Cisco Nexus 3600 NX-OS VXLAN Configuration Guide, Release 9.2(x)

First Published: 2018-07-18

Last Modified: 2018-11-05

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

© 2018 Cisco Systems, Inc. All rights reserved.

CONTENTS

PREFACE

Preface vii

Audience vii

Document Conventions vii

Related Documentation for Cisco Nexus 3600 Platform Switches viii

Documentation Feedback viii

Communications, Services, and Additional Information viii

CHAPTER 1

New and Changed Information 1

New and Changed Information 1

CHAPTER 2

Overview 3

Licensing Requirements 3

Supported Platforms 3

CHAPTER 3

Configuring VXLANs 5

Overview 5

VXLAN Overview 5

VXLAN Encapsulation and Packet Format 6

VXLAN Tunnel Endpoints 6

VXLAN Packet Forwarding Flow 7

ECMP and LACP Load Sharing with VXLANs 7

Advertising Primary IP Address 7

Guidelines and Limitations for VXLANs 8

Considerations for VXLAN Deployment 9

Enabling a VXLAN 9

Mapping a VLAN to a VXLAN VNI 10

Configuring a Routing Protocol for NVE Unicast Addresses 10
Creating and Configuring an NVE Interface 11
Configuring a VXLAN VTEP in vPC 12
Configuring Replication for a VNI 14
Configuring Multicast Replication 14
Configuring IGMP Snooping Over VXLAN 15
Overview of IGMP Snooping Over VXLAN 15
Guidelines and Limitations for IGMP Snooping Over VXLAN 15
Configuring IGMP Snooping Over VXLAN 15
Verifying the VXLAN Configuration 15

CHAPTER 4 **Configuring VXLAN BGP EVPN 19**
Information About VXLAN BGP EVPN 19
Guidelines and Limitations for VXLAN BGP EVPN 19
Considerations for VXLAN BGP EVPN Deployment 20
Network Considerations for VXLAN Deployments 20
Considerations for the Transport Network 21
BGP EVPN Considerations for VXLAN Deployment 22
Configuring VXLAN BGP EVPN 22
Enabling VXLAN 22
Configuring VLAN and VXLAN VNI 22
Configuring VRF for VXLAN Routing 22
Configuring SVI for Hosts for VXLAN Routing 23
Configuring VRF Overlay VLAN for VXLAN Routing 23
Configuring VNI Under VRF for VXLAN Routing 24
Configuring Anycast Gateway for VXLAN Routing 24
Configuring the NVE Interface and VNIs 24
Configuring BGP on the VTEP 25
Configuring RD and Route Targets for VXLAN Bridging 26
Configuring BGP for EVPN on the Spine 26
Disabling VXLANs 27
Duplicate Detection for IP and MAC Addresses 28
Verifying the VXLAN Configuration 29
Example of VXLAN BGP EVPN (EBGP) 30

Example of VXLAN BGP EVPN (IBGP) 41
Example Show Commands 49

APPENDIX A **DHCP Relay in VXLAN BGP EVPN 53**
DHCP Relay in VXLAN BGP EVPN Overview 53
Guidelines and Limitations for DHCP Relay 54
DHCP Relay in VXLAN BGP EVPN Example 54
Basic VXLAN BGP EVPN Configuration 55
DHCP Relay on VTEPs 59
Client on Tenant VRF and Server on Layer 3 Default VRF 59
Client on Tenant VRF (SVI X) and Server on the Same Tenant VRF (SVI Y) 63
Client on Tenant VRF (VRF X) and Server on Different Tenant VRF (VRF Y) 66
Client on Tenant VRF and Server on Non-Default Non-VXLAN VRF 69
Configuring VPC Peers Example 71
vPC VTEP DHCP Relay Configuration Example 73

Preface

This preface includes the following sections:

- Audience, on page vii
- Document Conventions, on page vii
- Related Documentation for Cisco Nexus 3600 Platform Switches, on page viii
- Documentation Feedback, on page viii
- Communications, Services, and Additional Information, on page viii

Audience

This publication is for network administrators who install, configure, and maintain Cisco Nexus switches.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Convention	Description
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

Convention	Description
`screen font`	Terminal sessions and information the switch displays are in screen font.
`boldface screen font`	Information you must enter is in boldface screen font.
italic screen font	Arguments for which you supply values are in italic screen font.
< >	Nonprinting characters, such as passwords, are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Related Documentation for Cisco Nexus 3600 Platform Switches

The entire Cisco Nexus 3600 platform switch documentation set is available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-3000-series-switches/tsd-products-support-series-home.html

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus3k-docfeedback@cisco.com. We appreciate your feedback.

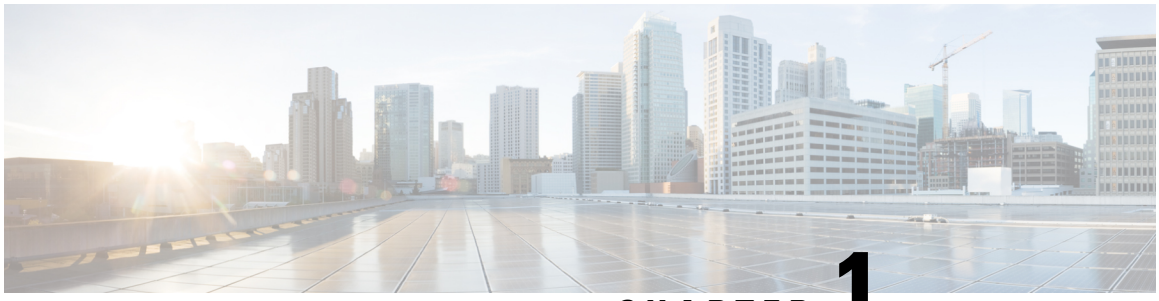
Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Preface



CHAPTER 1

New and Changed Information

This chapter provides release-specific information for each new and changed feature in the *Cisco Nexus 3600 NX-OS VXLAN Configuration Guide*.

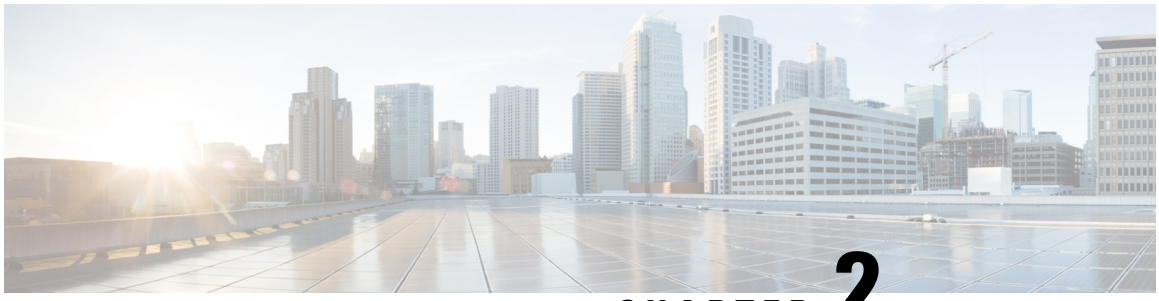
- New and Changed Information, on page 1

New and Changed Information

Table 1: New and Changed Features

Feature	Description	Changed in Release	Where Documented
	Changed the document title from 9.x to 9.2(x)		Title page
VXLAN DHCP IPv4 Relay	Support added for Cisco Nexus 36180YC-R and 3636C-R switches.	9.2(2)	DHCP Relay in VXLAN BGP EVPN, on page 53
CLI Simplification	Support added for the reduction of CLI commands.	9.2(1)	Example of VXLAN BGP EVPN (EBGP), on page 30 Example of VXLAN BGP EVPN (IBGP), on page 41
VXLAN with vPC	Added support for 3600 platforms.	9.2(1)	Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide. Release 9.x

New and Changed Information



CHAPTER 2

Overview

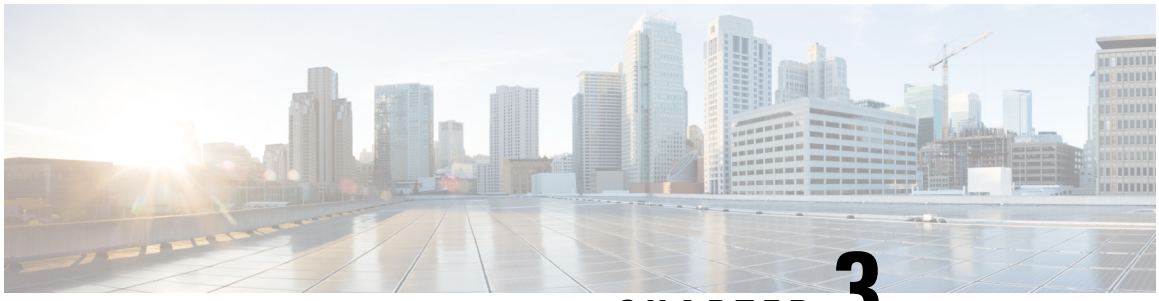
- Licensing Requirements, on page 3
- Supported Platforms, on page 3

Licensing Requirements

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the *Cisco NX-OS Licensing Guide* and the *Cisco NX-OS Licensing Options Guide*.

Supported Platforms

Starting with Cisco NX-OS release 7.0(3)I7(1), use the Nexus Switch Platform Support Matrix to know from which Cisco NX-OS releases various Cisco Nexus 9000 and 3000 switches support a selected feature.



CHAPTER 3

Configuring VXLANs

This chapter contains the following sections:

- Overview, on page 5
- ECMP and LACP Load Sharing with VXLANs, on page 7
- Advertising Primary IP Address, on page 7
- Guidelines and Limitations for VXLANs, on page 8
- Considerations for VXLAN Deployment, on page 9
- Enabling a VXLAN, on page 9
- Mapping a VLAN to a VXLAN VNI, on page 10
- Configuring a Routing Protocol for NVE Unicast Addresses, on page 10
- Creating and Configuring an NVE Interface, on page 11
- Configuring a VXLAN VTEP in vPC, on page 12
- Configuring Replication for a VNI, on page 14
- Configuring Multicast Replication, on page 14
- Configuring IGMP Snooping Over VXLAN, on page 15
- Verifying the VXLAN Configuration, on page 15

Overview

VXLAN Overview

The Cisco Nexus 3600 platform switches are designed for a hardware-based Virtual Extensible LAN (VXLAN) function. These switches can extend Layer 2 connectivity across the Layer 3 boundary and integrate between VXLAN and non-VXLAN infrastructures. Virtualized and multitenant data center designs can be shared over a common physical infrastructure.

VXLANs enable you to extend Layer 2 networks across the Layer 3 infrastructure by using MAC-in-UDP encapsulation and tunneling. In addition, you can use a VXLAN to build a multitenant data center by decoupling tenant Layer 2 segments from the shared transport network.

When deployed as a VXLAN gateway, the Cisco Nexus 3600 platform switches can connect VXLAN and classic VLAN segments to create a common forwarding domain so that tenant devices can reside in both environments.

A VXLAN has the following benefits:

- Flexible placement of multitenant segments throughout the data center.

  It extends Layer 2 segments over the underlying shared network infrastructure so that tenant workloads can be placed across physical pods in the data center.

- Higher scalability to address more Layer 2 segments.

  A VXLAN uses a 24-bit segment ID called the VXLAN network identifier (VNID). The VNID allows a maximum of 16 million VXLAN segments to coexist in the same administrative domain. (In comparison, traditional VLANs use a 12-bit segment ID that can support a maximum of 4096 VLANs.)

- Utilization of available network paths in the underlying infrastructure.

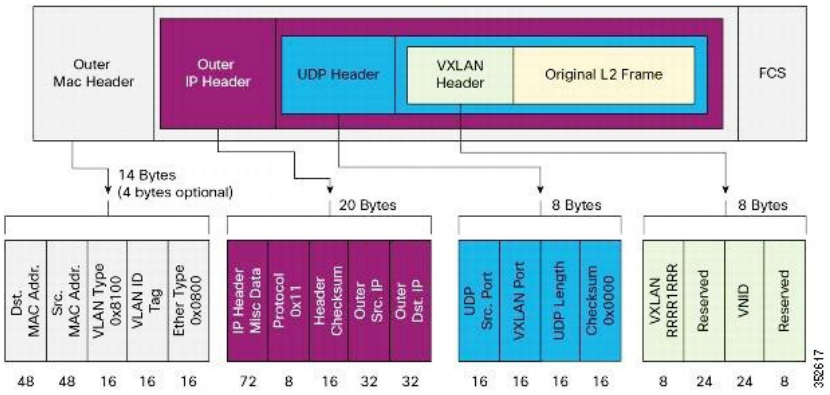
  VXLAN packets are transferred through the underlying network based on its Layer 3 header. It uses equal-cost multipath (ECMP) routing and link aggregation protocols to use all available paths.

VXLAN Encapsulation and Packet Format

A VXLAN is a Layer 2 overlay scheme over a Layer 3 network. It uses MAC-in-UDP encapsulation to extend Layer 2 segments across the data center network. The transport protocol over the physical data center network is IP plus UDP.

A VXLAN defines a MAC-in-UDP encapsulation scheme where the original Layer 2 frame has a VXLAN header added and is then placed in a UDP-IP packet. With this MAC-in-UDP encapsulation, VXLAN tunnels Layer 2 network over the Layer 3 network. The VXLAN packet format is shown in the following figure.

Figure 1: VXLAN Packet Format



A VXLAN uses an 8-byte VXLAN header that consists of a 24-bit VNID and a few reserved bits. The VXLAN header and the original Ethernet frame are in the UDP payload. The 24-bit VNID identifies the Layer 2 segments and maintains Layer 2 isolation between the segments. A VXLAN can support 16 million LAN segments.

VXLAN Tunnel Endpoints

A VXLAN uses VXLAN tunnel endpoint (VTEP) devices to map tenants' end devices to VXLAN segments and to perform VXLAN encapsulation and deencapsulation. Each VTEP device has two types of interfaces:

- Switch port interfaces on the local LAN segment to support local endpoint communication through bridging

- IP interfaces to the transport network where the VXLAN encapsulated frames will be sent

A VTEP device is identified in the IP transport network by using a unique IP address, which is a loopback interface IP address. The VTEP device uses this IP address to encapsulate Ethernet frames and transmits the encapsulated packets to the transport network through the IP interface. A VTEP device learns the remote VTEP IP addresses and the remote MAC address-to-VTEP IP mapping for the VXLAN traffic that it receives.

The VXLAN segments are independent of the underlying network topology; conversely, the underlying IP network between VTEPs is independent of the VXLAN overlay. The IP network routes the encapsulated packets based on the outer IP address header, which has the initiating VTEP as the source IP address and the terminating VTEP or multicast group IP address as the destination IP address.

VXLAN Packet Forwarding Flow

A VXLAN uses stateless tunnels between VTEPs to transmit traffic of the overlay Layer 2 network through the Layer 3 transport network.

ECMP and LACP Load Sharing with VXLANs

Encapsulated VXLAN packets are forwarded between VTEPs based on the native forwarding decisions of the transport network. Most data center transport networks are designed and deployed with multiple redundant paths that take advantage of various multipath load-sharing technologies to distribute traffic loads on all available paths.

