Juniper Networks | Engineering Simplicity

# Network Configuration Example

## IP Clos Fabric for a Campus Network

Published
2023-10-16

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net



*Network Configuration Example IP Clos Fabric for a Campus Network*


The information in this document is current as of the date on the title page.

## YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

## END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# About This Guide

Use this network configuration example to configure an IP Clos network for a campus network.

# Table of Contents

CHAPTER 1

# IP Clos (End-to-End) EVPN-VXLAN Network

# Overview of IP Clos Fabrics for Campus Networks

IN THIS SECTION



## About This Network Configuration Example

This network configuration example (NCE) describes how to deploy an IP Clos architecture to support a campus networking environment. The use case shows how you can deploy a single campus fabric that uses EVPN in the control plane and VXLAN tunnels in the overlay network with Juniper Mist Access Points integration.

## Use Case Overview

IN THIS SECTION



Enterprise networks are undergoing massive transitions to accommodate the growing demand for cloud-ready networks and the plethora of IoT and mobile devices. As the number of devices grows, so does network complexity with an ever greater need for scalability and segmentation. To meet these challenges, you need a network with increased scalability and operational simplification. IP Clos networks provide increased scalability and segmentation using a well-understood standards-based approach.

Most traditional campus architectures use single-vendor, chassis-based technologies that work well in small, static campuses with few endpoints. However, they are too rigid to support the scalability and changing needs of modern large enterprises.

A Juniper Networks EVPN-VXLAN fabric is a highly scalable architecture that is simple, programmable, and built on a standards-based architecture that is common across campuses and data centers.

The EVPN-VXLAN campus architecture uses a Layer 3 IP-based underlay network and an EVPN-VXLAN overlay network. The simple IP-based Layer 3 network underlay limits the Layer 2 broadcast domain and eliminates the need for spanning tree protocols (STP). A flexible overlay network based on a VXLAN tunnels combined with an EVPN control plane efficiently provides Layer 3 or Layer 2 connectivity.

This architecture decouples the virtual topology from the physical topology, which improves network flexibility and simplifies network management. Endpoints that require Layer 2 adjacency, such as IoT devices, can be placed anywhere in the network and remain connected to the same logical Layer 2 network.

With an EVPN-VXLAN campus architecture, you can easily add core, distribution, and access layer devices as your business grows without having to redesign the network. EVPN-VXLAN is vendor-agnostic, so you can use the existing access layer infrastructure and gradually migrate to access layer switches that support EVPN-VXLAN capabilities.

## Benefits of Campus Fabric: IP Clos

With increasing number of devices connecting to the network, you will need to scale your campus network rapidly without adding complexity. Many IoT devices have limited networking capabilities and require Layer 2 adjacency across buildings and campuses. Traditionally, this problem was solved by extending VLANs between endpoints using data plane based flood and learn mechanisms. This approach is inefficient because it uses excessive network bandwidth. It is also difficult to manage because you need to configure and manually manage VLANs to extend them to new network ports. This problem increases multifold when you take into consideration the explosive growth of IoT and mobile devices.

The benefit of having an IP Clos network is that you can easily connect a number of switches in an IP Clos network or campus fabric. IP Clos extends the EVPN fabric to connect VLANs across multiple buildings by stretching the Layer 2 VXLAN network with routing occurring in the access device. The IP Clos network encompasses the distribution, core, and access layers of your topology.

An EVPN-VXLAN fabric solves these issues and provides the following benefits:

- Reduced flooding and learning—Control plane-based Layer 2/Layer 3 learning reduces the flood and learn issues associated with data plane learning. Learning MAC addresses in the forwarding plane has an adverse impact on network performance as the number of endpoints grows. The EVPN control plane handles the exchange and learning of routes, so newly learned MAC addresses are not exchanged in the forwarding plane

- Scalability—Faster control plane-based Layer 2/Layer 3 learning allows the EVPN-VXLAN network to scale up to support a larger number of mobile devices.

