP address 192.168.200.1/32 for this purpose. This Loopback IP address is used as a tenant/VRF specific RP for the whole Multicast address range under the vrf context configuration section. Note that the interface is attached to vrf and PIM-SM is enabled in it. However, the IP address is not advertised by Underlay Network routing protocol.

```
interface loopback200
  description ** Internal RP **
  vrf member TENANT77
  ip address 192.168.200.1/32
  ip pim sparse-mode
!
vrf context TENANT77
  ip pim rp-address 192.168.200.1 group-list 224.0.0.0/4
```

Example 19-3: *Loopback for Distributed-RP of TENANT77.*

Enable TRM on leaf switches

Enable feature "next-generation MVPN". This enables BGP MVPN address-family and its sub-commands. Enable IGMP snooping for VLANs used in VXLAN fabric. Note that command "***ip multicast overlay-spt-only***" that gratuitously generates the "*Source Active Auto-Discovery Route*" (MVPN Route-Type 5) is enabled by default on Nexus 9000 switches when BGP MVPN afi Control Plane is implemented.

```
feature ngmvpn
ip igmp snooping vxlan
ip multicast overlay-spt-only
```

Example 19-4: *Enabling TRM on leaf switches.*

Define the tenant-based Multicast group for Multicast traffic.

Define the Multicast Group for TENANT77 under Interface NVE1. This will be the Multicast tunnel which aggregates all TENANT77 Multicast flows. This Multicast group address should be different than the Underlay Network Multicast Group for L2BUM.

```
interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  member vni 10000
    mcast-group 238.0.0.10
  member vni 10077 associate-vrf
    mcast-group 238.101.102.103
  member vni 20000
    mcast-group 238.0.0.10
```

Example 19-5: *Enabling TRM on Leaf-103.*

Prevent PIM neighbor establishment within a specific VLAN

In case that there is a PIM enabled routers connected into some of the VLANs, define the route-map that prevents PIM-peering with VLAN SVI. Note that in order to route-map to work, it has to point to prefix-list.

```
interface Vlan10
  no shutdown
  vrf member TENANT77
  ip address 172.16.10.1/24
  ip pim sparse-mode
  ip pim neighbor-policy PREVENT-PIM-NEIGHBOR
  fabric forwarding mode anycast-gateway
```

Example 19-6: *Enabling TRM on Leaf-103.*

BGP afi IPv4 MVPN peering (Leaf)

Enable BGP address-family "*IPv4 MVPN*" towards the spine switch. Enable auto-generated Route-Targets for MVPN export/import policy under vrf context configuration.

```
router bgp 65000
  router-id 192.168.77.103
  address-family ipv4 unicast
  address-family ipv4 mvpn
  address-family l2vpn evpn
  neighbor 192.168.77.11
    remote-as 65000
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family ipv4 mvpn
      send-community extended
    address-family l2vpn evpn
      send-community extended
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 20000 l2
    rd auto
    route-target import auto
    route-target export auto
vrf context TENANT77
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto mvpn
    route-target both auto evpn
```

Example 19-7: *Enabling IPv4 MVPN address-family on Leaf-103.*

BGP afi IPv4 MVPN peering (Spine)

Example 19-8 illustrates the Spine-11 BGP MVPN configuration. BGP MVPN is enabled per neighbor basis. Spine-11 is also a BGP MVPN route-reflector for leaf switches.

```
router bgp 65000
  router-id 192.168.77.11
  address-family ipv4 unicast
  address-family ipv4 mvpn
  address-family l2vpn evpn
  neighbor 192.168.77.101
    remote-as 65000
    update-source loopback77
    address-family ipv4 mvpn
      send-community extended
```

```
      route-reflector-client
    address-family l2vpn evpn
      send-community
      send-community extended
      route-reflector-client
  neighbor 192.168.77.102
    remote-as 65000
    update-source loopback77
    address-family ipv4 mvpn
      send-community extended
      route-reflector-client
    address-family l2vpn evpn
      send-community
      send-community extended
      route-reflector-client
  neighbor 192.168.77.103
    remote-as 65000
    update-source loopback77
    address-family ipv4 mvpn
      send-community extended
      route-reflector-client
    address-family l2vpn evpn
      send-community
      send-community extended
      route-reflector-client
```

Example 19-8: *Enabling BGP MVPN Afi on Spine-11.*

Tenant Routed Multicast (TRM) operation

Inter-VN TRM uses Shared and Source-Specific Multicast delivery trees just like Intra-VN L2VNI solution. This section explains the process of how switches establish these trees.

Shared/Source-Specific tree for Inter-VN

The process is started by Leaf-101. It sends a PIM Register message to the RP Spine-11 by using the default vrf RP IP address 192.168.238.238 as a destination IP in the outer IP header. The inner IP header carries the Source-Specific group (S, G) information. Leaf-101 creates both Shared and Source-Specific entries in its MRIB. It adds the interface E1/1 towards RP into IIL and interfaces NVE1 into OIL of Shared group (*, G) because Shared tree is rooted by Spine-11. The Source-Specific tree is formed in opposite direction because it is rooted by Leaf-101 itself. When Spine-11 receives the PIM-Register message, it first updates its MRIB entries and then sends a PIM Register-Stop message back to Leaf-101 just like in Intra-VN process.

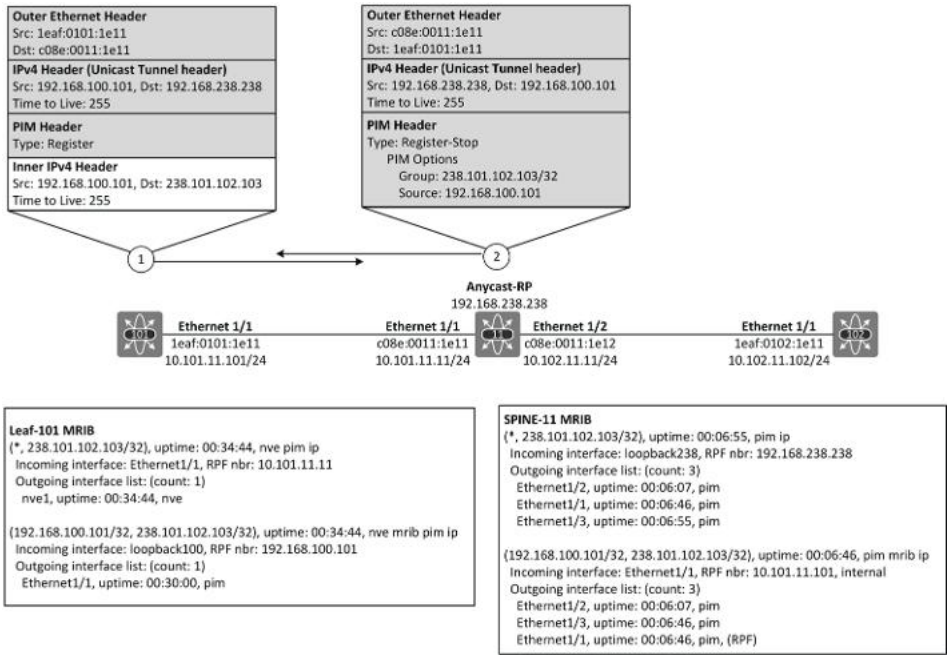


Figure 19-12: *The process of building* a *Source-Specific Tree.*

Capture 19-6 shows the PIM-Register message sent by Leaf-101.

```
Ethernet II, Src: 1e:af:01:01:1e:11, Dst: c0:8e:00:11:1e:11
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 192.168.238.238
    <snipped>
    Time to live: 255
    Protocol: PIM (103)
    <snipped>
    Source: 192.168.100.101
    Destination: 192.168.238.238
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0001 = Type: Register (1)
    Reserved byte(s): 00
    Checksum: 0x9eff [correct]
    [Checksum Status: Good]
    PIM Options
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.101.102.103
    <snipped>
    Time to live: 255
    Protocol: PIM (103)
    <snipped>
    Source: 192.168.100.101
    Destination: 238.101.102.103
```