A typical VXLAN transport network is an IP-routing network that uses the standard IP equal cost multipath (ECMP) to balance the traffic load among multiple best paths. To avoid out-of-sequence packet forwarding, flow-based ECMP is commonly deployed. An ECMP flow is defined by the source and destination IP addresses and optionally, the source and destination TCP or UDP ports in the IP packet header.

All the VXLAN packet flows between a pair of VTEPs have the same outer source and destination IP addresses, and all VTEP devices must use one identical destination UDP port that can be either the Internet Allocated Numbers Authority (IANA)-allocated UDP port 4789 or a customer-configured port. The only variable element in the ECMP flow definition that can differentiate VXLAN flows from the transport network standpoint is the source UDP port. A similar situation for Link Aggregation Control Protocol (LACP) hashing occurs if the resolved egress interface that is based on the routing and ECMP decision is an LACP port channel. LACP uses the VXLAN outer-packet header for link load-share hashing, which results in the source UDP port being the only element that can uniquely identify a VXLAN flow.

In the Cisco Nexus 3600 platform switches implementation of VXLANs, a hash of the inner frame's header is used as the VXLAN source UDP port. As a result, a VXLAN flow can be unique. The IP address and UDP port combination is in its outer header while the packet traverses the underlay transport network.

Advertising Primary IP Address

On a vPC-enabled leaf or border leaf switch, by default all Layer-3 routes are advertised with the secondary IP address (VIP) of the leaf switch VTEP as the BGP next-hop IP address. Prefix routes and leaf switch generated routes are not synced between vPC leaf switches. Using the VIP as the BGP next-hop for these types of routes can cause traffic to be forwarded to the wrong vPC leaf or border leaf switch and black-holed. The provision to use the primary IP address (PIP) as the next-hop when advertising prefix routes or loopback interface routes in BGP on vPC-enabled leaf or border leaf switches allows users to select the PIP as BGP

next-hop when advertising these types of routes so that traffic will always be forwarded to the right vPC-enabled leaf or border leaf switch.

The configuration command for advertising the PIP is **advertise-pip**.

The following is a sample configuration:

```
switch(config)# router bgp 65536
  address-family 12vpn evpn
    advertise-pip
interface nve 1
    advertise virtual-rmac
```

The **advertise-pip** command lets BGP use the PIP as next-hop when advertising prefix routes or leaf-generated routes if vPC is enabled.

VMAC (virtual-mac) is used with VIP and system MAC is used with PIP when the VIP/PIP feature is enabled.

With the **advertise-pip** and **advertise virtual-rmac** commands enabled, type 5 routes are advertised with PIP and type 2 routes are still advertised with VIP. In addition, VMAC will be used with VIP and system MAC will be used with PIP.

Note The **advertise-pip** and **advertise-virtual-rmac** commands must be enabled and disabled together for this feature to work properly. If you enable or disable one and not the other, it is considered an invalid configuration.

Guidelines and Limitations for VXLANs

VXLAN has the following guidelines and limitations:

- IGMP snooping is supported on VXLAN VLANs.
- VXLAN Layer 2 Gateway functionality is supported.
- VXLAN Flood and Learn functionality is not supported.
- Ensure that the network can accommodate an additional 50 bytes for the VXLAN header.
- Only one Network Virtualization Edge (NVE) interface is supported on a switch.
- Layer 3 VXLAN uplinks are not supported in a nondefault virtual and routing forwarding (VRF) instance.
- Switched Port Analyzer (SPAN) for ports carrying VXLAN-encapsulated traffic is not supported.
- VXLAN with Layer 3 VPN is not supported.
- VXLAN with ingress replication is not supported.
- MLD snooping is not supported on VXLAN VLANs.
- ACLs and QoS policies are not supported on VXLAN VLANs.
- DHCP snooping is not supported on VXLAN VLANs.

Considerations for VXLAN Deployment

The following are some of the considerations while deploying VXLANs:

- A loopback interface IP is used to uniquely identify a VTEP device in the transport network.
- To establish IP multicast routing in the core, an IP multicast configuration, PIM configuration, and Rendezvous Point (RP) configuration are required.
- You can configure VTEP-to-VTEP unicast reachability through any IGP protocol.
- VXLAN multicast traffic should always use the RPT shared tree.
- An RP for the multicast group on the VTEP is a supported configuration. However, you must configure the RP for the multicast group at the spine layer/upstream device. Because all multicast traffic traverses the RP, it is more efficient to have this traffic directed to a spine layer/upstream device.

Enabling a VXLAN

Enabling VXLANs involves the following:

- Enabling the VXLAN feature
- Enabling VLAN to VN-Segment mapping

Before you begin

Ensure that you have installed the VXLAN Enterprise license.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **[no] feature nv overlay**	Enables the VXLAN feature.
Step 3	switch (config)# **[no] feature vn-segment-vlan-based**	Configures the global mode for all VXLAN bridge domains. Enables VLAN to VN-Segment mapping. VLAN to VN-Segment mapping is always one-to-one.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable a VXLAN and configure VLAN to VN-Segment mapping:

```
switch# configure terminal
switch(config)# feature nv overlay
switch(config)# feature vn-segment-vlan-based
switch(config)# copy running-config startup-config
```

Mapping a VLAN to a VXLAN VNI

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vlan** *vlan-id*	Specifies a VLAN.
Step 3	switch(config-vlan)# **vn-segment** *vnid*	Specifies the VXLAN virtual network identifier (VNID). The range of values for vnid is 1 to 16777214.

Example

This example shows how to map a VLAN to a VXLAN VNI:

```
switch# configure terminal
switch(config)# vlan 3100
switch(config-vlan)# vn-segment 5000
```

Configuring a Routing Protocol for NVE Unicast Addresses

Configuring a routing protocol for unicast addresses involves the following:

- Configuring a dedicated loopback interface for NVE reachability.
- Configuring the routing protocol network type.
- Specifying the routing protocol instance and area for an interface.
- Enabling PIM sparse mode in case of multicast replication.

Note Open shortest path first (OSPF) is used as the routing protocol in the examples.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **interface loopback** *instance*	Creates a dedicated loopback interface for the NVE interface. The instance range is from 0 to 1023.
Step 3	switch(config-if)# **ip address** *ip-address/length*	Configures an IP address for this interface.
Step 4	switch(config-if)# **ip ospf network** {**broadcast** | **point-to-point**}	Configures the OSPF network type to a type other than the default for an interface.
Step 5	switch(config-if)# **ip router ospf** *instance-tag* **area** *area-id*	Specifies the OSPF instance and area for an interface.
Step 6	switch(config-if)# **ip pim sparse-mode**	Enables PIM sparse mode on this interface. The default is disabled. Enable the PIM sparse mode in case of multicast replication.

Example

This example shows how to configure a routing protocol for NVE unicast addresses:

```
switch# configure terminal
switch(config)# interface loopback 10
switch(config-if)# ip address 222.2.2.1/32
switch(config-if)# ip ospf network point-to-point
switch(config-if)# ip router ospf 1 area 0.0.0.0
```

Creating and Configuring an NVE Interface

An NVE interface is the overlay interface that initiates and terminates VXLAN tunnels. You can create and configure an NVE (overlay) interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface nve** *instance*	Creates a VXLAN overlay interface that initiates and terminates VXLAN tunnels. **Note** Only one NVE interface is allowed on the switch.
Step 3	switch(config-if-nve)# **source-interface loopback** *instance*	Specifies a source interface. The source interface must be a loopback interface that is configured on the switch with a valid /32 IP address. This /32 IP address must

	Command or Action	Purpose
		be known by the transit routers in the transport network and the remote VTEPs.

Example

This example shows how to create and configure an NVE interface:

```
switch# configure terminal
switch(config)# interface nve 1
switch(config-if-nve)# source-interface loopback 10
```

Configuring a VXLAN VTEP in vPC

Procedure

Step 1 Enter global configuration mode.

```
switch# configure terminal
```

Step 2 Enable the vPC feature on the device.

```
switch(config)# feature vpc
```

Step 3 Enable the interface VLAN feature on the device.

```
switch(config)# feature interface-vlan
```

Step 4 Enable the LACP feature on the device.

```
switch(config)# feature lacp
```

Step 5 Enable the PIM feature on the device.

```
switch(config)# feature pim
```

Step 6 Enables the OSPF feature on the device.

```
switch(config)# feature ospf
```

Step 7 Define a PIM RP address for the underlay multicast group range.

```
switch(config)# ip pim rp-address 192.168.100.1 group-list 224.0.0/4
```

Step 8 Create the VLAN to be used as a backup link.

```
switch(config)# vlan 10
```

Step 9 Create the SVI used for the backup routed path over the vPC peer-link.

```
switch(config)# interface vlan 10
switch(config-if)# ip address 10.10.10.1/30
switch(config-if)# ip router ospf UNDERLAY area 0
switch(config-if)# ip pim sparse-mode
switch(config-if)# no ip redirects
```

```
switch(config-if)# mtu 9216
```

Step 10 Create primary and secondary IP addresses.

```
switch(config)# interface loopback 0
switch(config-if)# description Control_plane_Loopback
switch(config-if)# ip address x.x.x.x/32
switch(config-if)# ip address y.y.y.y/32 secondary
switch(config-if)# ip router ospf process tag area area id
switch(config-if)# ip pim sparse-mode
switch(config-if)# no shutdown
```

Step 11

```
switch(config)# interface loopback 1
switch(config-if)# description Data_Plane_loopback
switch(config-if)# ip address z.z.z.z/32
switch(config-if)# ip router ospf process tag area area id
switch(config-if)# ip pim sparse-mode
switch(config-if)# no shutdown
```

Step 12 Create a vPC domain.

```
switch(config)# vpc domain 10
```

Step 13 Configure the IPv4 address for the remote end of the vPC peer-keepalive link.

```
switch(config-vpc-domain)# peer-keepalive destination 172.28.x.x
```

Note The system does not form the vPC peer link until you configure a vPC peer-keepalive link

The management ports and VRF are the defaults.

Note We recommend that you configure a separate VRF and use a Layer 3 port from each vPC peer device in that VRF for the vPC peer-keepalive link. For more information about creating and configuring VRFs, see the Cisco Nexus 3600 Series NX-OS Unicast Routing Configuration Guide.

Step 14 Enable Peer-Gateway on the vPC domain.

```
switch(config-vpc-domain)# peer-gateway
```

Note Disable IP redirects on all interface-vlans of this vPC domain for correct operation of this feature.

Step 15 Enable Peer-switch on the vPC domain.

```
switch(config-vpc-domain)# peer-switch
```

Note Disable IP redirects on all interface-vlans of this vPC domain for correct operation of this feature.

Step 16 Enable IP ARP synchronize under the vPC domain to facilitate faster ARP table population following device reload.

```
switch(config-vpc-domain)# ip arp synchronize
```

Step 17 (Optional) Enable IPv6 nd synchronization under the vPC domain to facilitate faster nd table population following device reload.

```
switch(config-vpc-domain)# ipv6 nd synchronize
```

Step 18 Create the vPC peer-link port-channel interface and add two member interfaces.

```
switch(config)# interface port-channel 1
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan 1,100-200
switch(config-if)# mtu 9216
switch(config-if)# vpc peer-link
switch(config-if)# no shutdown
switch(config-if)# interface Ethernet 1/1, 1/20
switch(config-if)# switchport
switch(config-if)# mtu 9216
switch(config-if)# channel-group 1 mode active
switch(config-if)# no shutdown
```

Step 19 Modify the STP hello-time, forward-time, and max-age time.

As a best practice, we recommend changing the **hello-time** to four seconds to avoid unnecessary TCN generation when the vPC role change occurs. As a result of changing the **hello-time**, it is also recommended to change the **max-age** and **forward-time** accordingly.

```
switch(config)# spanning-tree vlan 1-3967 hello-time 4
switch(config)# spanning-tree vlan 1-3967 forward-time 30
switch(config)# spanning-tree vlan 1-3967 max-age 40
```

Step 20 (Optional) Enable the delay restore timer for SVI's.

We recommend that you tune this value when the SVI or VNI scale is high. For example, when the SVI count is 1000, we recommend setting the delay restore for interface-vlan to 45 seconds.

```
switch(config-vpc-domain)# delay restore interface-vlan 45
```

Configuring Replication for a VNI

Replication for VXLAN network identifier (VNI) can be configured in one of two ways:

- Multicast replication

Configuring Multicast Replication

Before you begin

- Ensure that the NVE interface is created and configured.
- Ensure that the source interface is specified.

Procedure

	Command or Action	Purpose
Step 1	switch(config-if-nve)# **member vni** {*vnid* **mcast-group** *multicast-group-addr* | *vnid-range* **mcast-group** *start-addr* [*end-addr*]}	Maps VXLAN VNIs to the NVE interface and assigns a multicast group to the VNIs.

Example

This example shows how to map a VNI to an NVE interface and assign it to a multicast group:

```
switch(config-if-nve)# member vni 5000 mcast-group 225.1.1.1
```

Configuring IGMP Snooping Over VXLAN

Overview of IGMP Snooping Over VXLAN

Starting with Cisco NX-OS Release 7.0(3)F3(4), you can configure IGMP snooping over VXLAN. The configuration of IGMP snooping is same in VXLAN as in configuration of IGMP snooping in regular VLAN domain. For more information on IGMP snooping, see the *Configuring IGMP Snooping* chapter in the Cisco Nexus 3600 NX-OS Multicast Routing Configuration Guide, Release 7.x.

Guidelines and Limitations for IGMP Snooping Over VXLAN

See the following guidelines and limitations for IGMP snooping over VXLAN:

- For IGMP snooping over VXLAN, all the guidelines and limitations of VXLAN apply.
- IGMP snooping over VXLAN is not supported on any FEX enabled platforms and FEX ports.

Configuring IGMP Snooping Over VXLAN

Procedure

	Command or Action	Purpose
Step 1	switch(config)#**ip igmp snooping vxlan**	Enables IGMP snooping for VXLAN VLANs. You have to explicitly configure this command to enable snooping for VXLAN VLANs.
Step 2	switch(config)#**ip igmp snooping disable-nve-static-router-port**	Configures IGMP snooping over VXLAN to not include NVE as static mrouter port using this global CLI command. IGMP snooping over VXLAN has the NVE interface as mrouter port by default.

Verifying the VXLAN Configuration

Use one of the following commands to verify the VXLAN configuration, to display the MAC addresses, and to clear the MAC addresses:

Command	Purpose
show nve interface nve *id*	Displays the configuration of an NVE interface.
show nve vni	Displays the VNI that is mapped to an NVE interface.
show nve peers	Displays peers of the NVE interface.
show nve vxlan-params	Displays the VXLAN UDP port configured.
show mac address-table	Displays both VLAN and VXLAN MAC addresses.
clear mac address-table dynamic	Clears all MAC address entries in the MAC address table.