- Consistent network—A universal EVPN-VXLAN-based architecture across campuses and data centers means a consistent end-to-end network for endpoints and applications. In addition, you can enable

microsegmentation and macrosegmentation with EVPN-VXLAN to minimize Layer 2 flooding, reduce security threats, and simplify the network.

- Location-agnostic connectivity—The EVPN-VXLAN campus architecture provides a consistent endpoint experience no matter where the endpoint is located. Some endpoints require Layer 2 reachability, such as legacy building security systems or IoT devices. The Layer 2 VXLAN overlay provides Layer 2 reachability across campuses without any changes to the underlay network. With our standards-based network access control integration, an endpoint can be connected anywhere in the network.

## Technical Overview

IN THIS SECTION



### Understanding VXLAN

Network overlays are created by encapsulating traffic and tunneling it over a physical network. The Virtual Extensible LAN (VXLAN) tunneling protocol encapsulates Layer 2 Ethernet frames in Layer 4 UDP datagrams that are themselves encapsulated into IP for transport over the underlay. VXLAN enables virtual Layer 2 subnets (or VLANs) that can span the underlying physical Layer 3 network.

In a VXLAN overlay network, each Layer 2 subnet or segment is uniquely identified by a virtual network identifier (VNI). A VNI segments traffic the same way that a VLAN ID does. As is the case with VLANs, endpoints within the same virtual network can communicate directly with each other. Endpoints in different virtual networks require a device that supports inter-VXLAN routing, which is typically a router or a high-end switch.

The entity that performs VXLAN encapsulation and decapsulation is called a VXLAN tunnel endpoint (VTEP). Each VXLAN tunnel endpoint is assigned a unique IP address. Normally these VTEP addresses match the device's loopback address.

## VXLAN Control Plane Limitations

VXLAN can be deployed as a tunneling protocol across a Layer 3 IP fabric data center without a control plane protocol. However, the use of VXLAN tunnels alone does not change the flood and learn behavior of the Ethernet protocol, which has inherent limitations in terms of scalability and efficiency.

The two primary methods for using VXLAN without a control plane protocol—static unicast VXLAN tunnels and VXLAN tunnels that are signaled with a multicast underlay—do not solve the inherent flood and learn problem and are difficult to scale in large multitenant environments. An EVPN control plane provides a scalable solution for the flood and learn problems with Ethernet.

## Understanding EVPN

Ethernet VPN (EVPN) is a standards-based protocol that provides virtual multipoint bridged connectivity between different domains over an IP or IP/MPLS backbone network. EVPN enables seamless multitenant, flexible services that can be extended on demand.

EVPN leverages BGP signaling to allow the network to carry both Layer 2 MAC and Layer 3 IP information simultaneously to optimize routing and switching decisions. This control plane technology uses Multiprotocol BGP (MP-BGP) for MAC and IP address endpoint distribution, where MAC addresses are treated as routes. EVPN enables devices acting as VTEPs to exchange reachability information with each other about their endpoints.

EVPN provides multipath forwarding and redundancy through an all-active model. The access layer can connect to two or more distribution devices and forward traffic using all of the links. If an access link or distribution device fails, traffic flows from the access layer toward the distribution layer using the remaining active links. For traffic in the other direction, remote distribution devices update their forwarding tables to send traffic to the remaining active distribution devices connected to the multihomed Ethernet segment.

The benefits of using EVPNs include:

- MAC address mobility
- Multitenancy
- Load balancing across multiple links
- Fast convergence

The technical capabilities of EVPN include:

- Minimal flooding—EVPN creates a control plane that shares end host MAC addresses between VTEPs in the same EVPN segment, which minimizes flooding and facilitates MAC address learning.
- Multihoming—EVPN supports multihoming for client devices. A control protocol like EVPN that enables synchronization of endpoint addresses between the distribution switches is needed to support multihoming, because traffic traveling across the topology needs to be intelligently moved across multiple paths.
- Aliasing—EVPN leverages all-active multihoming to allow a remote distribution device to load-balance traffic across the network toward the access layer.
- Split horizon—Split horizon prevents the looping of broadcast, unknown unicast, and multicast (BUM) traffic in a network. With split horizon, a packet is never sent back over the same interface it was received on.