Capture 19-6: *PIM-Register message sent by leaf-101.*

Capture 19-7 shows the PIM Register-Stop message sent by Spine-11. This process follows the same principle as joining the L2 Source-Specific Tree explained previously. As a result, both switches have Source-Specific entry (192.168.100.101/32, 238.101.102.103/32) in their MRIB. This process also applies to Spine-11 and Leaf-102. These delivery trees are later used for tunneling/aggregating all tenant-specific L3 Multicast traffic. Note that Leaf-101 does not learn Source-Specific tree information originated by Leaf-102.

```
Ethernet II, Src: c0:8e:00:11:1e:11, Dst: 1e:af:01:01:1e:11
Internet Protocol Version 4, Src: 192.168.238.238, Dst: 192.168.100.101
    <snipped>
    Time to live: 255
    Protocol: PIM (103)
    <snipped>
    Source: 192.168.238.238
    Destination: 192.168.100.101
Protocol Independent Multicast
    0010 .... = Version: 2
    .... 0010 = Type: Register-stop (2)
    <snipped>
    PIM Options
        Group: 238.101.102.103/32
        Source: 192.168.100.101
```

Capture 19-7: *PIM Register-Stop message sent by Spine-11.*

Verification

Example 19-9 shows the MRIB of Leaf-101 concerning TRM L3 Multicast Group 238.101.102.103. The RPF-source for the Shared tree (*, G) is 192.168.238.238 which is the Underlay Network Anycast-RP (Spine-11). The VXLAN Flags describes that the Multicast frames received from a shared tree are VXLAN encapsulated.

The RPF-Source for the Source-Specific tree (S,G) is Leaf-101 itself with IP address of interface NVE1 which is also listed in IIL while the interface E1/1 (Uplink to Spine-11) is in OIL. VXLAN field in the source-specific tree shows that Multicast flows for this group will be sent with VXLAN encapsulation. In addition, the output shows that Leaf-101 is received the Register-Stop message from Spine-11. At this phase there hasn't been any Multicast data sent or received for either group and that is why the status is marked as "Inactive Flow".

```
Leaf-101# sh ip mroute 238.101.102.103 detail
(*, 238.101.102.103/32), uptime: 00:41:27, nve(1) ip(0) pim(0)
  RPF Change only
  RPF-Source: 192.168.238.238 [41/110]
  RD-RT ext comm Route-Import:
  Data Created: No
  VXLAN Flags
    VXLAN Encap
    VXLAN Last Hop
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: Ethernet1/1, RPF nbr: 10.101.11.11
  Outgoing interface list: (count: 1) (bridge-only: 0)
    nve1, uptime: 00:41:27, nve


(192.168.100.101/32, 238.101.102.103/32), uptime: 00:41:27, nve(0) mrib(0)
ip(0)
 pim(1)
```

```
  RPF-Source: 192.168.100.101 [0/0]
  RD-RT ext comm Route-Import:
  Data Created: No
  Received Register stop
  VXLAN Flags
    VXLAN Encap
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: loopback100, RPF nbr: 192.168.100.101
  Outgoing interface list: (count: 1) (bridge-only: 0)
    Ethernet1/1, uptime: 00:34:02, pim
```

Example 19-9: *Partial MRIB output on Leaf-101.*

Example 19-10 shows the MRIB of Spine-11 concerning TRM L3 Multicast Group 238.101.102.103.

```
Spine-11# sh ip mroute 238.101.102.103 detail
IP Multicast Routing Table for VRF "default"

Total number of routes: 9
Total number of (*,G) routes: 2
Total number of (S,G) routes: 6
Total number of (*,G-prefix) routes: 1

(*, 238.101.102.103/32), uptime: 01:21:31, pim(3) ip(0)
  RPF-Source: 192.168.238.238 [0/0]
  Data Created: No
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: loopback238, RPF nbr: 192.168.238.238
  Outgoing interface list: (count: 2) (bridge-only: 0)
    Ethernet1/1, uptime: 01:12:45, pim
    Ethernet1/2, uptime: 01:21:31, pim


(192.168.100.101/32, 238.101.102.103/32), uptime: 01:12:01, pim(3) mrib(0)
ip(0)

  RPF-Source: 192.168.100.101 [41/110]
  Data Created: No
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: Ethernet1/1, RPF nbr: 10.101.11.101, internal
  Outgoing interface list: (count: 2) (bridge-only: 0)
    Ethernet1/1, uptime: 01:10:01, pim, (RPF)
    Ethernet1/2, uptime: 01:12:01, pim


(192.168.100.102/32, 238.101.102.103/32), uptime: 01:21:17, pim(3) mrib(0)
ip(0)

  RPF-Source: 192.168.100.102 [41/110]
  Data Created: No
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: Ethernet1/2, RPF nbr: 10.102.11.102, internal
    Outgoing interface list: (count: 2) (bridge-only: 0)
      Ethernet1/2, uptime: 01:12:16, pim, (RPF)
      Ethernet1/1, uptime: 01:12:45, pim
```

Example 19-10: *Partial MRIB output on Spine-11.*

TRM Control Plane operation.

This section describes the Control Plane operation. It starts by explaining how the local host informs its upstream leaf switch about its willingness to receive Multicast flow from some specific flow. Next, this section introduces the usage of BGP MVPN "*Source Active Auto-Discovery Route (SA A-D)" [MVPN Route-type 5]* in the situation when a leaf switch starts receiving a Multicast flow from one of its locally connected host. The last part discusses the operation of the BGP MVPN when remote leaf receives the BGP update message carrying SA A-D NLRI.

IGMP membership report

Host Bebe in figure 19-13 wants to receive the Multicast traffic from group 239.77.77.77. It sends an IGMP Membership Report by using group-specific destination MAC address. When Leaf-102 receives the IGMP Membership Report, it creates a Shared Tree entry in its VRF specific MRIB. Leaf-102 install VLAN 20 into OIL and sets itself as a root for the tree. Because there is no sender for this flow, its state is marked as "Inactive Flow".