Example

This example shows how to display the configuration of an NVE interface:

```
switch# show nve interface nve 1
Interface: nve1, State: up, encapsulation: VXLAN
Source-interface: loopback10 (primary: 111.1.1.1, secondary: 0.0.0.0)
```

This example shows how to display the VNI that is mapped to an NVE interface for multicast replication:

```
switch# show nve vni
Interface          VNI        Multicast-group   VNI State
----------------   --------   ---------------   ---------
nve1               5000       225.1.1.1          Up
```

This example shows how to display the VNI that is mapped to an NVE interface for ingress replication:

```
switch# show nve vni
Interface          VNI        Multicast-group   VNI State
----------------   --------   ---------------   ---------
nve1               5000       0.0.0.0           Up
```

This example shows how to display the peers of an NVE interface:

```
switch# show nve peers
Interface          Peer-IP           Peer-State
----------------   ---------------   -------------
nve1               111.1.1.1          Up
```

This example shows how to display the VXLAN UDP port configured:

```
switch# show nve vxlan-params
VxLAN Dest. UDP Port: 4789
```

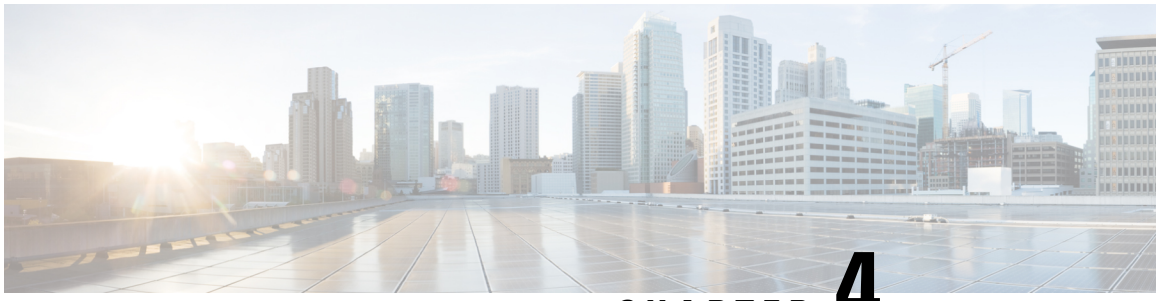
This example shows how to display both VLAN and VXLAN MAC addresses:

```
Added draft comment: hidden contentswitch# show mac address-table
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
```

```
        age - seconds since first seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY   Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
* 109      0000.0410.0902    dynamic   470        F    F  Po2233
* 109      0000.0410.0912    dynamic   470        F    F  Po2233
* 109      0000.0410.0912    dynamic   470        F    F  nve1(1.1.1.200)
* 108      0000.0410.0802    dynamic   470        F    F  Po2233
* 108      0000.0410.0812    dynamic   470        F    F  Po2233
* 107      0000.0410.0702    dynamic   470        F    F  Po2233
* 107      0000.0410.0712    dynamic   470        F    F  Po2233
* 107      0000.0410.0712    dynamic   470        F    F  nve1(1.1.1.200)
* 106      0000.0410.0602    dynamic   470        F    F  Po2233
* 106      0000.0410.0612    dynamic   470        F    F  Po2233
* 105      0000.0410.0502    dynamic   470        F    F  Po2233
* 105      0000.0410.0512    dynamic   470        F    F  Po2233
* 105      0000.0410.0512    dynamic   470        F    F  nve1(1.1.1.200)
* 104      0000.0410.0402    dynamic   470        F    F  Po2233
* 104      0000.0410.0412    dynamic   470        F    F  Po2233
```

This example shows how to clear all MAC address entries in the MAC address table:

```
switch# clear mac address-table dynamic
switch#
```



CHAPTER 4

Configuring VXLAN BGP EVPN

This chapter contains the following sections:

- Information About VXLAN BGP EVPN, on page 19
- Configuring VXLAN BGP EVPN, on page 22
- Configuring Anycast Gateway for VXLAN Routing, on page 24
- Configuring the NVE Interface and VNIs, on page 24
- Configuring BGP on the VTEP, on page 25
- Configuring RD and Route Targets for VXLAN Bridging, on page 26
- Configuring BGP for EVPN on the Spine, on page 26
- Disabling VXLANs, on page 27
- Duplicate Detection for IP and MAC Addresses, on page 28
- Verifying the VXLAN Configuration, on page 29
- Example of VXLAN BGP EVPN (EBGP), on page 30
- Example of VXLAN BGP EVPN (IBGP), on page 41
- Example Show Commands, on page 49

Information About VXLAN BGP EVPN

Guidelines and Limitations for VXLAN BGP EVPN

VXLAN BGP EVPN has the following guidelines and limitations:

- SVI and sub-interfaces as core links are not supported along with Layer 2 GW configurations.
- In a VXLAN EVPN setup, border leaves must use unique route distinguishers, preferably using **auto rd** command. It is not supported to have same route distinguishers in different border leaves.
- ARP suppression is only supported for a VNI if the VTEP hosts the First-Hop Gateway (Distributed Anycast Gateway) for this VNI. The VTEP and the SVI for this VLAN have to be properly configured for the distributed anycast gateway operation, for example, global anycast gateway MAC address configured and anycast gateway feature with the virtual IP address on the SVI.
- The **show** commands with the **internal** keyword are not supported.
- DHCP snooping (Dynamic Host Configuration Protocol snooping) is not supported on VXLAN VLANs.

- SPAN for VXLAN uplink interface is not supported.
- RACLs are not supported on Layer 3 uplinks for VXLAN traffic.
- RACLS and PACLs are not supported for VXLAN VLANs.
- QoS classification is not supported for VXLAN VLANs.
- Uplink ports can be of type Layer 3 interface, sub-interface, or a Layer 3 port-channel interface. However with Layer 2 GW sub-interface uplink ports are not supported.
- For EBGP, it is recommended to use a single overlay EBGP EVPN session between loopbacks.
- Bind NVE to a loopback address that is separate from other loopback addresses that are required by Layer 3 protocols. A best practice is to use a dedicated loopback address for VXLAN.
- VXLAN BGP EVPN does not support an NVE interface in a non-default VRF.
- It is recommended to configure a single BGP session over the loopback for an overlay BGP session.
- The VXLAN UDP port number is used for VXLAN encapsulation. For Cisco Nexus NX-OS, the UDP port number is 4789. It complies with IETF standards and is not configurable.
- VXLAN does not support co-existence with the MPLS feature.
- VXLAN with Layer 3 VPN is not supported.
- VXLAN with ingress replication is not supported.
- MLD snooping is not supported on VXLAN VLANs.
- DHCP snooping is not supported on VXLAN VLANs.

Considerations for VXLAN BGP EVPN Deployment

- A loopback address is required when using the **source-interface config** command. The loopback address represents the local VTEP IP.
- To establish IP multicast routing in the core, IP multicast configuration, PIM configuration, and RP configuration is required.
- VTEP to VTEP unicast reachability can be configured through any IGP/BGP protocol.
- As a best practice when changing the IP address of a VTEP device, enter the **shut** command on the loopback interface used by the NVE interface and then enter the **no shut** command before changing the IP address.
- Every tenant VRF needs a VRF overlay VLAN and SVI for VXLAN routing.

Network Considerations for VXLAN Deployments

- MTU Size in the Transport Network

  Due to the MAC-to-UDP encapsulation, VXLAN introduces 50-byte overhead to the original frames. Therefore, the maximum transmission unit (MTU) in the transport network needs to be increased by 50 bytes. If the overlays use a 1500-byte MTU, the transport network needs to be configured to accommodate

1550-byte packets at a minimum. Jumbo-frame support in the transport network is required if the overlay applications tend to use larger frame sizes than 1500 bytes.

- ECMP and LACP Hashing Algorithms in the Transport Network

  As described in a previous section, Cisco Nexus 3600 platform switches introduce a level of entropy in the source UDP port for ECMP and LACP hashing in the transport network. As a way to augment this implementation, the transport network uses an ECMP or LACP hashing algorithm that takes the UDP source port as an input for hashing, which achieves the best load-sharing results for VXLAN encapsulated traffic.

- Multicast Group Scaling

  The VXLAN implementation on Cisco Nexus 3600 platform switches uses multicast tunnels for broadcast, unknown unicast, and multicast traffic forwarding. Ideally, one VXLAN segment mapping to one IP multicast group is the way to provide the optimal multicast forwarding. It is possible, however, to have multiple VXLAN segments share a single IP multicast group in the core network. VXLAN can support up to 16 million logical Layer 2 segments, using the 24-bit VNID field in the header. With one-to-one mapping between VXLAN segments and IP multicast groups, an increase in the number of VXLAN segments causes a parallel increase in the required multicast address space and the amount of forwarding states on the core network devices. At some point, multicast scalability in the transport network can become a concern. In this case, mapping multiple VXLAN segments to a single multicast group can help conserve multicast control plane resources on the core devices and achieve the desired VXLAN scalability. However, this mapping comes at the cost of suboptimal multicast forwarding. Packets forwarded to the multicast group for one tenant are now sent to the VTEPs of other tenants that are sharing the same multicast group. This causes inefficient utilization of multicast data plane resources. Therefore, this solution is a trade-off between control plane scalability and data plane efficiency.

  Despite the suboptimal multicast replication and forwarding, having multiple-tenant VXLAN networks to share a multicast group does not bring any implications to the Layer 2 isolation between the tenant networks. After receiving an encapsulated packet from the multicast group, a VTEP checks and validates the VNID in the VXLAN header of the packet. The VTEP discards the packet if the VNID is unknown to it. Only when the VNID matches one of the VTEP's local VXLAN VNIDs, does it forward the packet to that VXLAN segment. Other tenant networks will not receive the packet. Thus, the segregation between VXLAN segments is not compromised.

Considerations for the Transport Network

The following are considerations for the configuration of the transport network:

- On the VTEP device:
  - Enable and configure IP multicast.
  - Create and configure a loopback interface with a /32 IP address.
  - Enable IP multicast on the loopback interface.
  - Advertise the loopback interface /32 addresses through the routing protocol (static route) that runs in the transport network.
  - Enable IP multicast on the uplink outgoing physical interface.
- Throughout the transport network:

- Enable and configure IP multicast.

BGP EVPN Considerations for VXLAN Deployment

Configuring VXLAN BGP EVPN

Enabling VXLAN

Enable VXLAN and the EVPN.

Procedure

	Command or Action	Purpose
Step 1	**feature vn-segment**	Enable VLAN-based VXLAN
Step 2	**feature nv overlay**	Enable VXLAN
Step 3	**nv overlay evpn**	Enable the EVPN control plane for VXLAN.

Configuring VLAN and VXLAN VNI

Procedure

	Command or Action	Purpose
Step 1	**vlan** *number*	Specify VLAN.
Step 2	**vn-segment** *number*	Map VLAN to VXLAN VNI to configure Layer 2 VNI under VXLAN VLAN.

Configuring VRF for VXLAN Routing

Configure the tenant VRF.

Procedure

	Command or Action	Purpose
Step 1	**vrf context** *vxlan*	Configure the VRF.
Step 2	**vni** *number*	Specify VNI.
Step 3	**rd auto**	Specify VRF RD (route distinguisher).
Step 4	**address-family ipv4 unicast**	Configure address family for IPv4.

	Command or Action	Purpose
Step 5	**route-target both auto**	**Note** Specifying the **auto** option is applicable only for IBGP. Manually configured route targets are required for EBGP.
Step 6	**route-target both auto evpn**	**Note** Specifying the **auto** option is applicable only for IBGP. Manually configured route targets are required for EBGP.
Step 7	**address-family ipv6 unicast**	Configure address family for IPv6.
Step 8	**route-target both auto**	**Note** Specifying the **auto** option is applicable only for IBGP. Manually configured route targets are required for EBGP.
Step 9	**route-target both auto evpn**	**Note** Specifying the **auto** option is applicable only for IBGP. Manually configured route targets are required for EBGP.

Configuring SVI for Hosts for VXLAN Routing

Configure the SVI for hosts.

Procedure

	Command or Action	Purpose
Step 1	**vlan** *number*	Specify VLAN
Step 2	**interface** *vlan-number*	Specify VLAN interface.
Step 3	**vrf member** *vxlan-number*	Configure SVI for host.
Step 4	**ip address** *address*	Specify IP address.

Configuring VRF Overlay VLAN for VXLAN Routing

Procedure

	Command or Action	Purpose
Step 1	**vlan** *number*	Specify VLAN.

	Command or Action	Purpose
Step 2	**vn-segment** *number*	Specify vn-segment.

Configuring VNI Under VRF for VXLAN Routing

Configures a Layer 3 VNI under a VRF overlay VLAN. (A VRF overlay VLAN is a VLAN that is not associated with any server facing ports. All VXLAN VNIs that are mapped to a VRF, need to have their own internal VLANs allocated to it.)

Procedure

	Command or Action	Purpose
Step 1	**vrf context** *vxlan*	Create a VXLAN Tenant VRF
Step 2	**vni** *number*	Configure Layer 3 VNI under VRF.

Configuring Anycast Gateway for VXLAN Routing

Procedure

	Command or Action	Purpose
Step 1	**fabric forwarding anycast-gateway-mac** *address*	Configure distributed gateway virtual MAC address **Note** One virtual MAC per VTEP **Note** All VTEPs should have the same virtual MAC address
Step 2	**fabric forwarding mode anycast-gateway**	Associate SVI with anycast gateway under VLAN configuration mode.

Configuring the NVE Interface and VNIs

Procedure

	Command or Action	Purpose
Step 1	**interface** *nve-interface*	Configure the NVE interface.
Step 2	**host-reachability protocol bgp**	This defines BGP as the mechanism for host reachability advertisement

	Command or Action	Purpose
Step 3	**member vni** *vni* **associate-vrf**	Add Layer-3 VNIs, one per tenant VRF, to the overlay. **Note** Required for VXLAN routing only.
Step 4	**member vni** *vni*	Add Layer 2 VNIs to the tunnel interface. switch# member vni 900001 associate-vrf
Step 5	**mcast-group** *address*	Configure the mcast group on a per-VNI basis

Configuring BGP on the VTEP

Procedure

	Command or Action	Purpose
Step 1	**router bgp** *number*	Configure BGP.
Step 2	**router-id** *address*	Specify router address.
Step 3	**neighbor** *address* **remote-as** *number*	Define MP-BGP neighbors. Under each neighbor define l2vpn evpn.
Step 4	**address-family ipv4 unicast**	Configure address family for IPv4.
Step 5	**address-family l2vpn evpn**	Configure address family Layer 2 VPN EVPN under the BGP neighbor. **Note** Address-family ipv4 evpn for vxlan host-based routing
Step 6	(Optional) **Allowas-in**	Allows duplicate AS numbers in the AS path. Configure this parameter on the leaf for eBGP when all leafs are using the same AS, but the spines have a different AS than leafs.
Step 7	**send-community extended**	Configures community for BGP neighbors.
Step 8	**vrf** *vrf-name*	Specify VRF.
Step 9	**address-family ipv4 unicast**	Configure address family for IPv4.
Step 10	**advertise** *l2vpn* **evpn**	Enable advertising EVPN routes.
Step 11	**address-family ipv6 unicast**	Configure address family for IPv6.
Step 12	**advertise** *l2vpn* **evpn**	Enable advertising EVPN routes.

Configuring RD and Route Targets for VXLAN Bridging

Procedure

	Command or Action	Purpose
Step 1	**evpn**	Configure VRF.
Step 2	**vni** *number* **l2**	**Note** Only Layer 2 VNIs need to be specified.
Step 3	**rd auto**	Define VRF RD (route distinguisher) to configure VRF context.
Step 4	**route-target import auto**	Define VRF Route Target and import policies.
Step 5	**route-target export auto**	Define VRF Route Target and export policies.