## Underlay Network

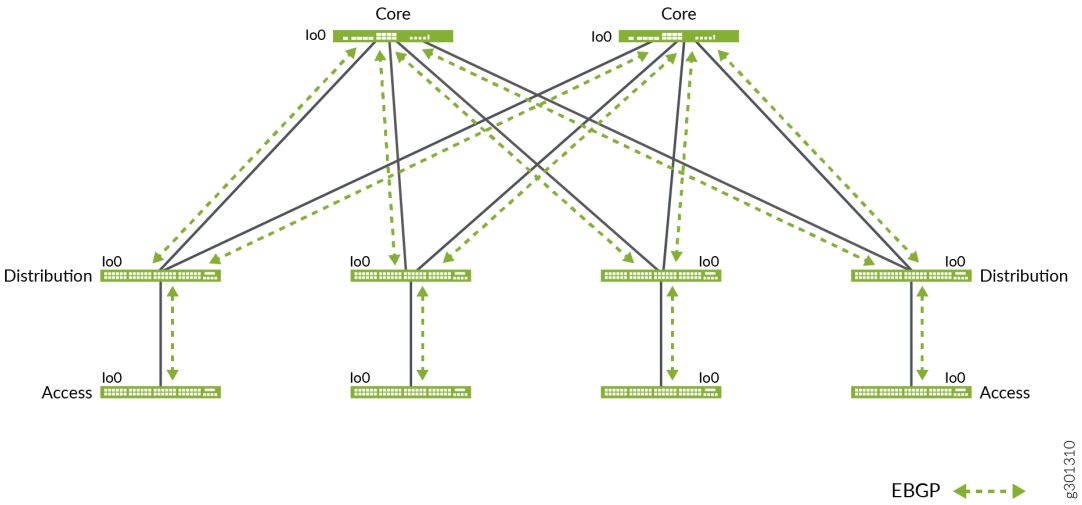
An EVPN-VXLAN fabric architecture makes the network infrastructure simple and consistent across campuses and data centers. All the core and distribution devices must be connected to each other using a Layer 3 infrastructure. We recommend deploying a Clos-based IP fabric with a spine-leaf-based topology to ensure predictable performance and to enable a consistent, scalable architecture.

The primary requirement in the underlay network is that all core and distribution devices have loopback reachability to one another. The loopback addresses are used to establish IBGP peering relationships used to exchange EVPN routes in the overlay network.

You can use any Layer 3 routing protocol to exchange loopback addresses between the access, core, and distribution devices. BGP provides benefits like better prefix filtering, traffic engineering, and route tagging, while OSPF is relatively simple to configure and troubleshoot.

We are using EBGP as the underlay routing protocol in this example because of its ease of use. Figure 1 on page 7 shows the topology of the underlay network.