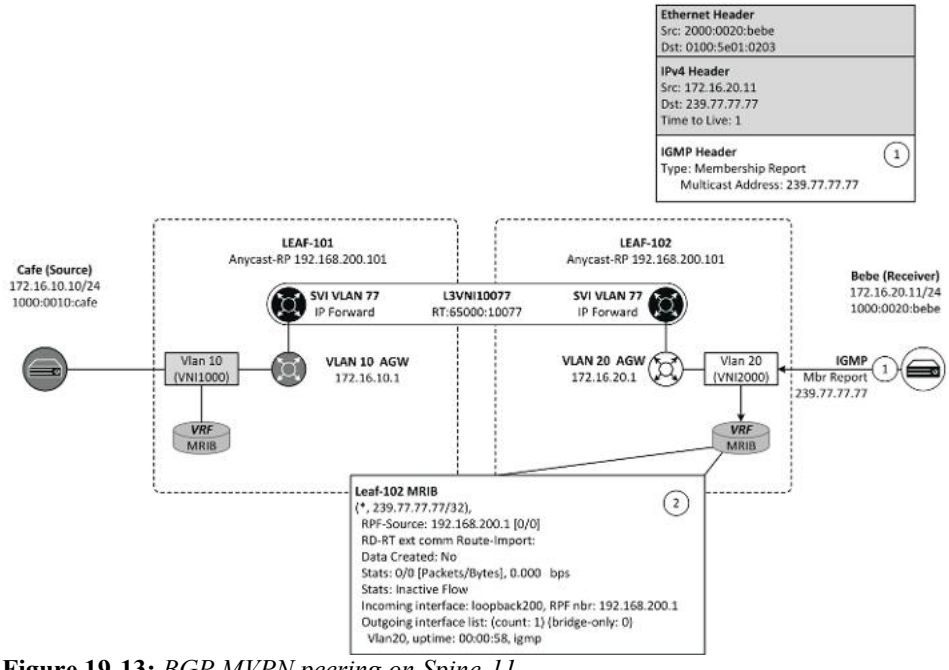


Figure 19-13: *BGP MVPN peering on Spine-11.*

Example 19-11 shows the MRIB of vrf context TENANT on Leaf-102. The shared group (*, 239.77.77.77) is created based on the IGMP Membership report received from VLAN 20.

```
Leaf-102# sh ip mroute 239.77.77.77 detail vrf TENANT77
(*, 239.77.77.77/32), uptime: 00:01:14, igmp(1) ip(0) pim(0)
  RPF-Source: 192.168.200.1 [0/0]
  RD-RT ext comm Route-Import:
  Data Created: No
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: loopback200, RPF nbr: 192.168.200.1
  Outgoing interface list: (count: 1) (bridge-only: 0)
    Vlan20, uptime: 00:01:14, igmp
```

Example 19-11: *Tenant based MRIB entry on Leaf-102.*

MVPN Source-Active Auto-Discovery

Leaf-101 starts receiving Multicast flow to group 239.77.77.77 from host Cafe. Leaf-101 creates Source-Specific (172.16.10.10/32, 239.77.77.77) group entry in TENANT77 MRIB (example 19-12). IT adds VLAN 10 into Incoming Interface List while the Outgoing Interface List remains empty at this phase. Next, Leaf-101 creates a BGP IPv4 MVPN "*Source Active Auto-Discovery Route"* Route-Type 5 in its BGP Loc-RIB, from where it is sent to Spine-11 via Adj-RIB-Out. Leaf-101 checks if there are any interested receivers for Multicast flow by sending this SA A-D route. When Leaf-102 receives the SA A-D route, it installs it into Adj-RIB-In based on Route-Target 65000-10077, which is used in all leaf switches for vrf TENANT77 L3. From there, it is installed into Loc-RIB. The group information carried within BGP update is then installed into vrf TENANT77 specific MRIB.

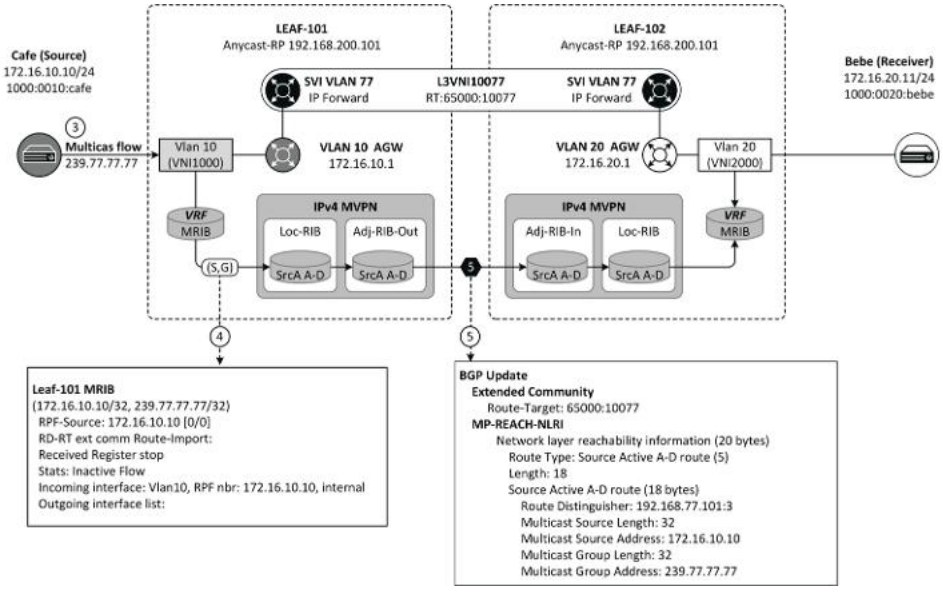


Figure 19-14: *Source-Active Auto-Discovery by Leaf-101.*

Example 19-12 shows the MRIB of Leaf-101. At this phase, the flow is still marked as "inactive" and there are no interfaces added into OIL.

```
Leaf-101# sh ip mroute 239.77.77.77 detail vrf TENANT77
IP Multicast Routing Table for VRF "TENANT77"

Total number of routes: 2
Total number of (*,G) routes: 0
Total number of (S,G) routes: 1
Total number of (*,G-prefix) routes: 1

(172.16.10.10/32, 239.77.77.77/32), uptime: 00:00:46, ip(0) pim(0) ngmvpn(1)
  RPF-Source: 172.16.10.10 [0/0]
  RD-RT ext comm Route-Import:
  Data Created: Yes
  Received Register stop
  Fabric dont age route
  Stats: 2/200 [Packets/Bytes], 34.783  bps
  Stats: Inactive Flow
  Incoming interface: Vlan10, RPF nbr: 172.16.10.10, internal
  interface list: (count: 0) (Fabric OIF 0) (bridge-only: 0)
```

Example 19-12: *Tenant based MRIB entry on Leaf-101.*

Example 19-13 shows the Source Active Auto-Discovery route in the BGP table of Leaf-101. Route-Distinguisher is formed based on BGP Router-Id and vrf-Id (3) of Leaf-101. The actual address includes information about:

[5]: Route-Type 5,
[172.16.10.10]: Real Source/Sender IP address
[239.77.77.77] Multicast group.

```
Leaf-101# sh bgp ipv4 mvpn route-type 5 detail
BGP routing table information for VRF default, address family IPv4 MVPN
Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
BGP routing table entry for [5][172.16.10.10][239.77.77.77]/64, version 7
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in mvpn

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (192.168.77.101)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: RT:65000:10077

  Path-id 1 advertised to peers:
    192.168.77.11
```

Example 19-13: *MVPN entry in Leaf-101 BGP table.*

Capture 19-8 shows the BGP Update forwarded by Spine-11.

```
Ethernet II, Src: c0:8e:00:11:1e:12 (c0:8e:00:11:1e:12), Dst: 1e:af:01:02:1e:11
(1e:af:01:02:1e:11)
Internet Protocol Version 4, Src: 192.168.77.11, Dst: 192.168.77.102
Transmission Control Protocol, Src Port: 51182, Dst Port: 179, Seq: 20, Ack: 1,
Len: 95
Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 95
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 72
    Path attributes
        Path Attribute - ORIGIN: IGP
        Path Attribute - AS_PATH: empty
        Path Attribute - LOCAL_PREF: 100
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 8
            Carried extended communities: (1 community)
                Route Target: 65000:10077
        Path Attribute - ORIGINATOR_ID: 192.168.77.101
            Flags: 0x80, Optional, Non-transitive, Complete
            Type Code: ORIGINATOR_ID (9)
            Length: 4
            Originator identifier: 192.168.77.101
        Path Attribute - CLUSTER_LIST: 192.168.77.11
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length,
            Type Code: MP_REACH_NLRI (14)
            Length: 29
            Address family identifier (AFI): IPv4 (1)
            Subsequent address family identifier (SAFI): MCAST-VPN (5)
            Next hop network address (4 bytes)
                Next Hop: 192.168.100.101
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (20 bytes)
                Route Type: Source Active A-D route (5)
                Length: 18
                Source Active A-D route (18 bytes)
                    Route Distinguisher: 192.168.77.101:3
                    Multicast Source Length: 32
                    Multicast Source Address: 172.16.10.10
                    Multicast Group Length: 32
                    Multicast Group Address: 239.77.77.77
```