Configuring BGP for EVPN on the Spine

Procedure

	Command or Action	Purpose
Step 1	**route-map permitall permit 10**	Configure route-map. **Note** The route-map keeps the next-hop unchanged for EVPN routes. • Required for eBGP. • Optional for iBGP.
Step 2	**set ip next-hop unchanged**	Set next-hop address. **Note** The route-map keeps the next-hop unchanged for EVPN routes. • Required for eBGP. • Optional for iBGP.
Step 3	**router bgp** *autonomous system number*	Specify BGP.
Step 4	**address-family l2vpn evpn**	Configure address family Layer 2 VPN EVPN under the BGP neighbor.

	Command or Action	Purpose
Step 5	**retain route-target all**	Configure retain route-target all under address-family Layer 2 VPN EVPN [global]. **Note** Required for eBGP. Allows the spine to retain and advertise all EVPN routes when there are no local VNI configured with matching import route targets.
Step 6	**neighbor** *address* **remote-as** *number*	Define neighbor.
Step 7	**address-family l2vpn evpn**	Configure address family Layer 2 VPN EVPN under the BGP neighbor.
Step 8	**disable-peer-as-check**	Disables checking the peer AS number during route advertisement. Configure this parameter on the spine for eBGP when all leafs are using the same AS but the spines have a different AS than leafs. **Note** Required for eBGP.
Step 9	**send-community extended**	Configures community for BGP neighbors.
Step 10	**route-map permitall out**	Applies route-map to keep the next-hop unchanged. **Note** Required for eBGP.

Disabling VXLANs

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enters configuration mode.
Step 2	**no nv overlay evpn**	Disables EVPN control plane.
Step 3	**no feature vn-segment-vlan-based**	Disables the global mode for all VXLAN bridge domains
Step 4	**no feature nv overlay**	Disables the VXLAN feature.
Step 5	(Optional) **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Duplicate Detection for IP and MAC Addresses

Cisco NX-OS supports duplicate detection for IP and MAC addresses. This enables the detection of duplicate IP or MAC addresses based on the number of moves in a given time-interval (seconds).

The default is 5 moves in 180 seconds. (Default number of moves is 5 moves. Default time-interval is 180 seconds.)

- For IP addresses:
  - After the 5th move within 180 seconds, the switch starts a 30 second lock (hold down timer) before checking to see if the duplication still exists (an effort to prevent an increment of the sequence bit). This 30 second lock can occur 5 times (this means 5 moves in 180 seconds for 5 times) before the switch permanently locks or freezes the duplicate entry.
- For MAC addresses:
  - After the 5th move within 180 seconds, the switch starts a 30 second lock (hold down timer) before checking to see if the duplication still exists (an effort to prevent an increment of the sequence bit). This 30 second lock can occur 3 times (this means 5 moves in 180 seconds for 3 times) before the switch permanently locks or freezes the duplicate entry.

The following are example commands to help the configuration of the number of VM moves in a specific time interval (seconds) for duplicate IP-detection:

Command	Description

`switch(config)# fabric forwarding ?`
`anycast-gateway-mac`
`dup-host-ip-addr-detection`
	Available sub-commands:
	• Anycast gateway MAC of the switch.
	• To detect duplicate host addresses in n seconds.

`switch(config)# fabric forwarding dup-host-ip-addr-detection ?` `<1-1000>`	The number of host moves allowed in n seconds. The range is 1 to 1000 moves; default is 5 moves.
`switch(config)# fabric forwarding dup-host-ip-addr-detection 100 ?` `<2-36000>`	The duplicate detection timeout in seconds for the number of host moves. The range is 2 to 36000 seconds; default is 180 seconds.
`switch(config)# fabric forwarding dup-host-ip-addr-detection 100 10`	Detects duplicate host addresses (limited to 100 moves) in a period of 10 seconds.

The following are example commands to help the configuration of the number of VM moves in a specific time interval (seconds) for duplicate MAC-detection:

Command	Description

`switch(config)# l2rib dup-host-mac-detection ?`
`<1-1000>`
`default`
	Available sub-commands for L2RIB:
	• The number of host moves allowed in n seconds. The range is 1 to 1000 moves.
	• Default setting (5 moves in 180 in seconds).

`switch(config)# l2rib dup-host-mac-detection 100 ?` `<2-36000>`	The duplicate detection timeout in seconds for the number of host moves. The range is 2 to 36000 seconds; default is 180 seconds.
`switch(config)# l2rib dup-host-mac-detection 100 10`	Detects duplicate host addresses (limited to 100 moves) in a period of 10 seconds.

Verifying the VXLAN Configuration

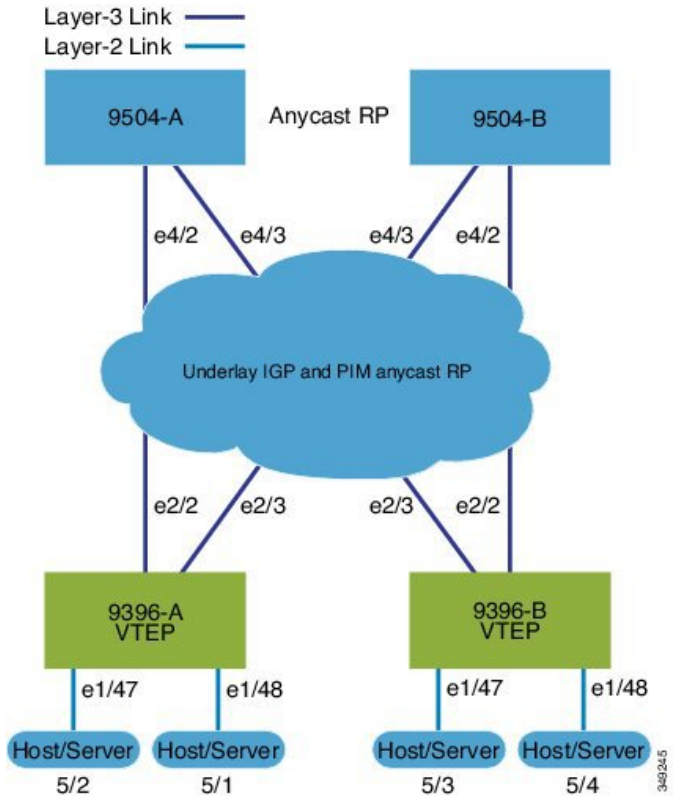
To display the VXLAN configuration information, enter one of the following commands:

Command	Purpose
show tech-support vxlan	Displays related VXLAN tech-support information.
show logging level nve	Displays logging level.
show tech-support nve	Displays related NVE tech-support information.
show tech-support vxlan-evpn	Displays related VXLAN EVPN tech-support information.
show tech-support vxlan platform	Displays VXLAN platform related tech-support information.
show run interface nve	Displays NVE overlay interface configuration.
show nve interface	Displays NVE overlay interface status.
show nve peers	Displays NVE peer status.
show nve peers *peer_IP_address* **interface** *interface_ID* **counters**	Displays per NVE peer statistics.
clear nve peers *peer_IP_address* **interface** *interface_ID* **counters**	Clears per NVE peer statistics.
show nve vni	Displays VXLAN VNI status.
show nve vxlan-params	Displays VXLAN parameters, such as VXLAN destination or UDP port.

Example of VXLAN BGP EVPN (EBGP)

An example of a VXLAN BGP EVPN (EBGP):

Figure 2: VXLAN BGP EVPN Topology (EBGP)



EBGP between Spine and Leaf

- Spine (9504-A)
  - Enable the EVPN control plane

    ```
    nv overlay evpn
    ```

  - Enable the relevant protocols

    ```
    feature bgp
    feature pim
    ```

  - Configure Loopback for local VTEP IP, and BGP

    ```
    interface loopback0
      ip address 10.1.1.1/32
      ip pim sparse-mode
    ```

  - Configure Loopback for Anycast RP

    ```
    interface loopback1
    ```

```
  ip address 100.1.1.1/32
  ip pim sparse-mode
```

- Configure Anycast RP

```
ip pim rp-address 100.1.1.1 group-list 225.0.0.0/8
ip pim rp-candidate loopback1 group-list 225.0.0.0/8
ip pim log-neighbor-changes
ip pim ssm range 232.0.0.0/8
ip pim anycast-rp 100.1.1.1 10.1.1.1
ip pim anycast-rp 100.1.1.1 20.1.1.1
```

- Configure route-map used by EBGP for Spine

```
route-map permitall permit 10
  set ip next-hop unchanged
```

- Enable OSPF for underlay routing

```
router ospf 1
  log-adjacency-changes detail
```

- Configure interfaces for Spine-leaf interconnect

```
interface Ethernet4/2
  ip address 192.168.1.42/24
  ip pim sparse-mode
  no shutdown

interface Ethernet4/3
  ip address 192.168.2.43/24
  ip pim sparse-mode
  no shutdown
```

- Configure the BGP overlay for the EVPN address family.

```
router bgp 100
  router-id 10.1.1.1
  address-family l2vpn evpn
    nexthop route-map permitall
    retain route-target all
  neighbor 30.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      disable-peer-as-check
      send-community extended
      route-map permitall out
  neighbor 40.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      disable-peer-as-check
      send-community extended
      route-map permitall out
```

- Configure the BGP underlay.

```
neighbor 192.168.1.43 remote-as 200
```

```
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
```

- Spine (9504-B)
  - Enable the EVPN control plane and the relevant protocols

```
feature telnet
feature nxapi
feature bash-shell
feature scp-server
nv overlay evpn
feature bgp
feature pim
feature lldp
```

  - Configure Anycast RP

```
ip pim rp-address 100.1.1.1 group-list 225.0.0.0/8
ip pim rp-candidate loopback1 group-list 225.0.0.0/8
ip pim log-neighbor-changes
ip pim ssm range 232.0.0.0/8
ip pim anycast-rp 100.1.1.1 10.1.1.1
ip pim anycast-rp 100.1.1.1 20.1.1.1
vlan 1-1002
route-map permitall permit 10
  set ip next-hop unchanged
```

  - Configure interfaces for Spine-leaf interconnect

```
interface Ethernet4/2
  ip address 192.168.4.42/24
  no shutdown

interface Ethernet4/3
  ip address 192.168.3.43/24
  no shutdown
```

  - Configure Loopback for local VTEP IP, and BGP

```
interface loopback0
  ip address 20.1.1.1/32
```

  - Configure the BGP overlay for the EVPN address family.

```
router bgp 100
  router-id 20.1.1.1
  address-family l2vpn evpn
    retain route-target all
  neighbor 30.1.1.1 remote-as 200
    update-source loopback0
    ebgp-multihop 3
    address-family l2vpn evpn
      disable-peer-as-check
      send-community extended
      route-map permitall out
  neighbor 40.1.1.1 remote-as 200
```

```
    ebgp-multihop 3
    address-family l2vpn evpn
      disable-peer-as-check
      send-community extended
      route-map permitall out
```

- Configure the BGP underlay.

```
neighbor 192.168.1.43 remote-as 200
    address-family ipv4 unicast
      allowas-in
      disable-peer-as-check
```

- Leaf (9396-A)
  - Enable the EVPN control plane

```
nv overlay evpn
```

  - Enable the relevant protocols

```
feature bgp
feature interface-vlan
feature dhcp
```

  - Enable VxLAN with distributed anycast-gateway using BGP EVPN

```
feature vn-segment-vlan-based
feature nv overlay
fabric forwarding anycast-gateway-mac 0000.2222.3333
```

  - Enable PIM RP

```
ip pim rp-address 100.1.1.1 group-list 225.0.0.0/8
```

  - Configure Loopback for BGP

```
interface loopback0
  ip address 30.1.1.1/32
```

  - Configure Loopback for local VTEP IP

```
interface loopback1
  ip address 50.1.1.1/32
```

  - Configure interfaces for Spine-leaf interconnect

```
interface Ethernet2/2
  no switchport
  load-interval counter 1 5
  ip address 192.168.1.22/24
  no shutdown

interface Ethernet2/3
  no switchport
  load-interval counter 1 5
```

```
  ip address 192.168.3.23/24
  no shutdown
```

- Create the VRF overlay VLAN and configure the vn-segment.

```
vlan 101
  vn-segment 900001
```

- Configure VRF overlay VLAN/SVI for the VRF

```
interface Vlan101
  no shutdown
  vrf member vxlan-900001
```

- Create VLAN and provide mapping to VXLAN

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

- Create VRF and configure VNI

```
vrf context vxlan-900001
  vni 900001
```

Note The **rd auto** and **route-target** commands are automatically configured unless one or more are entered as overrides.

```
  rd auto
  address-family ipv4 unicast
    route-target import 65535:101 evpn
    route-target export 65535:101 evpn
    route-target import 65535:101
    route-target export 65535:101
  address-family ipv6 unicast
    route-target import 65535:101 evpn
    route-target export 65535:101 evpn
    route-target import 65535:101
    route-target export 65535:101
```

- Create server facing SVI and enable distributed anycast-gateway

```
interface Vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24
  ipv6 address 4:1:0:1::1/64
  fabric forwarding mode anycast-gateway
  ip dhcp relay address 192.168.100.1 use-vrf default

interface Vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24
```

```
  ipv6 address 4:2:0:1::1/64
  fabric forwarding mode anycast-gateway
```

-

Note You can choose either of the following two options for creating the NVE interface. Use Option 1 for a small number of VNIs. Use Option 2 to configure a large number of VNIs.

Create the network virtualization endpoint (NVE) interface

Option 1

```
interface nve1
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 10000 associate-vrf
  mcast-group 224.1.1.1
  member vni 10001 associate-vrf
  mcast-group 224.1.1.1
  member vni20000
  suppress-arp
  mcast-group 225.1.1.1
  member vni 20001
  suppress-arp
  mcast-group 225.1.1.1
```

Option 2

```
interface nve1
  no shutdown
  source-interface loopback 1
  host-reachibility protocol bgp
  global suppress-arp
  global mcast-group 224.1.1.1 L3
  global mcast-group 255.1.1.1 L2
  member vni 10000 associate-vrf
  member vni 10001 associate-vrf
  member vni 10002 associate-vrf
  member vni 10003 associate-vrf
  member vni 10004 associate-vrf
  member vni 10005 associate-vrf
  member vni 20000
  member vni 20001
  member vni 20002
  member vni 20003
  member vni 20004
  member vni 20005
```

- Configure interfaces for hosts/servers.

```
interface Ethernet1/47
  switchport access vlan 1002
interface Ethernet1/48
  switchport access vlan 1001
```

- Configure BGP

```
router bgp 200
router-id 30.1.1.1
  neighbor 10.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
      allowas-in
      send-community extended
    address-family l2vpn evpn
      allowas-in
      send-community extended
  neighbor 20.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
      allowas-in
      send-community extended
    address-family l2vpn evpn
      allowas-in
      send-community extended
  vrf vxlan-900001

      advertise l2vpn evpn
```

Note The following commands in EVPN mode do not need to be entered.