**Figure 1: Underlay Network Topolgy**

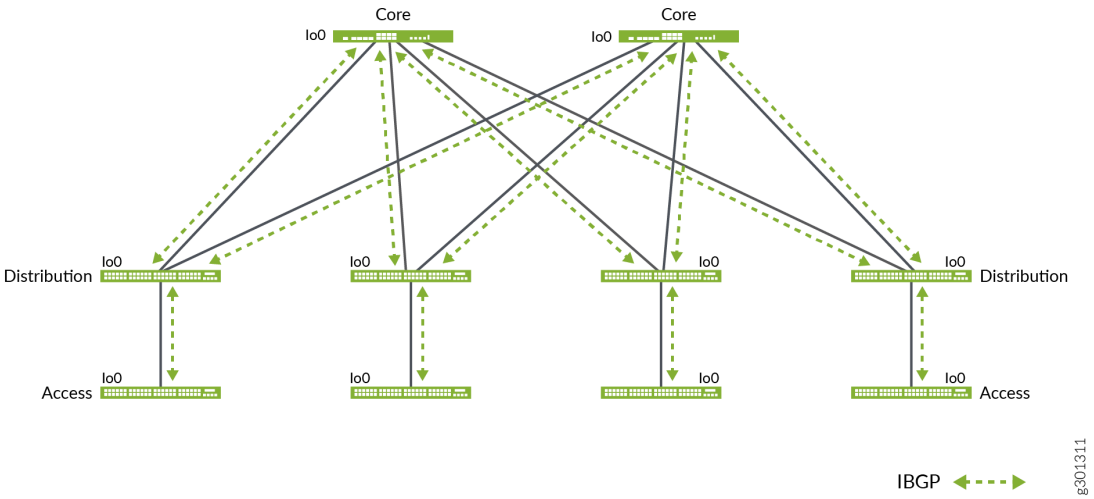



## Overlay Network Control Plane

MP-BGP with EVPN signaling acts as the overlay control plane protocol. The core and distribution devices establish IBGP sessions between each other.

To eliminate the need for full mesh IBGP sessions between all devices, the core switches act as route reflectors with the access and distribution devices functioning as route reflector clients. Route reflectors enable simple and consistent IBGP configuration on all distribution switches and dramatically improves control plane scalability. In this example, we use hierarchical route-reflectors. Figure 2 on page 8 shows the topology of the overlay network.

**Figure 2: Overlay Network Topology**




## Overlay Data Plane

This architecture uses VXLAN as the overlay data plane encapsulation protocol. A Juniper switch that functions as a Layer 2 or Layer 3 VXLAN gateway acts as the VTEP to encapsulate and decapsulate data packets.

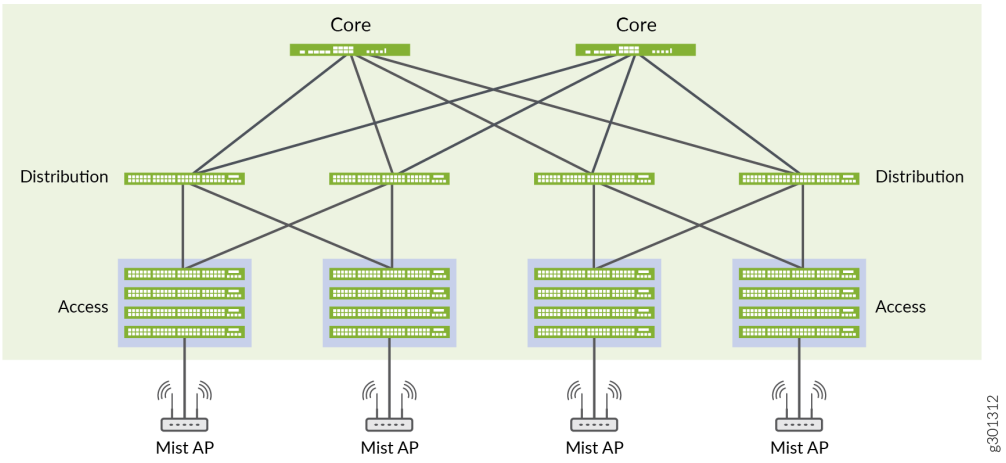
## Access Layer

The access layer provides network connectivity to end-user devices, such as personal computers, VoIP phones, printers, IoT devices, as well as connectivity to wireless access point devices. In this IP Clos campus design, the EVPN-VXLAN network extends all the way to the access layer switches.

In this example, each access switch or Virtual Chassis is multihomed to two or more distribution switches. With EVPN running as the control plane protocol, any access switch or Virtual Chassis device can enable active-active multihoming on its interfaces. EVPN provides a standards-based multihoming solution that scales horizontally across any number of distribution layer switches.

Figure 3 on page 9 shows the topology of the access layer devices after multihoming.

**Figure 3: Access Layer Topology**




## Juniper Access points

For this example we choose Juniper Access points as our preferred access point devices. They are designed from the ground up to meet the stringent networking needs of the modern cloud and smart-device era. Juniper Mist delivers unique capabilities for both wired and wireless LAN.

- Wired and wireless assurance—Mist is enabled with wired and wireless assurance. Once configured, Service Level Expectations (SLE) for key wired and wireless performance metrics such as throughput, capacity, roaming, and uptime are addressed in the Mist platform. This NCE uses Mist wired assurance services.
- Marvis—An integrated AI engine that provides rapid wired and wireless troubleshooting, trending analysis, anomaly detection, and proactive problem remediation.