Capture 19-8: *Source Active Auto-Discovery Route sent by Leaf-101.*

Example 19-14 shows the received BGP Update in Adj-RIB-In and BGP Loc-RIB of Leaf-102. Note that as per normal import process, receiving switch will change the global administrator from RD to its own IP.

```
Leaf-102# sh bgp ipv4 mvpn route-type 5 detail
BGP routing table information for VRF default, address family IPv4 MVPN
Route Distinguisher: 192.168.77.101:3
BGP routing table entry for [5][172.16.10.10][239.77.77.77]/64, version 7
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in mvpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
             Imported paths list: default
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65000:10077
      Originator: 192.168.77.101 Cluster list: 192.168.77.11

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.102:3    (L3VNI 10077)
BGP routing table entry for [5][172.16.10.10][239.77.77.77]/64, version 8
Paths: (1 available, best #1)
Flags: (0x00001a) (high32 00000000) on xmit-list, is in mvpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop, in rib
             Imported from 192.168.77.101:3:[5][172.16.10.10][239.77.77.77]/64
  AS-Path: NONE, path sourced internal to AS
    192.168.100.101 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:65000:10077
      Originator: 192.168.77.101 Cluster list: 192.168.77.11
  Path-id 1 not advertised to any peer
```

Example 19-14: *MVPN entry in Leaf-102 BGP table.*

When Leaf-102 receives the BGP Update from Spine-11, it imports route based on the Route-target 65000:10077, which is used for L3VNI. Leaf-102 creates Source-Specific Multicast group entry in its MRIB using the information carried within the BGP update. Then it adds VLAN20, which already is installed into OIL of shared group (*, 239.77.77.77), into OIL of Source-Specific group (172.16.10.10/32, 239.77.77.77/32). The Leaf-102 generates the BGP IPv4 MVPN *"Source-Tree Join Route [MVPN Route-Type 7]"* and send it via Spine-11 to Leaf-101. By doing this, it informs the Leaf-101 that it has at least one local host waiting for Multicast flow to group 239.77.77.77. Leaf-101 receives the BGP Update and imports it NLRI into Adj-RIB-in and from there into BGP Loc-RIB. Then it updates the Source-Specific group (172.16.10.10/32, 239.77.77.77) entry by adding L3 routing interface (SVI) into OIL.

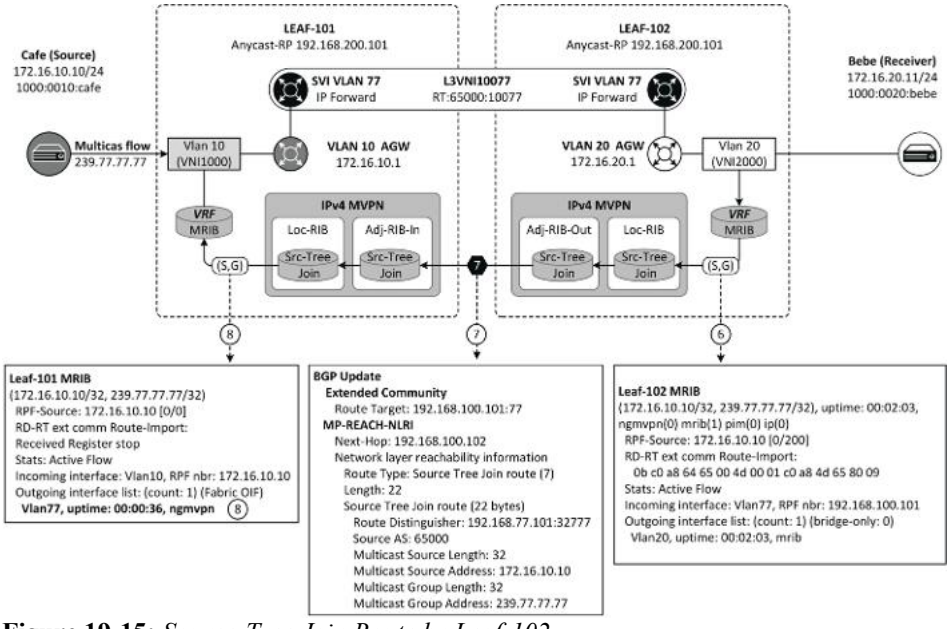


Figure 19-15: *Source-Tree Join Route by Leaf-102.*

Example 19-15 shows the vrf TENANT MRIB entries concerning Multicast destination 239.77.77.77. THE "*RD-RT Extended Community Route-Import*" field is expressed in HEX format. The HEX portion can be divided into two entities, Route-Target and Route-Distinguisher.

c0 a8 64 65 00 4d (RT):
192.168.100.101 and 00 4d = 77 => RT: 192.168.100.101:77

c0 a8 4d 65 80 98 (RD)
192.168.77.101 and 80 09 = 32777 => RD: 192.168.77.101:32777
This information is taken from the Source Active Auto-Discover route sent by Leaf-101.
The status of the group (172.16.10.10/32, 238.77.77.77/32) is now marked as "Active Flow".

```
Leaf-102# sh ip mroute 239.77.77.77 detail vrf TENANT77
IP Multicast Routing Table for VRF "TENANT77"

Total number of routes: 3
Total number of (*,G) routes: 1
Total number of (S,G) routes: 1
Total number of (*,G-prefix) routes: 1

(*, 239.77.77.77/32), uptime: 00:03:32, igmp(1) ip(0) pim(0)
  RPF-Source: 192.168.200.1 [0/0]
  RD-RT ext comm Route-Import:
  Data Created: No
  Stats: 0/0 [Packets/Bytes], 0.000   bps
  Stats: Inactive Flow
  Incoming interface: loopback200, RPF nbr: 192.168.200.1
```

```
  Outgoing interface list: (count: 1) (bridge-only: 0)
    Vlan20, uptime: 00:03:32, igmp


(172.16.10.10/32, 239.77.77.77/32), uptime: 00:02:03, ngmvpn(0) mrib(1) pim(0)
ip(0)
  RPF-Source: 172.16.10.10 [0/200]
  RD-RT ext com Route-Import:0b c0 a8 64 65 00 4d 00 01 c0 a8 4d 65 80 09
  Data Created: No
  Fabric dont age route, Fabric Source
  Stats: 4/400 [Packets/Bytes], 26.667  bps
  Stats: Active Flow
  Incoming interface: Vlan77, RPF nbr: 192.168.100.101
  Outgoing interface list: (count: 1) (bridge-only: 0)
    Vlan20, uptime: 00:02:03, mrib
```