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

Note The **rd auto** and **route-target auto** commands are automatically configured unless one or more are entered as overrides.

```
rd auto
route-target import auto
route-target export auto



router bgp 200
router-id 30.1.1.1
  neighbor 10.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
      allowas-in
      send-community extended
    address-family l2vpn evpn
      allowas-in
      send-community extended
  neighbor 20.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
      allowas-in
      send-community extended
    address-family l2vpn evpn
      allowas-in
      send-community extended
  vrf vxlan-900001
```

Note The following **advertise** command is optional.

```
advertise l2vpn evpn
```

Note The **rd auto** and **route-target** commands are automatically configured unless one or more are entered as overrides.

Note The following EVPN mode commands are optional.

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

- Leaf (9396-B)
  - Enable the EVPN control plane functionality and the relevant protocols

```
feature telnet
feature nxapi
feature bash-shell
feature scp-server
nv overlay evpn
feature bgp
feature pim
feature interface-vlan
feature vn-segment-vlan-based
feature lldp
feature nv overlay
```

  - Enable VxLAN with distributed anycast-gateway using BGP EVPN

```
fabric forwarding anycast-gateway-mac 0000.2222.3333
```

  - Create the VRF overlay VLAN and configure the vn-segment

```
vlan 1-1002
vlan 101
  vn-segment 900001
```

  - Create VLAN and provide mapping to VXLAN

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

  - Create VRF and configure VNI

```
vrf context vxlan-900001
  vni 900001
```

Note The following commands are automatically configured unless one or more are entered as overrides.

```
  rd auto
  address-family ipv4 unicast
    route-target import 65535:101 evpn
    route-target export 65535:101 evpn
    route-target import 65535:101
    route-target export 65535:101
  address-family ipv6 unicast
    route-target import 65535:101 evpn
    route-target export 65535:101 evpn
    route-target import 65535:101 evpn
    route-target export 65535:101 evpn
```

- Configure internal control VLAN/SVI for the VRF

```
interface Vlan1

interface Vlan101
  no shutdown
  vrf member vxlan-900001
```

- Create server facing SVI and enable distributed anycast-gateway

```
interface Vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24
  ipv6 address 4:1:0:1::1/64
  fabric forwarding mode anycast-gateway

interface Vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24
  ipv6 address 4:2:0:1::1/64
  fabric forwarding mode anycast-gateway
```

- Create the network virtualization endpoint (NVE) interface

Note You can choose either of the following two procedures for creating the NVE interface. Use Option 1 for a small number of VNIs. Use Option 2 to configure a large number of VNIs.

Option 1

```
interface nve1
```

```
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 10000 associate-vrf
  mcast-group 224.1.1.1
  member vni 10001 associate-vrf
  mcast-group 224.1.1.1
  member vni20000
  suppress-arp
  mcast-group 225.1.1.1
  member vni 20001
  suppress-arp
  mcast-group 225.1.1.1
```

Option 2

```
interface nve1
  no shutdown
  source-interface loopback 1
  host-reachibility protocol bgp
  global suppress-arp
  global mcast-group 224.1.1.1 L3
  global mcast-group 255.1.1.1 L2
  member vni 10000 associate-vrf
  member vni 10001 associate-vrf
  member vni 10002 associate-vrf
  member vni 10003 associate-vrf
  member vni 10004 associate-vrf
  member vni 10005 associate-vrf
  member vni 20000
  member vni 20001
  member vni 20002
  member vni 20003
  member vni 20004
  member vni 20005
```

- Configure interfaces for hosts/servers

```
interface Ethernet1/47
  switchport access vlan 1002

interface Ethernet1/48
  switchport access vlan 1001
```

- Configure interfaces for Spine-leaf interconnect

```
interface Ethernet2/1

interface Ethernet2/2
  no switchport
  load-interval counter 1 5
  ip address 192.168.4.22/24
  ip pim sparse-mode
  no shutdown

interface Ethernet2/3
  no switchport
  load-interval counter 1 5
  ip address 192.168.2.23/24
  ip pim sparse-mode
```

```
  no shutdown
```

- Configure Loopback for BGP

```
interface loopback0
  ip address 40.1.1.1/32
```

- Configure Loopback for local VTEP IP

```
interface loopback1
  ip address 51.1.1.1/32
  ip pim sparse-mode
```

- Configure BGP

```
router bgp 200
router-id 40.1.1.1
  neighbor 10.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
      allowas-in
      send-community extended
    address-family l2vpn evpn
      allowas-in
      send-community extended
  neighbor 20.1.1.1 remote-as 100
    update-source loopback0
    ebgp-multihop 3
      allowas-in
      send-community extended
    address-family l2vpn evpn
      allowas-in
      send-community extended
  vrf vxlan-900001
```

Note The following **advertise** command is optional.

```
advertise l2vpn evpn
```

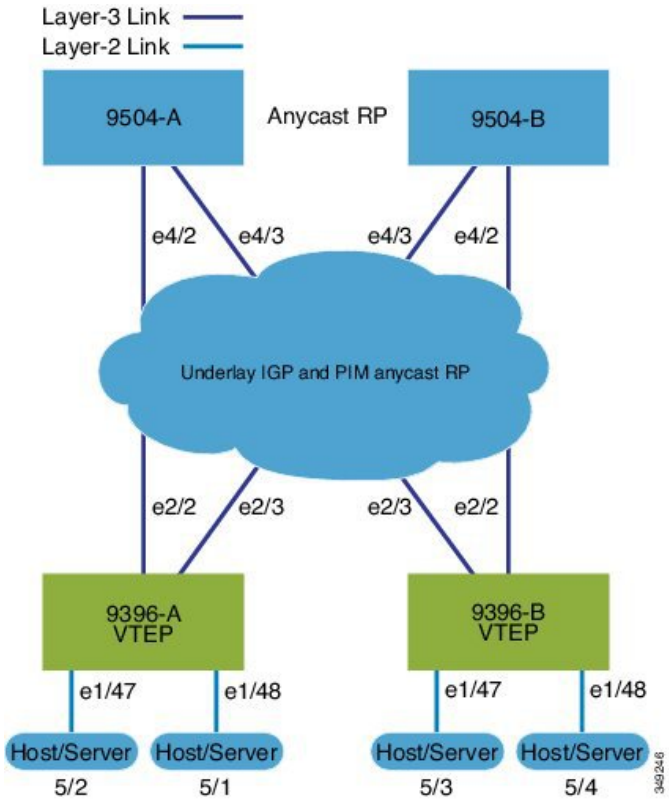
Note The **rd auto** and **route-target** commands are optional unless you want to use them to override the **import** or **export** options.

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

Example of VXLAN BGP EVPN (IBGP)

An example of a VXLAN BGP EVPN (IBGP):

Figure 3: VXLAN BGP EVPN Topology (IBGP)



IBGP between Spine and Leaf

- Spine (9504-A)
  - Enable the EVPN control plane

    ```
    nv overlay evpn
    ```

  - Enable the relevant protocols

    ```
    feature ospf
    feature bgp
    ```

  - Configure Loopback for local VTEP IP, and BGP

    ```
    interface loopback0
      ip address 10.1.1.1/32
      ip router ospf 1 area 0.0.0.0
    ```

  - Enable OSPF for underlay routing

```
router ospf 1
```

- Configure interfaces for Spine-leaf interconnect

```
interface Ethernet4/2
  ip address 192.168.1.42/24
  ip router ospf 1 area 0.0.0.0
  no shutdown

interface Ethernet4/3
  ip address 192.168.2.43/24
  ip router ospf 1 area 0.0.0.0
  no shutdown
```

- Configure BGP

```
router bgp 65535
router-id 10.1.1.1
  neighbor 30.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-reflector-client
  neighbor 40.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-reflector-client
```

- Spine (9504-B)
  - Enable the EVPN control plane and the relevant protocols

```
feature telnet
feature nxapi
feature bash-shell
feature scp-server
nv overlay evpn
feature ospf
feature bgp
feature lldp
```

  - Configure interfaces for Spine-leaf interconnect

```
interface Ethernet4/2
  ip address 192.168.4.42/24
  ip router ospf 1 area 0.0.0.0
  no shutdown

interface Ethernet4/3
  ip address 192.168.3.43/24
  ip router ospf 1 area 0.0.0.0
  no shutdown
```

  - Configure Loopback for local VTEP IP, and BGP

```
interface loopback0
  ip address 20.1.1.1/32
```

```
  ip router ospf 1 area 0.0.0.0
```

- Configure Loopback for Anycast RP

```
interface loopback1
  ip address 100.1.1.1/32
  ip router ospf 1 area 0.0.0.0
```

- Enable OSPF for underlay routing

```
router ospf 1
```

- Configure BGP

```
router bgp 65535
router-id 20.1.1.1
  neighbor 30.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
      route-reflector-client
  neighbor 40.1.1.1 remote-as 65535
    update-source loopback0
     address-family l2vpn evpn
      send-community both
      route-reflector-client
```

- Leaf (9396-A)
  - Enable the EVPN control plane

```
nv overlay evpn
```

  - Enable the relevant protocols

```
feature ospf
feature bgp
feature interface-vlan
```

  - Enabling OSPF for underlay routing

```
router ospf 1
```

  - Configure Loopback for local VTEP IP, and BGP

```
interface loopback0
  ip address 30.1.1.1/32
  ip router ospf 1 area 0.0.0.0
```

  - Configure interfaces for Spine-leaf interconnect

```
interface Ethernet2/2
  no switchport
  ip address 192.168.1.22/24
  ip router ospf 1 area 0.0.0.0
```

```
  no shutdown

interface Ethernet2/3
  no switchport
  ip address 192.168.3.23/24
  ip router ospf 1 area 0.0.0.0
  no shutdown
```

- Create overlay VRF VLAN and configure vn-segment

```
vlan 101
  vn-segment 900001
```

- Configure VRF overlay VLAN/SVI for the VRF

```
interface Vlan101
  no shutdown
  vrf member vxlan-900001
```

- Create VLAN and provide mapping to VXLAN

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

- Create VRF and configure VNI

```
vrf context vxlan-900001
  vni 900001
```

Note The **rd auto** and **route-target** commands are automatically configured unless one or more are entered as overrides.

```
rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn
```

- Create server facing SVI and enable distributed anycast-gateway

```
interface Vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24
  ipv6 address 4:1:0:1::1/64
  fabric forwarding mode anycast-gateway

interface Vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24
  ipv6 address 4:2:0:1::1/64
  fabric forwarding mode anycast-gateway
```

•

Note You can choose either of the following two options for creating the NVE interface. Use Option 1 for a small number of VNIs. Use Option 2 to configure a large number of VNIs.

Create the network virtualization endpoint (NVE) interface

Option 1

```
interface nve1
  no shutdown
  source-interface loopback0
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2001001
    suppress-arp
    mcast-group 225.4.0.1
  member vni 2001002
    suppress-arp
    mcast-group 225.4.0.1
```

Option 2

```
Interface nve1
  source-interface loopback 1
  host-reachability protocol bgp
  global suppress-arp
  global mcast-group 255.1.1.1 L2
  global mcast-group 255.1.1.2 L3
  member vni 10000
  member vni 20000
  member vni 30000
```

• Configure BGP

```
router bgp 65535
router-id 30.1.1.1
  neighbor 10.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  neighbor 20.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  vrf vxlan-900001
    address-family ipv4 unicast
      advertise l2vpn evpn
```

Note The following commands in EVPN mode do not need to be entered.

```
evpn
  vni 2001001 l2
  vni 2001002 l2
```

Note The **rd auto** and **route-target auto** commands are automatically configured unless one or more are entered as overrides.

```
rd auto
    route-target import auto
    route-target export auto
```

Note The **rd auto** and **route-target** commands are automatically configured unless you want to use them to override the **import** or **export** options.

Note The following EVPN mode commands are optional.

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

- Leaf (9396-B)
  - Enable the EVPN control plane functionality and the relevant protocols

```
feature telnet
feature nxapi
feature bash-shell
feature scp-server
nv overlay evpn
feature ospf
feature bgp
feature interface-vlan
feature vn-segment-vlan-based
feature lldp
feature nv overlay
```

  - Enable VxLAN with distributed anycast-gateway using BGP EVPN

```
fabric forwarding anycast-gateway-mac 0000.2222.3333
```

  - Create overlay VRF VLAN and configure vn-segment

```
vlan 1-1002
vlan 101
  vn-segment 900001
```

  - Create VLAN and provide mapping to VXLAN

```
vlan 1001
  vn-segment 2001001
vlan 1002
  vn-segment 2001002
```

- Create VRF and configure VNI

```
vrf context vxlan-900001
  vni 900001
```

Note The **rd auto** and **route-target** commands are automatically configured unless you want to use them to override the **import** or **export** options.

```
rd auto
  address-family ipv4 unicast
    route-target both auto
    route-target both auto evpn
  address-family ipv6 unicast
    route-target both auto
    route-target both auto evpn
```

- Configure internal control VLAN/SVI for the VRF

```
interface Vlan101
  no shutdown
  vrf member vxlan-900001
```

- Create server facing SVI and enable distributed anycast-gateway

```
interface Vlan1001
  no shutdown
  vrf member vxlan-900001
  ip address 4.1.1.1/24
  ipv6 address 4:1:0:1::1/64
  fabric forwarding mode anycast-gateway

interface Vlan1002
  no shutdown
  vrf member vxlan-900001
  ip address 4.2.2.1/24
  ipv6 address 4:2:0:1::1/64
  fabric forwarding mode anycast-gateway
```

-

Note You can choose either of the following two command procedures for creating the NVE interfaces. Use Option 1 for a small number of VNIs. Use Option 2 to configure a large number of VNIs.

Create the network virtualization endpoint (NVE) interface

Option 1

```
interface nve1
```

```
  no shutdown
  source-interface loopback0
  host-reachability protocol bgp
  member vni 900001 associate-vrf
  member vni 2001001
    suppress-arp
    mcast-group 225.4.0.1
  member vni 2001002
    suppress-arp
    mcast-group 225.4.0.1
```

Option 2

```
Interface nve1
  source-interface loopback0
  host-reachability protocol bgp
  global suppress-arp
  global mcast-group 255.4.0.1
  member vni 900001
  member vni 2001001
```

- Configure interfaces for hosts/servers

```
interface Ethernet1/47
  switchport access vlan 1002

interface Ethernet1/48
  switchport access vlan 1001
```

- Configure interfaces for Spine-leaf interconnect

```
interface Ethernet2/1

interface Ethernet2/2
  no switchport
  ip address 192.168.4.22/24
  ip router ospf 1 area 0.0.0.0
  no shutdown

interface Ethernet2/3
  no switchport
  ip address 192.168.2.23/24
  ip router ospf 1 area 0.0.0.0
  no shutdown
```

- Configure Loopback for local VTEP IP, and BGP

```
interface loopback0
  ip address 40.1.1.1/32
  ip router ospf 1 area 0.0.0.0
```

- Enabling OSPF for underlay routing

```
router ospf 1
```

- Configure BGP

```
router bgp 65535
```

```
router-id 40.1.1.1
  neighbor 10.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  neighbor 20.1.1.1 remote-as 65535
    update-source loopback0
    address-family l2vpn evpn
      send-community both
  vrf vxlan-900001
    address-family ipv4 unicast
      advertise l2vpn evpn
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

Note The **rd auto** and **route-target** commands are optional unless you want to use them to override the **import** or **export** options.