Today's IT departments look for a cohesive approach for managing wired and wireless networks. Juniper Networks offers a solution that simplifies and automate operations, provides end-to-end troubleshooting, and ultimately evolves into the Self-Driving Network™. The Integration of the Mist platform in this NCE addresses both of these challenges. For more details on Mist integration and EX switches, see How to Connect Mist Access Points and Juniper EX Series Switches.

## Campus IP Clos Fabric High Level Architecture

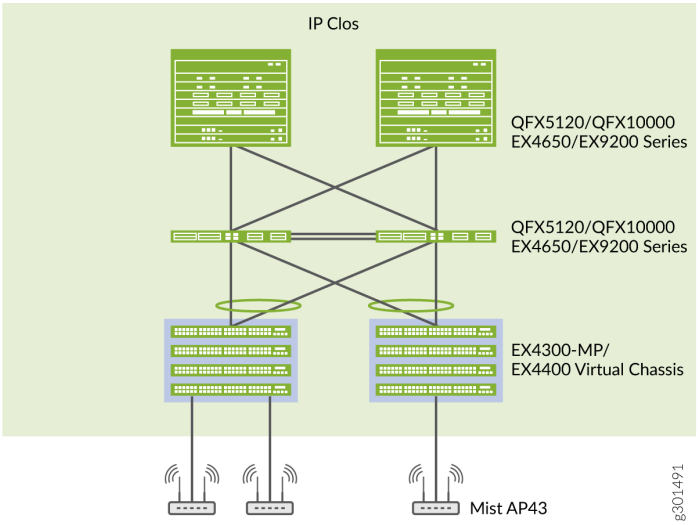
The campus fabric, with an EVPN-VXLAN architecture, decouples the overlay network from the underlay network. This approach addresses the needs of the modern enterprise network by allowing network administrators to create logical Layer 2 networks across one or more Layer 3 networks. By configuring different routing instances, you can enforce the separation of virtual networks because each routing instance has its own separate routing and switching table.

VXLAN is the overlay data plane encapsulation protocol that tunnels Ethernet frames between network endpoints over the Layer 3 IP network. Devices that perform VXLAN encapsulation and decapsulation for the network are referred to as a VXLAN tunnel endpoint (VTEP). Before a VTEP sends a frame into a VXLAN tunnel, it wraps the original frame in a VXLAN header that includes a virtual network identifier (VNI). The VNI maps the packet to the original VLAN at the ingress switch. After applying a VXLAN header, the frame is encapsulated into a UDP/IP packet for transmission to the remote VTEP over the IP fabric.

A campus fabric based on EVPN-VXLAN is a modern and scalable network that uses a BGP, OSPF, or IS-IS underlay from the core to the access layer switches. The access layer switches function as VTEPs that encapsulate and decapsulate the VXLAN traffic. In addition, these devices route and bridge packets in and out of VXLAN tunnels.

Figure 4 on page 10 show a campus fabric: IP Clos network with Juniper EX4300-MP, EX4650, EX9200, QFX 5120, and QFX10000 switches.

**Figure 4: IP Clos Topology**



# How to Configure an IP Clos Fabric for a Campus Network

IN THIS SECTION



## Requirements

This configuration example uses the following devices:

- Two EX9200 or QFX10000 switches as core devices, Software version: Junos OS Release 20.2R3
- Two EX4650 or QFX5120 switches as distribution devices, Software version: Junos OS Release 20.2R3
- Two EX4300-MP switches as the access layer, Software version: Junos OS Release 20.2R3 or Two EX4400 switches, Software version: Junos OS Release 21.1R1.
- One SRX650 security device
- One WAN router
- Juniper Access Points