Example 19-15: *MRIB entries concerning for 239.77.77.77 on MRIB of Leaf-102.*

As a response to *Source Active Auto-Discovery* and the fact that it has connected receiver, Leaf 102 generates the *Source Tree Join* Route BGP Update and sends it to Spine-11. The message contains the original RD and RT values stored in MRIB. This way only the switch for whom we are answering, in this case to Leaf-101 imports the route into its BGP table and from there into MRIB. In case where Leaf-102 receives another IGMP Join to same group, no matter from which vlan, it does not generate new MVPN Source Tree Join route because it is already receiving the flow.

```
Leaf-102# sh bgp ipv4 mvpn route-type 7 detail
BGP routing table information for VRF default, address family IPv4 MVPN
Route Distinguisher: 192.168.77.101:32777    (Local VNI: 10077)
BGP routing table entry for [7][172.16.10.10][239.77.77.77][65000]/96, version
9
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in mvpn

  Advertised path-id 1
  Path type: local, path is valid, is best path, no labeled nexthop
  AS-Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (192.168.77.102)
      Origin IGP, MED not set, localpref 100, weight 32768
      Extcommunity: RT:192.168.100.101:77

  Path-id 1 advertised to peers:
    192.168.77.11
```

Example 19-16: *Source Tree Join Route generated by Leaf-102.*

Capture 19-9 shows the *Source Tree Join Route.*

```
Ethernet II, Src: 1e:af:01:02:1e:11 (1e:af:01:02:1e:11), Dst: c0:8e:00:11:1e:12
(c0:8e:00:11:1e:12)
Internet Protocol Version 4, Src: 192.168.77.102, Dst: 192.168.77.11
Transmission Control Protocol, Src Port: 179, Dst Port: 51182, Seq: 1, Ack:
115, Len: 85
Border Gateway Protocol - UPDATE Message
    <snipped>
    Path attributes
        <snipped>
        Path Attribute - EXTENDED_COMMUNITIES
            Flags: 0xc0, Optional, Transitive, Complete
            Type Code: EXTENDED_COMMUNITIES (16)
            Length: 8
            Carried extended communities: (1 community)
```

```
                Route Target: 192.168.100.101:77
        Path Attribute - MP_REACH_NLRI
            Flags: 0x90, Optional, Extended-Length,
            Type Code: MP_REACH_NLRI (14)
            Length: 33
            Address family identifier (AFI): IPv4 (1)
            Subsequent address family identifier (SAFI): MCAST-VPN (5)
            Next hop network address (4 bytes)
                Next Hop: 192.168.100.102
            Number of Subnetwork points of attachment (SNPA): 0
            Network layer reachability information (24 bytes)
                Route Type: Source Tree Join route (7)
                Length: 22
                Source Tree Join route (22 bytes)
                    Route Distinguisher: 192.168.77.101:32777
                    Source AS: 65000
                    Multicast Source Length: 32
                    Multicast Source Address: 172.16.10.10
                    Multicast Group Length: 32
                    Multicast Group Address: 239.77.77.77
```

Capture 19-9: *Source Tree Join Route sent by Leaf-102.*

Example 19-17 shows that Leaf-101 has installed route into its BGP table under L3VNI.

```
Leaf-101# sh bgp ipv4 mvpn route-type 7 detail
BGP routing table information for VRF default, address family IPv4 MVPN
Route Distinguisher: 192.168.77.101:3    (L3VNI 10077)
BGP routing table entry for [7][172.16.10.10][239.77.77.77][65000]/96, version
9
Paths: (1 available, best #1)
Flags: (0x00001a) (high32 00000000) on xmit-list, is in mvpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop, in rib
             Imported from
192.168.77.101:32777:[7][172.16.10.10][239.77.77.77][65000]/96
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:192.168.100.101:77
      Originator: 192.168.77.102 Cluster list: 192.168.77.11

  Path-id 1 not advertised to any peer

Route Distinguisher: 192.168.77.101:32777
BGP routing table entry for [7][172.16.10.10][239.77.77.77][65000]/96, version
8
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in mvpn, is not in HW

  Advertised path-id 1
  Path type: internal, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
             Imported paths list: default
  AS-Path: NONE, path sourced internal to AS
    192.168.100.102 (metric 81) from 192.168.77.11 (192.168.77.11)
      Origin IGP, MED not set, localpref 100, weight 0
      Extcommunity: RT:192.168.100.101:77
      Originator: 192.168.77.102 Cluster list: 192.168.77.11

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

Example 19-17: *Source Tree Join Route received by Leaf-101.*

Example 19-18 verifies that the vrf TENANT77 MRIB on Leaf-101 is updated and the L3VNI SVI 77 is added into the OIL list. Now the Control Plane process is ready.

```
Leaf-101# sh ip mroute 239.77.77.77 det vrf TENANT77
IP Multicast Routing Table for VRF "TENANT77"

Total number of routes: 2
Total number of (*,G) routes: 0
Total number of (S,G) routes: 1
Total number of (*,G-prefix) routes: 1

(172.16.10.10/32, 239.77.77.77/32), uptime: 00:00:46, ip(0) pim(0) ngmvpn(1)
  RPF-Source: 172.16.10.10 [0/0]
  RD-RT ext comm Route-Import:
  Data Created: Yes
  Received Register stop
  Fabric dont age route
  Stats: 2/200 [Packets/Bytes], 34.783  bps
  Stats: Active Flow
  Incoming interface: Vlan10, RPF nbr: 172.16.10.10, internal
  Outgoing interface list: (count: 1) (Fabric OIF) (bridge-only: 0)
    Vlan77, uptime: 00:00:46, ngmvpn (fabric)
```

Example 19-18: *MRIB of Leaf-101.*

Data Plane Operation

Figure 19-16 illustrates the TRM Data Plane operation. The Multicast flow is simulated by sending an ICMP Query (ping) to 238.77.77.77 from the host Cafe (VLAN10). Host Bebe, with IP address 172.16.20.11/24 in VLAN 20 has joined to group 238.77.77.77. This example assumes that both hosts' IP/MAC information has already known by both leaf switches.

Ingress leaf operation

Leaf-101 receives Multicast flow to group 239.77.77.77 from the host Cafe. Leaf-101 forwards stream based on the MRIB of vrf TENANT77, which states that packets belonging to the group (172.16.10.10/32, 238.77.77.77/32) received from VLAN10, will be forwarded out of the vrf specific L3VNI interface SVI77 with VXLAN encapsulation. The VN-Id in VXLAN header is set to 10077. The outer IP header destination is set to vrf TENANT77 specific L3 Multicast group 238.102.103.104 while the source IP address is set to 192.168.100.101 (IP address of interface NVE1). The destination Multicast group address is used as a Multicast tunnel for all of the TENANT77 Inter-VN Multicast flows.

Spine operation

When Spine-11 receives the Multicast packets, it makes the forwarding decision based on the outer IP header source-destination IP addresses. It has Source-Specific tree entry (192.168.100.101/32, 238.101.102.103/32) in its default MRIB, which states that packets belonging to this group will be sent out from interfaces Ethernet 1/2.

Egress leaf operation

Leaf-102 receives the Multicast flow. Based on the destination UDP port 4789, it knows that flow specific packets are VXLAN encapsulated. After it has verified to which vrf packet belongs to by checking the VNI-Id from VXLAN header, it removes tunnel headers and makes forwarding decisions based on original IP header. Leaf-102 check the vrf TENANT77 MRIB and forward packets out of the VLAN20.

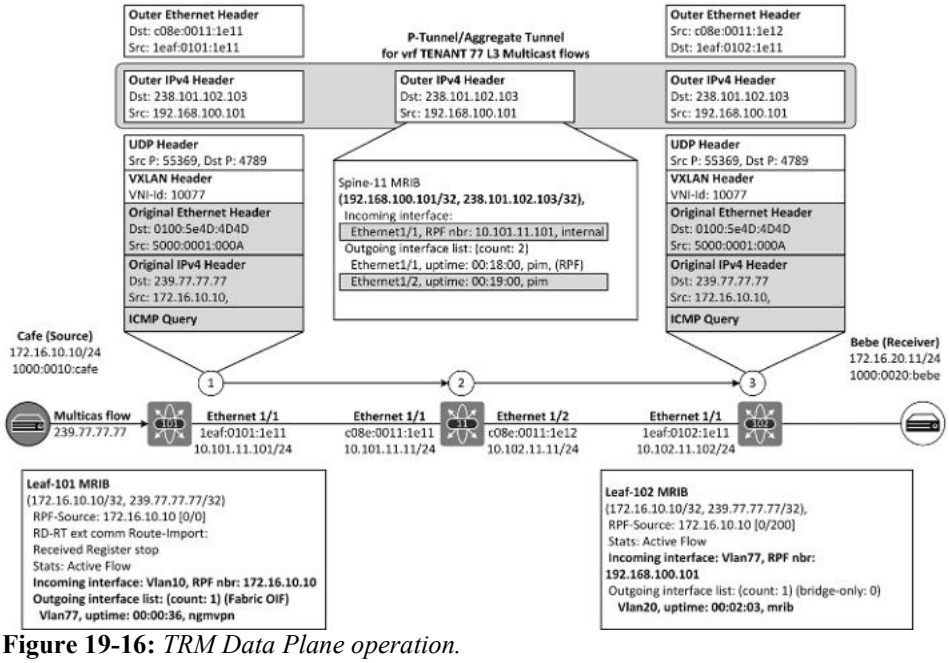


Figure 19-16: *TRM Data Plane operation.*

Capture 19-10 shows the captured Multicast packet. The outer IP header destination address is Multicast tunnel address. The inner destination IP address is the address of the destination for the Multicast flow.