```
evpn
  vni 2001001 l2
    rd auto
    route-target import auto
    route-target export auto
  vni 2001002 l2
    rd auto
    route-target import auto
    route-target export auto
```

Example Show Commands

- **show nve peers**

```
9396-B# show nve peers
Interface Peer-IP          Peer-State
--------- ---------------  ----------
nve1      30.1.1.1         Up
```

- **show nve vni**

```
9396-B# show nve vni
Codes: CP - Control Plane        DP - Data Plane
       UC - Unconfigured         SA - Suppress ARP

Interface VNI      Multicast-group   State Mode Type [BD/VRF]      Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1      900001   n/a               Up    CP   L3 [vxlan-900001]
nve1      2001001  225.4.0.1         Up    CP   L2 [1001]          SA
```

```
nve1      2001002  225.4.0.1         Up    CP   L2 [1002]          SA
```

- **show vxlan interface**

```
9396-B# show vxlan interface
Interface       Vlan    VPL Ifindex     LTL             HW VP
=========       ====    ===========     ===             =====
Eth1/47         1002    0x4c07d22e      0x10000         5697
Eth1/48         1001    0x4c07d02f      0x10001         5698
```

- **show bgp l2vpn evpn summary**

```
leaf3# show bgp l2vpn evpn summary
BGP summary information for VRF default, address family L2VPN EVPN
BGP router identifier 40.0.0.4, local AS number 10
BGP table version is 60, L2VPN EVPN config peers 1, capable peers 1
21 network entries and 21 paths using 2088 bytes of memory
BGP attribute entries [8/1152], BGP AS path entries [0/0]
BGP community entries [0/0], BGP clusterlist entries [1/4]

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
40.0.0.1        4    10    8570    8565       60    0    0    5d22h 6
```

- **show bgp l2vpn evpn**

```
leaf3# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 60, local router ID is 40.0.0.4
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: 40.0.0.2:32868
*>i[2]:[0]:[10001]:[48]:[0000.8816.b645]:[0]:[0.0.0.0]/216
                      40.0.0.2                           100          0 i
*>i[2]:[0]:[10001]:[48]:[0011.0000.0034]:[0]:[0.0.0.0]/216
                      40.0.0.2                           100          0 i
```

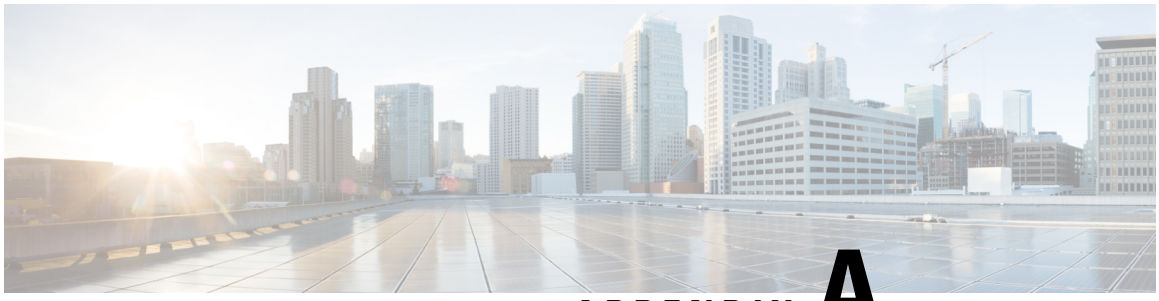
- **show l2route evpn mac all**

```
leaf3# show l2route evpn mac all
Topology    Mac Address    Prod   Next Hop (s)
----------- -------------- ------ ---------------
101         0000.8816.b645 BGP    40.0.0.2
101         0001.0000.0033 Local  Ifindex 4362086
101         0001.0000.0035 Local  Ifindex 4362086
101         0011.0000.0034 BGP    40.0.0.2
```

- **show l2route evpn mac-ip all**

```
leaf3# show l2route evpn mac-ip all
Topology ID Mac Address    Prod Host IP                 Next Hop (s)
```

```
----------- -------------- ---- --------------------------------------------------
101         0011.0000.0034 BGP  5.1.3.2                       40.0.0.2
102         0011.0000.0034 BGP  5.1.3.2                       40.0.0.2
```



APPENDIX A

DHCP Relay in VXLAN BGP EVPN

This chapter contains the following sections:

- DHCP Relay in VXLAN BGP EVPN Overview, on page 53
- Guidelines and Limitations for DHCP Relay , on page 54
- DHCP Relay in VXLAN BGP EVPN Example, on page 54
- Configuring VPC Peers Example, on page 71
- vPC VTEP DHCP Relay Configuration Example, on page 73

DHCP Relay in VXLAN BGP EVPN Overview

DHCP relay is supported by VXLAN BGP EVPN and is useful in a multi-tenant VXLAN EVPN deployment to provision DHCP service to EVPN tenant clients.

In a multi-tenant EVPN environment, DHCP relay uses the following sub-options of Option 82:

- Sub-option 151(0x97) - Virtual Subnet Selection

  (Defined in RFC#6607.)

  Used to convey VRF related information to the DHCP server in an MPLS-VPN and VXLAN EVPN multi-tenant environment.

- Sub-option 11(0xb) - Server ID Override

  (Defined in RFC#5107.)

  The server identifier (server ID) override sub-option allows the DHCP relay agent to specify a new value for the server ID option, which is inserted by the DHCP server in the reply packet. This sub-option allows the DHCP relay agent to act as the actual DHCP server such that the renew requests will come to the relay agent rather than the DHCP server directly. The server ID override sub-option contains the incoming interface IP address, which is the IP address on the relay agent that is accessible from the client. Using this information, the DHCP client sends all renew and release request packets to the relay agent. The relay agent adds all of the appropriate sub-options and then forwards the renew and release request packets to the original DHCP server. For this function, Cisco's proprietary implementation is sub-option 152(0x98). You can use the **ip dhcp relay sub-option type cisco** command to manage the function.

- Sub-option 5(0x5) - Link Selection

  (Defined in RFC#3527.)

The link selection sub-option provides a mechanism to separate the subnet/link on which the DHCP client resides from the gateway address (giaddr), which can be used to communicate with the relay agent by the DHCP server. The relay agent will set the sub-option to the correct subscriber subnet and the DHCP server will use that value to assign an IP address rather than the giaddr value. The relay agent will set the giaddr to its own IP address so that DHCP messages are able to be forwarded over the network. For this function, Cisco's proprietary implementation is sub-option 150(0x96). You can use the **ip dhcp relay sub-option type cisco** command to manage the function.

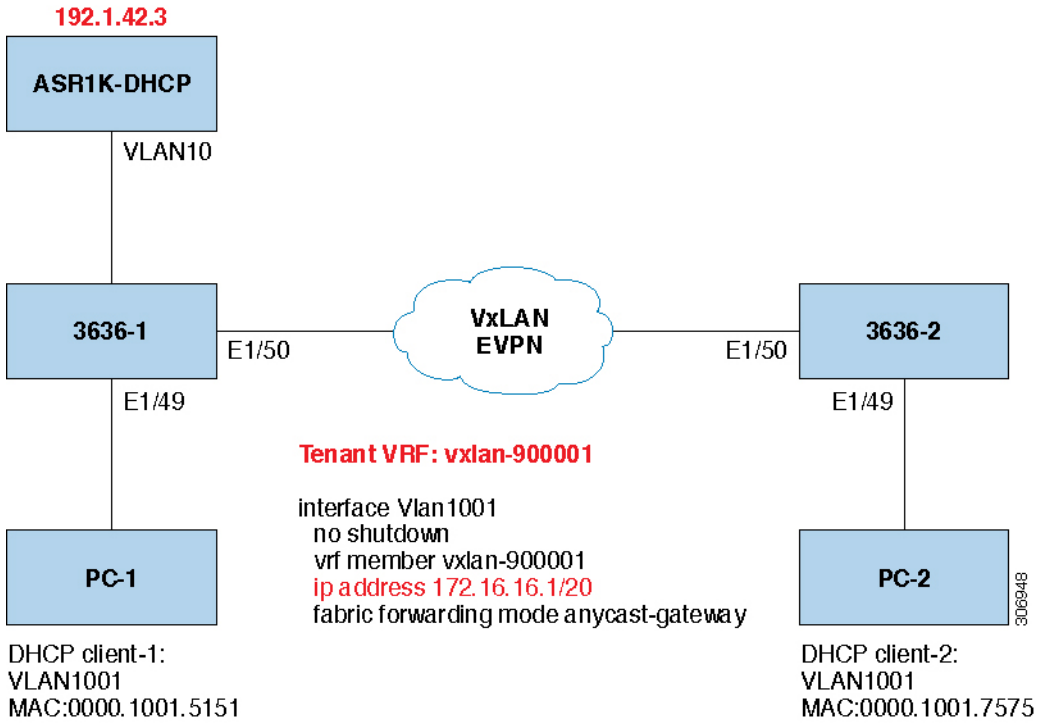
Guidelines and Limitations for DHCP Relay

The following are the guidelines and limitations for DHCP Relay in VXLAN BGP EVPN:

- Beginning in Cisco NX-OS Release 9.2(2), support is added for Cisco Nexus 3636C-R and 36180YC-R.
- IPv6 DHCP is not supported for Cisco Nexus 3636C-R and 36180YC-R switches.

DHCP Relay in VXLAN BGP EVPN Example

Figure 4: Example Topology



Topology characteristics:

- Switches 3636-1 and 3636-2 are VTEPs connected to VXLAN fabric.
- Client1 and client2 are DHCP clients in vlan1001. They belong to tenant VRF vxlan-900001.

- The DHCP server is ASR1K, a router that sits in vlan10.
- DHCP server configuration

```
ip vrf vxlan900001
ip dhcp excluded-address vrf vxlan900001 172.16.16.1 172.16.16.9
ip dhcp pool one
 vrf vxlan900001
 network 172.16.16.0 255.255.240.0
 defaultrouter 172.16.16.1
```

Basic VXLAN BGP EVPN Configuration

- 3636-1

Note You can choose either of the following two procedures to create NVE interfaces. Use the first option for a small number of VNIs. Use the second option to configure a large number of VNIs.

Option 1

```
interface nve1
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 10000 associate-vrf
  mcast-group 224.1.1.1
  member vni 10001 associate-vrf
  mcast-group 224.1.1.1
  member vni20000
  suppress-arp
  mcast-group 225.1.1.1
  member vni 20001
  suppress-arp
  mcast-group 225.1.1.1
```

Option 2

```
interface nve1
  no shutdown
  source-interface loopback 1
  host-reachibility protocol bgp
  global suppress-arp
  global mcast-group 224.1.1.1 L3
  global mcast-group 255.1.1.1 L2
  member vni 10000 associate-vrf
  member vni 10001 associate-vrf
  member vni 10002 associate-vrf
  member vni 10003 associate-vrf
  member vni 10004 associate-vrf
  member vni 10005 associate-vrf
  member vni 20000
  member vni 20001
  member vni 20002
```

```
  member vni 20003
  member vni 20004
  member vni 20005


interfaca Ethernetl/49
  switchport mode trunk
  switchport trunk allowed vlan 10,1001
  spanning—tree port type edge trunk

interface Ethernetl/50
  no switchport
  ip address 192.1.33.2/24
  ip router ospf 1 area 0.0.0.0
  ip pire sparse-mode
  no shutdown

interface loopback0
  ip address 1.1.1.1/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse—mode

interface loopbackl
  vrf member vxlan—900001
  ip address 11.11.11.11/32

router bgp 65535
  router—id 1.1.1.1
  log—neighbor-changes
  neighbor 2.2.2.2 remote—as 65535
    update—source loopback0
    address-family l2vpn evpn
      send-community both
  vrf vxlen—900001
    address—family ipv4 unicast
    network 11.11.11.11/32
    network 192.1.42.0/24
    advertise l2vpn evpn
evpn
  vni 2001001 12
```

Note The **rd auto** and **route-target** commands are automatically configured unless you want to use them to override the **import** or **export** options.

```
rd auto
    route—target import auto
    route—target export auto
```

- 3636-2

```
version 9.2(1)
hostname 3636-1

nv overlay evpn
feature vn—segment—vlan—based
feature nv overlay
```

```
fabric forwarding anycast—gateway—mac 0000.1111.2222

vlan 101
  vn—segment 900001
vlan 1001
  vn—segment 2001001

vrf context vxlan—900001
  vni 900001
  rd auto
  address—family ipv4 unicast
    route-target both auto
    route—target both auto evpn

interface VianlOl
no shutdown
vrf member vxlan-900001
ip forward

interface Vlanl00l
  no shutdown
  vrf member vxlan—900001
  ip address 172.16.16.1/20
  fabric forwarding mcde anycast—gateway
```

Note The **rd** and **route-target** commands are automatically configured unless you want to enter them to override the **import** or **export** options.

```
rd auto
  address—family ipv4 unicast
    route-target both auto
    route—target both auto evpn


interface VianlOl
no shutdown
vrf member vxlan-900001
ip forward

interface Vlanl00l
  no shutdown
  vrf member vxlan—900001
  ip address 172.16.16.1/20
  fabric forwarding mcde anycast—gateway
```

Note You can choose either of the following two procedures for creating the NVE interfaces. Use the first option for a small number of VNIs. Use the second option to configure a large number of VNIs.

Option 1

```
interface nve1
  no shutdown
  source-interface loopback1
  host-reachability protocol bgp
  member vni 10000 associate-vrf
```

```
  mcast-group 224.1.1.1
  member vni 10001 associate-vrf
  mcast-group 224.1.1.1
  member vni20000
  suppress-arp
  mcast-group 225.1.1.1
  member vni 20001
  suppress-arp
  mcast-group 225.1.1.1
```

Option 2

```
interface nve1
  no shutdown
  source-interface loopback 1
  host-reachibility protocol bgp
  global suppress-arp
  global mcast-group 224.1.1.1 L3
  global mcast-group 255.1.1.1 L2
  member vni 10000 associate-vrf
  member vni 10001 associate-vrf
  member vni 10002 associate-vrf
  member vni 10003 associate-vrf
  member vni 10004 associate-vrf
  member vni 10005 associate-vrf
  member vni 20000
  member vni 20001
  member vni 20002
  member vni 20003
  member vni 20004
  member vni 20005



interface Ethernetl/49
  switchport mode trunk
  switchport trunk alluwed vlan 10,1001
  spanning—tree port type edge trunk

interface Ethernetl/50
  no switchport
  ip address 192.1.34.2/24
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
  no shutdown

interface loopback0
  ip address 2.2.2.2/32
  ip router ospf 1 area 0.0.0.0
  ip pim sparse—mode

interface loopbackl
  vrf member vxlan—900001
  ip address 22.22.22.22/32

router bgp 65535
  router—id 2.2.2.2
  log—neighbor-changes
  neighbor 1.1.1.1 remote—as 65535
    update—source loopback0
    address-family l2vpn evpn
      send-community both
  vrf vxlen—900001
    address—family ipv4 unicast
```

```
    network 22.22.22.22/32

    advertise l2vpn evpn
evpn
  vni 2001001 l2
```

Note The **rd** and **route-target** commands are automatically configured unless you want to enter them to override the **import** or **export** options.