# Overview

IN THIS SECTION



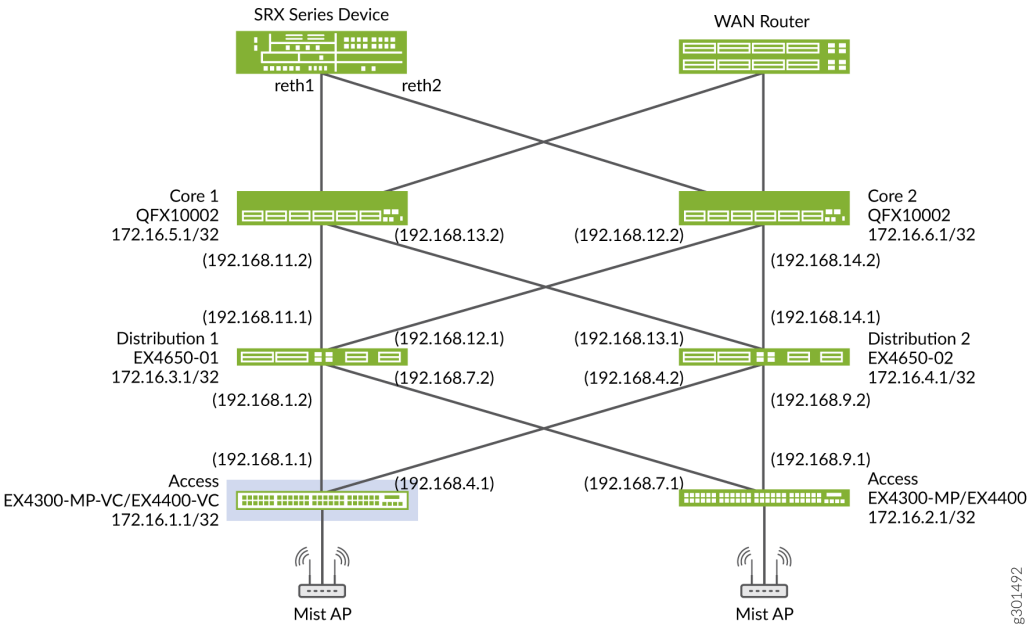
Use this network configuration example to deploy a single campus fabric with a Layer 3 IP-based underlay network that uses EVPN as the control plane protocol and VXLAN as the data plane protocol in the overlay network.

We will first configure EBGP as the underlay routing protocol to exchange loopback routes. Then, we will configure IBGP between the core and distribution devices in the overlay to share reachability information about endpoints in the fabric.

## Topology

In this example, we configure each device with a /32 loopback address. Figure 5 on page 13 shows the physical topology with an SRX series device, WAN router, access layer devices (EX-4300-MP), and it shows the IP addressing scheme that is used in this example. The SRX series router enforces policy rules for transit traffic by controlling traffic flow. It allows traffic that can pass through and denies the traffic that is not permitted based on the security policy that is created.

**Figure 5: EVPN-VXLAN Fabric**




## Configure the Underlay IP Fabric

**IN THIS SECTION**



### Overview

This section shows how to configure the IP fabric underlay on the core, distribution, and access layer switches using EBGP and how to configure the policy rules on the SRX server.

## Interface and Underlay Configuration

IN THIS SECTION



Use this section to configure the underlay on the core and distribution layer switches.

### Core 1 Configuration

### Step-by-Step Procedure

1. Configure the interfaces connected to the core devices.

```
set interfaces xe-0/0/3:0 unit 0 family inet address 192.168.11.2/24
set interfaces xe-0/0/3:1 unit 0 family inet address 192.168.13.2/24
```

2. Configure the loopback interface and router ID and enable per-packet load balancing

```
set interfaces lo0 unit 0 family inet address 172.16.5.1/32
set routing-options router-id 172.16.5.1
set routing-options forwarding-table export ecmp_policy
set policy-options policy-statement ecmp_policy then load-balance per-packet
set policy-options policy-statement ecmp_policy then accept
```

3. Configure the BGP underlay network.

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
from interface lo0.0
```

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
then accept
set groups UnderlayBGP protocols bgp group underlay-bgp type external
set groups UnderlayBGP protocols bgp group underlay-bgp export underlay-clos-export
set groups UnderlayBGP protocols bgp group underlay-bgp local-as 4200000005
set groups UnderlayBGP protocols bgp group underlay-bgp multipath multiple-as
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection minimum-
interval 1000
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection multiplier 3
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection session-mode
automatic
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.13.1 peer-as
4200000004
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.11.1 peer-as
4200000003
set apply-groups UnderlayBGP
```