```
Ethernet II, Src: c0:8e:00:11:1e:12, Dst: 01:00:5e:65:66:67
Internet Protocol Version 4, Src: 192.168.100.101, Dst: 238.101.102.103
User Datagram Protocol, Src Port: 55369, Dst Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 10077
    Reserved: 0
Ethernet II, Src: 50:00:00:01:00:07, Dst: 01:00:5e:4d:4d:4d
Internet Protocol Version 4, Src: 172.16.10.10, Dst: 239.77.77.77
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xa95b [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    Data (72 bytes)
```

Capture 19-10: *Multicast packet capture.*

Summary

This chapter introduces the Tenant Routed Multicast (TRM) operation from the Underlay Network and Overlay Network perspective. It also explains the BGP MVPN address-family and introduces two MVPN NLRI; *"Source Active Auto-Discovery route"* (MVPN Route-Type 7) and *"Source Tree Join route"* (MVPN Route-Type 7). In addition, this chapter discusses the TRM Data Plane operation.

References

[RFC4610] D. Farinassi and Y. Cai., “Anycast-RP using Using Protocol Independent Multicast (PIM)”, RFC4610, August 2006.

[RFC 4721] Y. Rekhter et al., “A Border Gateway Protocol 4 (BGP-4)”, RFC4271, January 2006

[RFC 4760] T. bates et al., “Multiprotocol Extensions for BGP-4”, RFC4760, January 2007.

[RFC6513] E. Rosen et al., “Multicast in MPLS/BGP IP VPNs”, RFC 6513, February 2012.

[RFC6514] R. Aggarwal et al., “BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs”, RFC 6514, February 2012.

[RFC7432] A. Sajassi et al., “BGP MPLS-Based Ethernet VPN”, RFC7432, February 2015.

[RFC7761] B. Fenner et al., “Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised), March 2016.

[EVPN-MVPN] B. Fenner et al., 2Seamless Multicast Interoperability between EVPN and MVPN PEs” draft-sajassi-bess-evpn-mvpn-seamless-interop-04:, July 2019.

Appendix A.

Configuration files for BGW switches and DC Core switch.

BGW-1 Configuration

```
BGW-1# sh run

!Command: show running-config
!Running configuration last done at: Wed Aug  7 09:19:25 2019
!Time: Wed Aug  7 09:21:43 2019

version 9.2(3) Bios:version
hostname BGW-1
vdc BGW-1 id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource vrf minimum 2 maximum 4096
  limit-resource port-channel minimum 0 maximum 511
  limit-resource u4route-mem minimum 248 maximum 248
  limit-resource u6route-mem minimum 96 maximum 96
  limit-resource m4route-mem minimum 58 maximum 58
  limit-resource m6route-mem minimum 8 maximum 8

nv overlay evpn
feature ospf
feature bgp
feature pim
feature fabric forwarding
feature interface-vlan
feature vn-segment-vlan-based
feature lacp
feature nv overlay

username admin password 5
$5$YTfyrnCx$D0BEzwcJJWm/PRjj/ykdkAySBr/9B6dsou/NWEAm6D
4  role network-admin
ip domain-lookup
copp profile strict
evpn multisite border-gateway 12
  delay-restore time 300
snmp-server user admin network-admin auth md5
0x42cd35684f49b26fca133253a1e0519d
 priv 0x42cd35684f49b26fca133253a1e0519d localizedkey
rmon event 1 description FATAL(1) owner PMON@FATAL
rmon event 2 description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 description ERROR(3) owner PMON@ERROR
rmon event 4 description WARNING(4) owner PMON@WARNING
rmon event 5 description INFORMATION(5) owner PMON@INFO

fabric forwarding anycast-gateway-mac 0001.0001.0001
ip pim rp-address 192.168.238.1 group-list 238.0.0.0/24 bidir
ip pim ssm range 232.0.0.0/8
vlan 1,10,30,40,50,77
vlan 10
  name L2VNI-for-VLAN10
  vn-segment 10000
vlan 30
  vn-segment 30000
vlan 40
```

```
  vn-segment 40000
vlan 50
  vn-segment 50000
vlan 77
  name TENANT77
  vn-segment 10077

route-map REDIST-TO-SITE-EXT-DCI permit 10
  match tag 1234
vrf context TENANT77
  vni 10077
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
vrf context management
hardware access-list tcam region racl 512
hardware access-list tcam region vpc-convergence 256
hardware access-list tcam region arp-ether 256 double-wide


interface Vlan1

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  multisite border-gateway interface loopback88
  member vni 10000
    multisite ingress-replication
    mcast-group 238.0.0.10
  member vni 10077 associate-vrf
  member vni 30000
    mcast-group 238.0.0.10
  member vni 40000
    mcast-group 238.0.0.10
  member vni 50000
    mcast-group 238.0.0.10

interface Ethernet1/1
  description **Fabric Internal **
  no switchport
  mac-address b063.0001.1e11
  medium p2p
  ip address 10.1.11.1/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  evpn multisite fabric-tracking
  no shutdown

interface Ethernet1/2
  description ** DCI Interface **
  no switchport
  mac-address b063.0001.1e12
  medium p2p
  ip address 10.1.88.1/24 tag 1234
  ip pim sparse-mode
  evpn multisite dci-tracking
  no shutdown

interface Ethernet1/3
  description **Fabric Internal **
```

```
  no switchport
  mac-address b063.0001.1e13
  medium p2p
  ip address 10.11.1.1/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet1/4
  description ** DCI Interface **
  no switchport
  mac-address b063.0001.1e14
  medium p2p
  ip address 10.88.1.1/24 tag 1234
  no shutdown

interface mgmt0
  vrf member management

interface loopback0
  description ** RID/Underlay **
  ip address 192.168.0.1/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode

interface loopback77
  description ** BGP peering **
  ip address 192.168.77.1/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode

interface loopback88
  description ** VIP for DCI-Inter-connect **
  ip address 192.168.88.12/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface loopback100
  description ** VTEP/Overlay **
  ip address 192.168.100.1/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
line console
line vty
boot nxos bootflash:/nxos.9.2.3.bin
router ospf UNDERLAY-NET
  router-id 192.168.0.1
router bgp 65012
  router-id 192.168.77.1
  no enforce-first-as
  address-family ipv4 unicast
    redistribute direct route-map REDIST-TO-SITE-EXT-DCI
  address-family l2vpn evpn
  neighbor 10.1.88.88
    remote-as 65088
    update-source Ethernet1/2
    address-family ipv4 unicast
  neighbor 192.168.77.11
    remote-as 65012
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
```