```
rd auto
    route—target import auto
    route—target export auto
```

DHCP Relay on VTEPs

The following are common deployment scenarios:

- Client on tenant VRF and server on Layer 3 default VRF.
- Client on tenant VRF (SVI X) and server on the same tenant VRF (SVI Y).
- Client on tenant VRF (VRF X) and server on different tenant VRF (VRF Y).
- Client on tenant VRF and server on non-default non-VXLAN VRF.

The following sections below move vlan10 to different VRFs to depict different scenarios.

Client on Tenant VRF and Server on Layer 3 Default VRF

Put DHCP server (192.1.42.3) into the default VRF and make sure it is reachable from both 3636-1 and 3636-2 through the default VRF.

```
3636-1# sh run int vl 10

!Command: show running-config interface Vlan10
!Time: Mon Aug 7 07:51:16 2018

version 9.2(1)

interface Vlan10
  no shutdown
  ip address 192.1.42.1/24
  ip router ospf 1 area 0.0.0.0

3636-1# ping 192.1.42.3 cou 1

PING 192.1.42.3 (192.1.42.3): 56 data bytes
64 bytes from 192.1.42.3: icmp_seq=0 ttl=254 time=0.593 ms
- 192.1.42.3 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
roundtrip min/avg/max = 0.593/0.592/0.593 ms

3636-2# ping 192.1.42.3 cou 1
```

```
PING 192.1.42.3 (192.1.42.3): 56 data bytes
64 bytes from 192.1.42.3: icmp_seq=0 ttl=252 time=0.609 ms
- 192.1.42.3 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.609/0.608/0.609 ms
```

DHCP Relay Configuration

- 3636-1

```
3636-1# sh run dhcp

!Command: show running—config dhcp
!Time: Mon Aug 6 08:26:00 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface Vlan1001
  ip dhcp relay address 192.1.42.3 use—vrf default
```

- 3636-2

```
3636-2# sh run dhcp

!Command: show running—config dhcp
!Time: Mon Aug 6 08:26:16 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interfaoe Vlan1001
  ip dhcp relay address 192.1.42.3 use—vrf default
```

Debug Output

- The following is a packet dump for DHCP interact sequences.

```
3636-1# ethanalyzer local interface inband display-filter
"udp.srcport==67 or udp.dstport==67" limit-captured frames 0

Capturing on inband
20150824 08:35:25.066530 0.0.0.0 -> 255.255.255.255 DHCP DHCP Discover - Transaction
ID 0x636a38fd
```

```
20150824 08:35:25.068141 192.1.42.1 -> 192.1.42.3 DHCP DHCP Discover - Transaction ID
0x636a38fd
20150824 08:35:27.069494 192.1.42.3 -> 192.1.42.1 DHCP DHCP Offer Transaction - ID
0x636a38fd
20150824 08:35:27.071029 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer Transaction - ID
0x636a38fd
20150824 08:35:27.071488 0.0.0.0 -> 255.255.255.255 DHCP DHCP Request Transaction - ID
 0x636a38fd
20150824 08:35:27.072447 192.1.42.1 -> 192.1.42.3 DHCP DHCP Request Transaction - ID
0x636a38fd
20150824 08:35:27.073008 192.1.42.3 -> 192.1.42.1 DHCP DHCP ACK Transaction - ID
0x636a38fd
20150824 08:35:27.073692 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK Transaction - ID
0x636a38fd
```

Note Ethanalyzer might not capture all DHCP packets because of inband interpretation issues when you use the filter. You can avoid this by using SPAN.

- DHCP Discover packet 3636-1 sent to DHCP server.

  giaddr is set to 192.1.42.1 (ip address of vlan10) and suboptions 5/11/151 are set accordingly.

```
Bootp flags: 0x0000 (unicast)
client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 192.1.42.1 (192.1.42.1)
client MAC address Hughes_01:51:51 (00:00:10:01:51:51)
client hardware address padding: 00000000000000000000
Server host name not given
Boot file name not given
Magic cookie: DHCP
Option: (53) DHCP Message Type
  Length: 1
  DHCP: Discover (1)
Option: (55) Parameter Request List
  Length: 4
  Parameter Request List Item: (1) Subnet Mask
  Parameter Request List Item: (3) Router
  Parameter Request List Item: (58) Renewal Time Value
  Parameter Request List Item: (59) Rebinding Time Value
Option: (61) client identifier
  Length: 7
  Hardware type: Ethernet (0x01)
  Client MAC address: Hughes_01:51:51 (00:00:10:01:51:51)
Option: (82) Agent Information Option
  Length: 47
Option 82 Suboption: (1) Agent Circuit ID
  Length: 10
  Agent Circuit ID: 01080006001e88690030
Option 82 Suboption: (2) Agent Remote ID
  Length: 6
  Agent Remote ID: f8c2882333a5
Option 82 Suboption: (151) VRF name/VPN ID
Option 82 Suboption: (11) Server ID Override
  Length: 4
  Server ID Override: 172.16.16.1 (172.16.16.1)
Option 82 Suboption: (5) Link selection
```

```
  Length: 4
  Link selection: 172.16.16.0 (172.16.16.0)



ASR1K-DHCP# sh ip dhcp bin
Bindings from all pools not associated with VRF:
IP address ClientID/ Lease expiration Type State Interface
         Hardware address/
         User name

Bindings from VRF pool vxlan900001:
IP address ClientID/ Lease expiration Type State Interface
         Hardware address/
         User name
172.16.16.10 0100.0010.0175.75 Aug 25 2015 09:21 AM Automatic Active GigabitEthernet2/1/0
172.16.16.11 0100.0010.0151.51 Aug 25 2015 08:54 AM Automatic Active GigabitEthernet2/1/0

3636-1# sh ip route vrf vxlan900001
IP Route Table for VRF "vxlan900001"
'*' denotes best ucast nexthop
'**' denotes best mcast nexthop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

11.11.11.11/32, ubest/mbest: 2/0, attached
  *via 11.11.11.11, Lo1, [0/0], 18:31:57, local
  *via 11.11.11.11, Lo1, [0/0], 18:31:57, direct
22.22.22.22/32, ubest/mbest: 1/0
  *via 2.2.2.2%default, [200/0], 18:31:57, bgp65535,internal, tag 65535 (evpn)segid:
900001 tunnelid: 0x2020202
encap: VXLAN

172.16.16.0/20, ubest/mbest: 1/0, attached
 *via 172.16.16.1, Vlan1001, [0/0], 18:31:57, direct
172.16.16.1/32, ubest/mbest: 1/0, attached
 *via 172.16.16.1, Vlan1001, [0/0], 18:31:57, local
172.16.16.10/32, ubest/mbest: 1/0
 *via 2.2.2.2%default, [200/0], 00:00:47, bgp65535,internal, tag 65535 (evpn)segid:
900001 tunnelid: 0x2020202
encap: VXLAN

172.16.16.11/32, ubest/mbest: 1/0, attached
 *via 172.16.16.11, Vlan1001, [190/0], 00:28:10, hmm

3636-1# ping 172.16.16.11 vrf vxlan900001 count 1
PING 172.16.16.11 (172.16.16.11): 56 data bytes
64 bytes from 172.16.16.11: icmp_seq=0 ttl=63 time=0.846 ms
- 172.16.16.11 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.846/0.845/0.846 ms

3636-1# ping 172.16.16.10 vrf vxlan900001 count 1
PING 172.16.16.10 (172.16.16.10): 56 data bytes
64 bytes from 172.16.16.10: icmp_seq=0 ttl=62 time=0.874 ms
- 172.16.16.10 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.874/0.873/0.874 ms
```

Client on Tenant VRF (SVI X) and Server on the Same Tenant VRF (SVI Y)

Put DHCP server (192.1.42.3) into VRF of vxlan-900001 and make sure it is reachable from both 3636-1 and 3636-2 through VRF of vxlan-900001.

```
3636-1# sh run int vl 10

!Command: show running-config interface Vlan10
!Time: Mon Aug 6 09:10:26 2018

version 9.2(1)

interface Vlan10
  no shutdown
  vrf member vxlan-900001
  ip address 192.1.42.1/24
```

Because 172.16.16.1 is an anycast address for vlan1001 configured on all the VTEPs, we need to pick up a unique address as the DHCP relay packet's source address to make sure the DHCP server can deliver a response to the original DHCP Relay agent. In this scenario, we use loopback1 and we need to make sure loopback1 is reachable from everywhere of VRF vxlan-900001.

```
3636-1# sh run int lo1

!Command: show running-config interface loopback1
!Time: Mon Aug 6 09:18:53 2018

version 9.2(1)

interface loopback1
  vrf member vxlan-900001
  ip address 11.11.11.11/32

3636-1# ping 192.1.42.3 vrf vxlan900001 source 11.11.11.11 cou 1
PING 192.1.42.3 (192.1.42.3) from 11.11.11.11: 56 data bytes
64 bytes from 192.1.42.3: icmp_seq=0 ttl=254 time=0.575 ms
- 192.1.42.3 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.575/0.574/0.575 ms

3636-2# sh run int lo1

!Command: show running-config interface loopback1
!Time: Mon Aug 6 09:19:30 2018

version 9.2(1)

interface loopback1
  vrf member vxlan900001
  ip address 22.22.22.22/32

3636-2# ping 192.1.42.3 vrf vxlan-900001 source 22.22.22.22 cou 1
PING 192.1.42.3 (192.1.42.3) from 22.22.22.22: 56 data bytes
64 bytes from 192.1.42.3: icmp_seq=0 ttl=253 time=0.662 ms
- 192.1.42.3 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.662/0.662/0.662 ms
```

DHCP Relay Configuration

- 3636-1

```
3636—1# sh run dhcp

!Command: show running-config dhcp
!Time: Mon Aug 6 08:26:00 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
I4ip dhcp relay information option vpn
ipv6 dhcp relay

interface Vlanl00l
  ip dhcp relay address 192.1.42.3
  ip dhcp relay source—interface loopback1
```

- 3636-2

```
3636—2# sh run dhcp

!Command: show running-config dhcp
!Time: Mon Aug 6 08:26:16 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface Vlanl00l
  ip dhcp relay address 192.1.42.3
  ip dhcp relay source—interface loopback1
```

Debug Output

- The following is a packet dump for DHCP interact sequences.

```
3636-1# ethanalyzer local interface inband display-filter
"udp.srcport==67 or udp.dstport==67" limit-captured frames 0

Capturing on inband
20150824 09:31:38.129393 0.0.0.0 -> 255.255.255.255 DHCP DHCP Discover - Transaction
ID 0x860cd13
20150824 09:31:38.129952 11.11.11.11 -> 192.1.42.3 DHCP DHCP Discover - Transaction ID
 0x860cd13
20150824 09:31:40.130134 192.1.42.3 -> 11.11.11.11 DHCP DHCP Offer - Transaction ID
0x860cd13
20150824 09:31:40.130552 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer - Transaction ID
```

```
0x860cd13
20150824 09:31:40.130990 0.0.0.0 -> 255.255.255.255 DHCP DHCP Request - Transaction ID
 0x860cd13
20150824 09:31:40.131457 11.11.11.11 -> 192.1.42.3 DHCP DHCP Request - Transaction ID
0x860cd13
20150824 09:31:40.132009 192.1.42.3 -> 11.11.11.11 DHCP DHCP ACK - Transaction ID
0x860cd13
20150824 09:31:40.132268 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK - TransactionID
0x860cd13
```

Note Ethanalyzer might not capture all DHCP packets because of inband interpretation issues when you use the filter. You can avoid this by using SPAN.

- DHCP Discover packet 3636-1 sent to DHCP server.

  giaddr is set to 11.11.11.11(loopback1) and suboptions 5/11/151 are set accordingly.

```
Bootstrap Protocol
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 1
  Transaction ID: 0x0860cd13
  Seconds elapsed: O
  Bootp flags: 0x0000 (unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 0.0.0.0 (0.0.0.0)
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent iP address: 11.11.11.11 (11.11.11.11)
  Client MAC address: Hughes_01:51:51 (00:00:10:01:51:51)
  Client hardware address padding: 00000000000000000000
  Server host name not given
  Boot file name not given
  Magic cookie: DHCP
  Option: (53) DHCP Message Type
    Length: 1
    DHCP: Discover (1)
  Option: (55) Parameter Request List
  Option: (61) Client Identifier
  Option: (82) Agent Information Option
    Length: 47
  Option 82 suboption: (1) Aqent Circuit ID
  Option 82 suboption: (151) Agent Remote ID
  Option 82 suboption: (11) Server ID Override
    Length: 4
    Server ID override: 172.16.16.1 (172.16.16.1)
  Option 82 suboption: (5) Link selection
    Length: 4
    Link selection: 172.16.16.0 (172.16.16.0)
```

```
ASR1K-DHCP# sh ip dhcp bin
Bindings from all pools not associated with VRF:
IP address ClientID/Lease expiration Type State Interface
         Hardware address/
         User name
```

```
Bindings from VRF pool vxlan-900001:
IP address ClientID/Lease expiration Type State Interface
         Hardware address/
         User name

172.16.16.10 0100.0010.0175.75 Aug 25 2015 10:02 AM Automatic Active GigabitEthernet2/1/0
172.16.16.11 0100.0010.0151.51 Aug 25 2015 09:50 AM Automatic Active GigabitEthernet2/1/0

3636-1# sh ip route vrf vxlan-900001
IP Route Table for VRF "vxlan-900001"
'*' denotes best ucast nexthop
'**' denotes best mcast nexthop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

11.11.11.11/32, ubest/mbest: 2/0, attached
  *via 11.11.11.11, Lo1, [0/0], 19:13:56, local
  *via 11.11.11.11, Lo1, [0/0], 19:13:56, direct
22.22.22.22/32, ubest/mbest: 1/0
  *via 2.2.2.2%default, [200/0], 19:13:56, bgp65535,internal, tag 65535 (evpn)segid:
900001 tunnelid: 0x2020202
encap: VXLAN
172.16.16.0/20, ubest/mbest: 1/0, attached
  *via 172.16.16.1, Vlan1001, [0/0], 19:13:56, direct
172.16.16.1/32, ubest/mbest: 1/0, attached
  *via 172.16.16.1, Vlan1001, [0/0], 19:13:56, local
172.16.16.10/32, ubest/mbest: 1/0
  *via 2.2.2.2%default, [200/0], 00:01:27, bgp65535,
internal, tag 65535 (evpn)segid: 900001 tunnelid: 0x2020202
encap: VXLAN
172.16.16.11/32, ubest/mbest: 1/0, attached
  *via 172.16.16.11, Vlan1001, [190/0], 00:13:56, hmm
192.1.42.0/24, ubest/mbest: 1/0, attached
  *via 192.1.42.1, Vlan10, [0/0], 00:36:08, direct
192.1.42.1/32, ubest/mbest: 1/0, attached
  *via 192.1.42.1, Vlan10, [0/0], 00:36:08, local
9372-1# ping 172.16.16.10 vrf vxlan-900001 cou 1
PING 172.16.16.10 (172.16.16.10): 56 data bytes
64 bytes from 172.16.16.10: icmp_seq=0 ttl=62 time=0.808 ms
- 172.16.16.10 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.808/0.808/0.808 ms

3636-1# ping 172.16.16.11 vrf vxlan-900001 cou 1
PING 172.16.16.11 (172.16.16.11): 56 data bytes
64 bytes from 172.16.16.11: icmp_seq=0 ttl=63 time=0.872 ms
- 172.16.16.11 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.872/0.871/0.872 ms
```

Client on Tenant VRF (VRF X) and Server on Different Tenant VRF (VRF Y)

The DHCP server is placed into another tenant VRF vxlan-900002 so that DHCP response packets can access the original relay agent. We use loopback2 to avoid any anycast ip address that is used as the source address for the DHCP relay packets.