**Core 2 Configuration**

### Step-by-Step Procedure

1. Configure the interfaces connected to the core devices.

```
set interfaces xe-0/0/30:0 unit 0 family inet address 192.168.12.2/24
set interfaces xe-0/0/30:3 unit 0 family inet address 192.168.14.2/24
```

2. Configure the loopback interface and router ID and enable per-packet load balancing.

```
set interfaces lo0 unit 0 family inet address 172.16.6.1/32
set routing-options router-id 172.16.6.1
set policy-options policy-statement ecmp_policy then load-balance per-packet
set policy-options policy-statement ecmp_policy then accept
set routing-options forwarding-table export ecmp_policy
```

3. Configure the BGP underlay network.

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
from interface lo0.0
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
```

```
then accept
set groups UnderlayBGP protocols bgp group underlay-bgp type external
set groups UnderlayBGP protocols bgp group underlay-bgp export underlay-clos-export
set groups UnderlayBGP protocols bgp group underlay-bgp local-as 4200000006
set groups UnderlayBGP protocols bgp group underlay-bgp multipath multiple-as
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection minimum-
interval 1000
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection multiplier 3
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection session-mode
automatic
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.14.1 peer-as
4200000004
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.12.1 peer-as
4200000003
set apply-groups UnderlayBGP
```

**Distribution 1 Configuration**

### Step-by-Step Procedure

1. Configure the interconnect interfaces between the two core devices and the connectivity to the distribution switches.

```
set interfaces ge-0/0/12 unit 0 family inet address 192.168.1.2/24
set interfaces ge-0/0/9 unit 0 family inet address 192.168.7.2/24
set interfaces xe-0/0/3 unit 0 family inet address 192.168.11.1/24
set interfaces xe-0/0/4 unit 0 family inet address 192.168.12.1/24
```

2. Configure the loopback interface and router ID.

```
set interfaces lo0 unit 0 family inet address 172.16.3.1/32
set routing-options router-id 172.16.3.1
```

3. Enable per-packet load balancing.

```
set routing-options forwarding-table export ecmp_policy
set policy-options policy-statement ecmp_policy then load-balance per-packet
set policy-options policy-statement ecmp_policy then accept
```

4. Configure the BGP underlay network.

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
from interface lo0.0
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
then accept
set groups UnderlayBGP protocols bgp group underlay-bgp type external
set groups UnderlayBGP protocols bgp group underlay-bgp export underlay-clos-export
set groups UnderlayBGP protocols bgp group underlay-bgp local-as 4200000003
set groups UnderlayBGP protocols bgp group underlay-bgp multipath multiple-as
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection minimum-
interval 1000
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection multiplier 3
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection session-mode
automatic
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.1.1 peer-as
4200000001
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.7.1 peer-as
4200000002
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.11.2 peer-as
4200000005
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.12.2 peer-as
4200000006
set apply-groups UnderlayBGP
```

**Distribution 2 Configuration**

### Step-by-Step Procedure

1. Configure the interconnect interfaces between the two core devices and the connectivity to distribution switches.

```
set interfaces ge-0/0/9 unit 0 family inet address 192.168.9.2/24
set interfaces xe-0/0/3 unit 0 family inet address 192.168.13.1/24
set interfaces xe-0/0/4 unit 0 family inet address 192.168.14.1/24
set interfaces ge-0/0/12 unit 0 family inet address 192.168.4.2/24
```

2. Configure the loopback interface and router ID.

```
set interfaces lo0 unit 0 family inet address 172.16.4.1/32
set routing-options router-id 172.16.4.1
```

3. Enable per-packet load balancing.

```
set routing-options forwarding-table export ecmp_policy
set policy-options policy-statement ecmp_policy then load-balance per-packet
set policy-options policy-statement ecmp_policy then accept
```