```
  neighbor 192.168.77.88
    remote-as 65088
    update-source loopback77
    ebgp-multihop 5
    peer-type fabric-external
    address-family l2vpn evpn
      send-community
      send-community extended
      rewrite-evpn-rt-asn
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 30000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 40000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 50000 l2
    rd auto
    route-target import auto
    route-target export auto
```

BGW-2 Configuration

```
BGW-2# sh run

!Command: show running-config
!Running configuration last done at: Wed Aug  7 09:19:31 2019
!Time: Wed Aug  7 09:24:10 2019

version 9.2(3) Bios:version
hostname BGW-2
vdc BGW-2 id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource vrf minimum 2 maximum 4096
  limit-resource port-channel minimum 0 maximum 511
  limit-resource u4route-mem minimum 248 maximum 248
  limit-resource u6route-mem minimum 96 maximum 96
  limit-resource m4route-mem minimum 58 maximum 58
  limit-resource m6route-mem minimum 8 maximum 8

nv overlay evpn
feature ospf
feature bgp
feature pim
feature fabric forwarding
feature interface-vlan
feature vn-segment-vlan-based
feature lacp
feature nv overlay

username admin password 5
$5$6O5Ozded$6G9z9ZYJnto10KgJSqYou0dZilxI2abRLQOgpBTzu8
A  role network-admin
ip domain-lookup
```

```
copp profile strict
evpn multisite border-gateway 12
  delay-restore time 300
snmp-server user admin network-admin auth md5
0x9bcc18427d4176f2aec8419a200a8bbf
 priv 0x9bcc18427d4176f2aec8419a200a8bbf localizedkey
rmon event 1 description FATAL(1) owner PMON@FATAL
rmon event 2 description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 description ERROR(3) owner PMON@ERROR
rmon event 4 description WARNING(4) owner PMON@WARNING
rmon event 5 description INFORMATION(5) owner PMON@INFO

fabric forwarding anycast-gateway-mac 0001.0001.0001
ip pim rp-address 192.168.238.1 group-list 238.0.0.0/24 bidir
ip pim ssm range 232.0.0.0/8
vlan 1,10,30,40,50,77
vlan 10
  name L2VNI-for-VLAN10
  vn-segment 10000
vlan 30
  vn-segment 30000
vlan 40
  vn-segment 40000
vlan 50
  vn-segment 50000
vlan 77
  name TENANT77
  vn-segment 10077

route-map REDIST-TO-SITE-EXT-DCI permit 10
  match tag 1234
vrf context TENANT77
  vni 10077
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
vrf context management
hardware access-list tcam region racl 512
hardware access-list tcam region vpc-convergence 256
hardware access-list tcam region arp-ether 256 double-wide


interface Vlan1

interface nve1
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  multisite border-gateway interface loopback88
  member vni 10000
    multisite ingress-replication
    mcast-group 238.0.0.10
  member vni 10077 associate-vrf
  member vni 30000
    mcast-group 238.0.0.10
  member vni 40000
    mcast-group 238.0.0.10
  member vni 50000
    mcast-group 238.0.0.10

interface Ethernet1/1
  description **Fabric Internal **
```

```
  no switchport
  mac-address b063.0002.1e11
  medium p2p
  ip address 10.2.11.2/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  evpn multisite fabric-tracking
  no shutdown

interface Ethernet1/2
  description ** DCI Interface **
  no switchport
  mac-address b063.0002.1e12
  medium p2p
  ip address 10.2.88.2/24 tag 1234
  ip ospf network point-to-point
  ip pim sparse-mode
  evpn multisite dci-tracking
  no shutdown

interface Ethernet1/3
  description **Fabric Internal **
  no switchport
  mac-address b063.0002.1e13
  medium p2p
  ip address 10.11.2.2/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet1/4
  description ** DCI Interface **
  no switchport
  mac-address b063.0002.1e14
  medium p2p
  ip address 10.88.2.2/24 tag 1234
  no shutdown
interface mgmt0
  vrf member management

interface loopback0
  description ** RID/Underlay **
  ip address 192.168.0.2/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode

interface loopback77
  description ** BGP peering **
  ip address 192.168.77.2/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode

interface loopback88
  description ** VIP for DCI-Inter-connect **
  ip address 192.168.88.12/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface loopback100
  description ** VTEP/Overlay **
  ip address 192.168.100.2/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
```

```
  ip pim sparse-mode
line console
line vty
boot nxos bootflash:/nxos.9.2.3.bin
router ospf UNDERLAY-NET
  router-id 192.168.0.2
router bgp 65012
  router-id 192.168.77.2
  no enforce-first-as
  address-family ipv4 unicast
    redistribute direct route-map REDIST-TO-SITE-EXT-DCI
  address-family l2vpn evpn
  neighbor 10.2.88.88
    remote-as 65088
    update-source Ethernet1/2
    peer-type fabric-external
    address-family ipv4 unicast
  neighbor 192.168.77.11
    remote-as 65012
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  neighbor 192.168.77.88
    remote-as 65088
    update-source loopback77
    ebgp-multihop 5
    peer-type fabric-external
    address-family l2vpn evpn
      send-community
      send-community extended
      rewrite-evpn-rt-asn
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 30000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 40000 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 50000 l2
    rd auto
    route-target import auto
    route-target export auto
```

BGW-3 Configuration

```
BGW-3# sh run

!Command: show running-config
!No configuration change since last restart
!Time: Wed Aug  7 09:36:25 2019

version 9.2(3) Bios:version
hostname BGW-3
```

```
vdc BGW-3 id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource vrf minimum 2 maximum 4096
  limit-resource port-channel minimum 0 maximum 511
  limit-resource u4route-mem minimum 248 maximum 248
  limit-resource u6route-mem minimum 96 maximum 96
  limit-resource m4route-mem minimum 58 maximum 58
  limit-resource m6route-mem minimum 8 maximum 8

nv overlay evpn
feature ospf
feature bgp
feature pim
feature fabric forwarding
feature interface-vlan
feature vn-segment-vlan-based
feature lacp
feature nv overlay

username admin password 5
$5$O9jHouJ4$gMMf.hMYXJRamUNys17VtdztzLMNq1PdMQDIc1xPZu
9  role network-admin
ip domain-lookup
copp profile strict
evpn multisite border-gateway 12
  delay-restore time 300
snmp-server user admin network-admin auth md5
0x423cb9002003f0f3c3acb917bba00bf8
 priv 0x423cb9002003f0f3c3acb917bba00bf8 localizedkey
rmon event 1 description FATAL(1) owner PMON@FATAL
rmon event 2 description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 description ERROR(3) owner PMON@ERROR
rmon event 4 description WARNING(4) owner PMON@WARNING
rmon event 5 description INFORMATION(5) owner PMON@INFO

fabric forwarding anycast-gateway-mac 0001.0001.0001
ip pim rp-address 192.168.238.1 group-list 238.0.0.0/24 bidir
ip pim ssm range 232.0.0.0/8
vlan 1,10,77
vlan 10
  name L2VNI-for-VLAN10
  vn-segment 10000
vlan 77
  name TENANT77
  vn-segment 10077