```
3636-1# sh run int vl 10
!Command: show runningconfig interface Vlan10
!Time: Tue Aug 6 08:48:22 2018
```

```
version 9.2(1)
interface Vlan10
  no shutdown
  vrf member vxlan900002
  ip address 192.1.42.1/24

3636-1# sh run int lo2
!Command: show runningconfig interface loopback2
!Time: Tue Aug 7 08:48:57 2018
version 9.2(1)
interface loopback2
  vrf member vxlan900002
  ip address 33.33.33.33/32

3636-2# sh run int lo2
!Command: show runningconfig interface loopback2
!Time: Tue Aug 7 08:48:44 2018
version 9.2(1)
interface loopback2
  vrf member vxlan900002
  ip address 44.44.44.44/32

9372-1# ping 192.1.42.3 vrf vxlan-900002 source 33.33.33.33 cou 1
PING 192.1.42.3 (192.1.42.3) from 33.33.33.33: 56 data bytes
64 bytes from 192.1.42.3: icmp_seq=0 ttl=254 time=0.544 ms
- 192.1.42.3 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.544/0.544/0.544 ms

3636-2# ping 192.1.42.3 vrf vxlan-900002 source 44.44.44.44 count 1
PING 192.1.42.3 (192.1.42.3) from 44.44.44.44: 56 data bytes
64 bytes from 192.1.42.3: icmp_seq=0 ttl=253 time=0.678 ms
- 192.1.42.3 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 0.678/0.678/0.678 ms
```

DHCP Relay Configuration

- 3636-1

```
3636—1# sh run dhcp

!Command: show running—config dhcp
!Time: Mon Aug 6 08:26:00 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlanlOOl
  ip dhcp relay address 192.1.42.3 use—vrf vxlan—900002
  ip dhcp relay source—interface loopback2
```

- 3636-2

```
!Command: show running-config dhcp
!Time: Mon Aug 6 08:26:16 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlaniOOl
  ip dhcp relay address 192.1.42.3 use-vrf vxlan—900002
  ip dhcp relay source—interface loopback2
```

Debug Output

- The following is a packet dump for DHCP interact sequences.

```
3636-1# ethanalyzer local interface inband display-filter "udp.srcport==67 or
udp.dstport==67" limit-captured-frames 0
Capturing on inband
20180806 08:59:35.758314 0.0.0.0 -> 255.255.255.255 DHCP DHCP Discover - Transaction
ID 0x3eebccae
20180806 08:59:35.758878 33.33.33.33 -> 192.1.42.3 DHCP DHCP Discover - Transaction ID
 0x3eebccae
20180806 08:59:37.759560 192.1.42.3 -> 33.33.33.33 DHCP DHCP Offer - Transaction ID
0x3eebccae
20180806 08:59:37.759905 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer - Transaction ID
0x3eebccae
20180806 08:59:37.760313 0.0.0.0 -> 255.255.255.255 DHCP DHCP Request - Transaction ID
 0x3eebccae
20180806 08:59:37.760733 33.33.33.33 -> 192.1.42.3 DHCP DHCP Request - Transaction ID
0x3eebccae
20180806 08:59:37.761297 192.1.42.3 -> 33.33.33.33 DHCP DHCP ACK - Transaction ID
0x3eebccae
20180806 08:59:37.761554 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK - Transaction ID
0x3eebccae
```

- DHCP Discover packet 3636-1 sent to DHCP server.

  giaddr is set to 33.33.33.33 (loopback2) and suboptions 5/11/151 are set accordingly.

```
Bootstrap Protocol
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 1
  Transaction ID: Ox3eebccae
  Seconds elapsed: O
  Bootp flags: 0x0000 (unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 0.0.0.0 (0.0.0.0)
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent IP address: 33.33.33.33 (33.33.33.33)
  Client MAC address: i-iughes_01:51:51 (00:00:10:01:51:51)
```

```
Client hardware address padding: 00000000000000000000
Server host name not given
Boot file name not given
Magic cookie: DHCP
Option: (53) DHCP Message Type
  Length: 1
  DHCP: Discover (1)
Option: (55) Parameter Request List
Option: (61) client identifier
Option: (82) Agent Information option
  Length: 47
Option 82 Suboption: (1) Agent circuit W
Option 82 suboption: (2) Agent Remote 10
Option 82 suboption: (151) VRF name/VPN ID
Option 82 Suboption: (11) Server ID Override
  Length: 4
  Server ID Override: 172.16.16.1 (172.16.16.1)
Option 82 Suboption: (5) Link selection
  Length: 4
  Link selection: 172.16.16.0 (172.16.16.0)
```

Client on Tenant VRF and Server on Non-Default Non-VXLAN VRF

The DHCP server is placed into the management VRF and is reachable the through M0 interface. The IP address changes to 10.122.164.147 accordingly.

```
3636-1# sh run int m0
!Command: show running-config interface mgmt0
!Time: Tue Aug 7 09:17:04 2018
version 9.2(1)
interface mgmt0
  vrf member management
  ip address 10.122.165.134/25

3636-1# ping 10.122.164.147 vrf management cou 1
PING 10.122.164.147 (10.122.164.147): 56 data bytes
64 bytes from 10.122.164.147: icmp_seq=0 ttl=251 time=1.024 ms
- 10.122.164.147 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 1.024/1.024/1.024 ms

3636-2# sh run int m0
!Command: show running-config interface mgmt0
!Time: Tue Aug 25 09:17:47 2015
version 7.0(3)I1(3)
interface mgmt0
  vrf member management
  ip address 10.122.165.148/25

3636-2# ping 10.122.164.147 vrf management cou 1
PING 10.122.164.147 (10.122.164.147): 56 data bytes
64 bytes from 10.122.164.147: icmp_seq=0 ttl=251 time=1.03 ms
- 10.122.164.147 ping statistics -
1 packets transmitted, 1 packets received, 0.00% packet loss
round-trip min/avg/max = 1.03/1.03/1.03 ms
```

DHCP Relay Configuration

- 3636-1

```
3636—1# sh run dhcp 3636—2# sh run dhcp

!Command: show running-config dhcp
!Time: Mon Aug 6 08:26:00 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhcp relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlanlOOl
  ip dhcp relay address 10.122.164.147 use—vrf management
```

- 3636-2

```
3636-2# sh run dhcp
!Command: show running-config dhcp
!Time: Tue Aug 7 09:17:47 2018

version 9.2(1)
feature dhcp

service dhcp
ip dhcp relay
ip dhop relay information option
ip dhcp relay information option vpn
ipv6 dhcp relay

interface VlanlOOl
ip dhcp relay address 10.122.164.147 use—vrf management
```

Debug Output

- The following is a packet dump for DHCP interact sequences.

```
3636-1# ethanalyzer local interface inband display-filter "udp.srcport==67 or
udp.dstport==67" limit-captured-frames 0
Capturing on inband
20180806 09:30:54.214998 0.0.0.0 -> 255.255.255.255 DHCP DHCP Discover - Transaction
ID 0x28a8606d
20180806 09:30:56.216491 172.16.16.1 -> 172.16.16.11 DHCP DHCP Offer - Transaction ID
0x28a8606d
20180806 09:30:56.216931 0.0.0.0 -> 255.255.255.255 DHCP DHCP Request - Transaction ID
 0x28a8606d
20180806 09:30:56.218426 172.16.16.1 -> 172.16.16.11 DHCP DHCP ACK - Transaction ID
0x28a8606d

3636-1# ethanalyzer local interface mgmt display-filter "ip.src==10.122.164.147 or
ip.dst==10.122.164.147" limit-captured-frames 0
Capturing on mgmt0
20180806 09:30:54.215499 10.122.165.134 -> 10.122.164.147 DHCP DHCP Discover - Transaction
 ID 0x28a8606d
20180806 09:30:56.216137 10.122.164.147 -> 10.122.165.134 DHCP DHCP Offer - Transaction
 ID 0x28a8606d
```

```
20180806 09:30:56.217444 10.122.165.134 -> 10.122.164.147 DHCP DHCP Request - Transaction
 ID 0x28a8606d
20180806 09:30:56.218207 10.122.164.147 -> 10.122.165.134 DHCP DHCP ACK - Transaction
ID 0x28a8606d
```

- DHCP Discover packet 3636-1 sent to DHCP server.

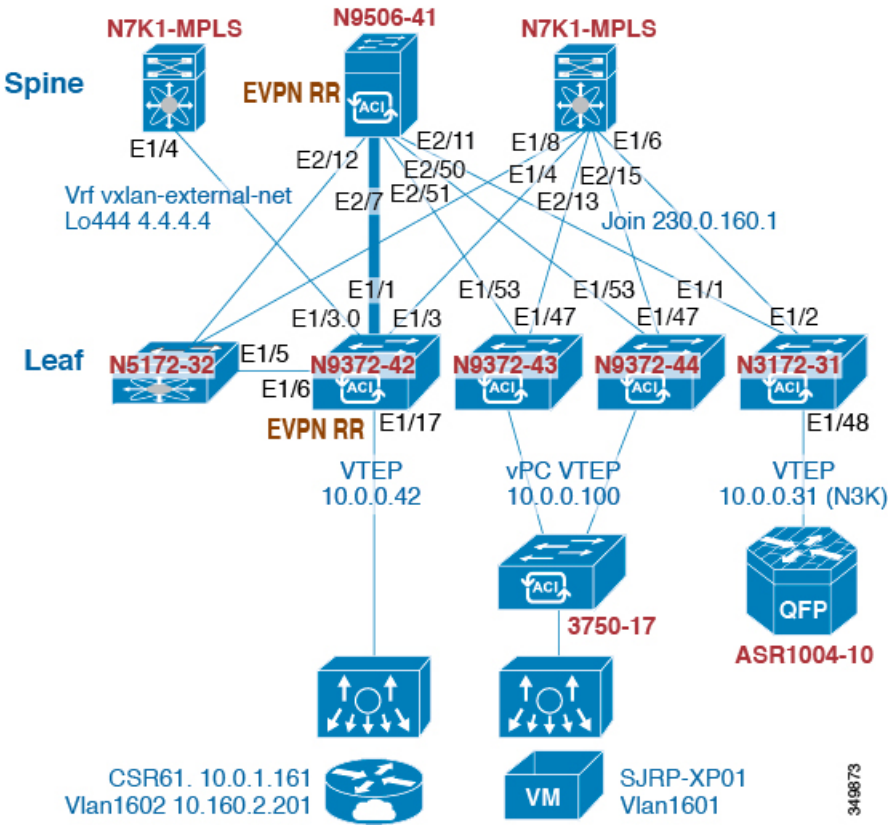
  giaddr is set to 10.122.165.134 (mgmt0) and suboptions 5/11/151 are set accordingly.

```
Bootstrap Protocol
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 1
  Transaction ID: 0x28a8606d
  Seconds elapsed: O
  Bootp flags: 0x0000 (Unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 0.0.0.0 (0.0.0.0)
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent IP address: 10.122.165.134 (10.122.165.134)
  Client MAC address: Hughes_01:51:51 (00:00:10:01:51:51)
  Client hardware address padding: 00000000000000000000
  Server host name not given
  Boot file name not given
  Magic cookie: DHCP
  Option: (53) DHCP Message Type
    Length: 1
    DHCP: Discover (1)
  Option: (55) Parameter Request List
  Option: (61) Client identifier
  Option: (82) Agent Information Option
    Length: 47
    Option 82 Suboption: (1) Agent Circuit ID
    Option 82 Suboption: (2) Aqent Remote ID
    Option 82 Suboption: (151) VRF name/VPN ID
    Option 82 Suboption: (11) Server ID Override
      Length: 4
      Server ID Override: 172.16.16.1 (172.16.16.1)
    Option 82 Suboption: (5) Link selection
      Length: 4
      Link selection: 172.16.16.0 (172.16.16.0)
```

Configuring VPC Peers Example

The following is an example of how to configure routing between VPC peers in the overlay VLAN for a DHCP relay configuration.

DHCP Relay in VXLAN BGP EVPN



- Enable DHCP service.

```
service dhcp
```

- Configure DHCP relay.

```
ip dhcp relay
ip dhcp relay information option
ip dhcp relay sub-option type cisco
ip dhcp relay information option vpn
```

- Create loopback under VRF where you need DHCP relay service.

```
interface loopback601
  vrf member evpn-tenant-kk1
  ip address 160.1.0.43/32
  ip router ospf 1 area 0     /* Only required for VPC VTEP. */
```

- Advertise LoX into the Layer 3 VRF BGP.

```
Router bgp 2
vrf X
   network 10.1.1.42/32
```

- Configure DHCP relay on the SVI under the VRF.

```
interface Vlan1601
  vrf member evpn-tenant-kk1
  ip address 10.160.1.254/24
  fabric forwarding mode anycast-gateway
  ip dhcp relay address 10.160.2.201
  ip dhcp relay source-interface loopback601
```

- Configure Layer 3 VNI SVI with **ip forward**.

```
interface Vlan1600
  vrf member evpn-tenant-kk1
    ip forward
```

- Create the routing VLAN/SVI forthe VPC VRF.

Note Only required for VPC VTEP.

```
Vlan 1605
interface Vlan1605
  vrf member evpn-tenant-kk1
  ip address 10.160.5.43/24
  ip router ospf 1 area 0.0.0.41
```

- Create the VRF routing.

Note Only required for VPC VTEP.

```
router ospf 1
vrf evpn-tenant-kk1
    router-id 10.160.5.43
```

vPC VTEP DHCP Relay Configuration Example

To address a need to configure a VLAN that is allowed across the MCT/peer-link, such as a vPC VLAN, an SVI can be associated to the VLAN and is created within the tenant VRF. This becomes an underlay peering, with the underlay protocol, such as OSPF, that needs the tenant VRF instantiated under the routing process.

Alternatively, instead of placing the SVI within the routing protocol and instantiate the Tenant-VRF under the routing process, you can use the static routes between the vPC peers across the MCT. This approach ensures that the reply from the server returns to the correct place and each VTEP uses a different loopback interface for the GiAddr.

The following are examples of these configurations:

- Configuration of SVI within underlay routing:

```
/* vPC Peer-1 */

router ospf UNDERLAY
vrf tenant-vrf

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.1/30
  ip router ospf UNDERLAY area 0.0.0.0


/* vPC Peer-2 */

router ospf UNDERLAY
vrf tenant-vrf

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.2/30
  ip router ospf UNDERLAY area 0.0.0.0
```

- Configuration of SVI using static routes between vPC peers across the MCT:

```
/* vPC Peer-1 */

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.1/30

vrf context tenant-vrf
ip route 192.168.1.2/30 192.168.1.1


/* vPC Peer-2 */

interface Vlan2000
  no shutdown
  mtu 9216
  vrf member tenant-vrf
  ip address 192.168.1.2/30

vrf context tenant-vrf
ip route 192.168.1.1/30 192.168.1.2
```