route-map REDIST-TO-SITE-EXT-DCI permit 10
  match tag 1234
vrf context TENANT77
  vni 10077
  rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
vrf context management
hardware access-list tcam region racl 512
hardware access-list tcam region vpc-convergence 256
hardware access-list tcam region arp-ether 256 double-wide


interface Vlan1

interface nve1
```

```
  no shutdown
  host-reachability protocol bgp
  source-interface loopback100
  multisite border-gateway interface loopback88
  member vni 10000
    multisite ingress-replication
    mcast-group 238.0.0.10
  member vni 10077 associate-vrf

interface Ethernet1/1
  description **Fabric Internal **
  no switchport
  mac-address b063.0003.1e11
  medium p2p
  ip address 10.3.12.3/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  evpn multisite fabric-tracking
  no shutdown

interface Ethernet1/2
  description ** DCI Interface **
  no switchport
  mac-address b063.0003.1e12
  medium p2p
  ip address 10.3.88.3/24 tag 1234
  evpn multisite dci-tracking
  no shutdown

interface Ethernet1/3
  description **Fabric Internal **
  no switchport
  mac-address b063.0003.1e13
  medium p2p
  ip address 10.12.3.3/24
  ip ospf network point-to-point
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface Ethernet1/4
  description ** DCI Interface **
  no switchport
  mac-address b063.0003.1e14
  medium p2p
  ip address 10.88.3.3/24 tag 1234
  no shutdown
interface mgmt0
  vrf member management

interface loopback0
  description ** RID/Underlay **
  ip address 192.168.0.3/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode

interface loopback77
  description ** BGP peering **
  ip address 192.168.77.3/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface loopback88
```

```
  description ** VIP for DCI-Inter-connect **
  ip address 192.168.88.34/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0

interface loopback100
  description ** VTEP/Overlay **
  ip address 192.168.100.3/32 tag 1234
  ip router ospf UNDERLAY-NET area 0.0.0.0
  ip pim sparse-mode
line console
line vty
boot nxos bootflash:/nxos.9.2.3.bin
router ospf UNDERLAY-NET
  router-id 192.168.0.3
router bgp 65034
  router-id 192.168.77.3
  no enforce-first-as
  address-family ipv4 unicast
    redistribute direct route-map REDIST-TO-SITE-EXT-DCI
    maximum-paths 5
    maximum-paths ibgp 5
  address-family l2vpn evpn
  neighbor 10.3.88.88
    remote-as 65088
    update-source Ethernet1/2
    address-family ipv4 unicast
  neighbor 10.88.3.88
    remote-as 65088
    update-source Ethernet1/4
    address-family ipv4 unicast
  neighbor 192.168.77.12
    remote-as 65034
    description ** Spine-11 BGP-RR **
    update-source loopback77
    address-family l2vpn evpn
      send-community extended
  neighbor 192.168.77.88
    remote-as 65088
    update-source loopback77
    ebgp-multihop 5
    peer-type fabric-external
    address-family l2vpn evpn
      send-community
      send-community extended
      rewrite-evpn-rt-asn
  vrf TENANT77
    address-family ipv4 unicast
      advertise l2vpn evpn
evpn
  vni 10000 l2
    rd auto
    route-target import auto
    route-target export auto
```

DC Core switch (RouteServer) Configuration

```
RouteServer-1# sh run

!Command: show running-config
!No configuration change since last restart
!Time: Wed Aug  7 09:38:18 2019

version 9.2(3) Bios:version
```

```
hostname RouteServer-1
vdc RouteServer-1 id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource vrf minimum 2 maximum 4096
  limit-resource port-channel minimum 0 maximum 511
  limit-resource u4route-mem minimum 128 maximum 128
  limit-resource u6route-mem minimum 96 maximum 96
  limit-resource m4route-mem minimum 58 maximum 58
  limit-resource m6route-mem minimum 8 maximum 8

nv overlay evpn
feature bgp
feature nv overlay

username admin password 5
$5$SAAwN66P$OSzsu5lztjirsP.UM0bkhSXhjkAqAnymcN0jNUwNc3
8  role network-admin
ip domain-lookup
copp profile strict
snmp-server user admin network-admin auth md5
0x842c130e837d0182abbfc3c8010e25f1
 priv 0x842c130e837d0182abbfc3c8010e25f1 localizedkey
rmon event 1 description FATAL(1) owner PMON@FATAL
rmon event 2 description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 description ERROR(3) owner PMON@ERROR
rmon event 4 description WARNING(4) owner PMON@WARNING
rmon event 5 description INFORMATION(5) owner PMON@INFO

vlan 1

route-map REDIST-TO-SITE-EXT-DCI permit 10
  match tag 1234
route-map RETAIN-NEXT-HOP permit 10
  set ip next-hop unchanged
vrf context abba
  address-family ipv4 unicast
    route-target import 65088:1
    route-target export 65088:1
    route-target both auto
vrf context beef
  address-family ipv4 unicast
    route-target import 65088:2
    route-target export 65088:2
vrf context management
hardware access-list tcam region racl 512
hardware access-list tcam region vpc-convergence 256
hardware access-list tcam region arp-ether 256 double-wide


interface Ethernet1/1
  description ** to BGW-1 **
  no switchport
  ip address 10.1.88.88/24
  no shutdown

interface Ethernet1/2
  description ** to BGW-2 **
  no switchport
  ip address 10.2.88.88/24
  no shutdown

interface Ethernet1/3
  description ** to BGW-3 **
```

```
  no switchport
  ip address 10.3.88.88/24
  no shutdown

interface Ethernet1/4
  description ** to BGW-4 **
  no switchport
  ip address 10.4.88.88/24
  no shutdown

interface mgmt0
  vrf member management

interface loopback77
  ip address 192.168.77.88/32 tag 1234

interface loopback88
  ip address 192.168.88.88/32 tag 1234
line console
line vty
boot nxos bootflash:/nxos.9.2.3.bin
router bgp 65088
  router-id 192.168.77.88
  address-family ipv4 unicast
    redistribute direct route-map REDIST-TO-SITE-EXT-DCI
    maximum-paths 2
  address-family l2vpn evpn
    maximum-paths 2
    maximum-paths ibgp 2
    retain route-target all
  template peer MULTI-SITE-OVERLAY-PEERING
    update-source loopback77
    ebgp-multihop 5
    address-family l2vpn evpn
      send-community
      send-community extended
      route-map RETAIN-NEXT-HOP out
  neighbor 10.1.88.1
    remote-as 65012
    address-family ipv4 unicast
  neighbor 10.2.88.2
    remote-as 65012
    address-family ipv4 unicast
  neighbor 10.3.88.3
    remote-as 65034
    address-family ipv4 unicast
  neighbor 10.4.88.4
    remote-as 65034
    address-family ipv4 unicast
  neighbor 192.168.77.1
    inherit peer MULTI-SITE-OVERLAY-PEERING
    remote-as 65012
    address-family l2vpn evpn
      rewrite-evpn-rt-asn
  neighbor 192.168.77.2
    inherit peer MULTI-SITE-OVERLAY-PEERING
    remote-as 65012
    address-family l2vpn evpn
      rewrite-evpn-rt-asn
  neighbor 192.168.77.3
    inherit peer MULTI-SITE-OVERLAY-PEERING
    remote-as 65034
    address-family l2vpn evpn
```

```
      rewrite-evpn-rt-asn
  neighbor 192.168.77.4
    inherit peer MULTI-SITE-OVERLAY-PEERING
    remote-as 65034
    address-family l2vpn evpn
      rewrite-evpn-rt-asn
```