4. Configure the BGP underlay network.

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
from interface lo0.0
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
then accept
set groups UnderlayBGP protocols bgp group underlay-bgp type external
set groups UnderlayBGP protocols bgp group underlay-bgp export underlay-clos-export
set groups UnderlayBGP protocols bgp group underlay-bgp local-as 4200000004
set groups UnderlayBGP protocols bgp group underlay-bgp multipath multiple-as
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection minimum-
interval 1000
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection multiplier 3
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection session-mode
automatic
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.9.1 peer-as
4200000002
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.13.2 peer-as
4200000005
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.14.2 peer-as
4200000006
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.4.1 peer-as
4200000001
set apply-groups UnderlayBGP
```

### Access switch 1 Configuration

### Step-by-Step Procedure

1. Specify the interfaces that connect to the distribution switches.

```
set interfaces ge-0/0/1 unit 0 family inet address 192.168.4.1/24
set interfaces ge-0/0/2 unit 0 family inet address 192.168.1.1/24
set interfaces lo0 unit 0 family inet address 172.16.1.1/32
```

2. (Optional) Configure a Virtual Chassis with non-stop routing and bridging for high availability.

```
set groups nsr_nsb system commit synchronize
set groups nsr_nsb chassis redundancy graceful-switchover
set groups nsr_nsb routing-options nonstop-routing
set groups nsr_nsb protocols layer2-control nonstop-bridging
set apply-groups nsr_nsb
set groups vc virtual-chassis preprovisioned
set groups vc virtual-chassis member 0 role routing-engine
set groups vc virtual-chassis member 0 serial-number XR0220140070
set groups vc virtual-chassis member 1 role routing-engine
set groups vc virtual-chassis member 1 serial-number XR0220140059
set groups vc virtual-chassis member 2 role line-card
set groups vc virtual-chassis member 2 serial-number XR0220140068
set apply-groups vc
```

3. Configure the underlay BGP.

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
from interface lo0.0
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
then accept
set groups UnderlayBGP protocols bgp group underlay-bgp type external
set groups UnderlayBGP protocols bgp group underlay-bgp export underlay-clos-export
set groups UnderlayBGP protocols bgp group underlay-bgp local-as 4200000001
set groups UnderlayBGP protocols bgp group underlay-bgp multipath multiple-as
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection minimum-
interval 1000
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection multiplier 3
```

```
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection session-mode
automatic
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.1.2 peer-as
4200000003
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.4.2 peer-as
4200000004
set apply-groups UnderlayBGP
```

**Access Switch 2 Configuration**

### Step-by-Step Procedure

1. Specify the interfaces to connect to distribution switches.

```
set interfaces ge-0/0/1 unit 0 family inet address 192.168.9.1/24
set interfaces ge-0/0/2 unit 0 family inet address 192.168.7.1/24
set interfaces lo0 unit 0 family inet address 172.16.2.1/32
```

2. (Optional) Configure a Virtual Chassis with non-stop routing and bridging for high availability.

```
set groups nsr_nsb system commit synchronize
set groups nsr_nsb chassis redundancy graceful-switchover
set groups nsr_nsb routing-options nonstop-routing
set groups nsr_nsb protocols layer2-control nonstop-bridging
set apply-groups nsr_nsb
set groups vc virtual-chassis preprovisioned
set groups vc virtual-chassis member 0 role routing-engine
set groups vc virtual-chassis member 0 serial-number XR0220190261
set groups vc virtual-chassis member 1 role routing-engine
set groups vc virtual-chassis member 1 serial-number XR0220190270
set apply-groups vc
```

3. Configure the underlay BGP.

```
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
from interface lo0.0
set groups UnderlayBGP policy-options policy-statement underlay-clos-export term loopback
then accept
set groups UnderlayBGP protocols bgp group underlay-bgp type external
```

```
set groups UnderlayBGP protocols bgp group underlay-bgp export underlay-clos-export
set groups UnderlayBGP protocols bgp group underlay-bgp local-as 4200000002
set groups UnderlayBGP protocols bgp group underlay-bgp multipath multiple-as
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection minimum-
interval 1000
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection multiplier 3
set groups UnderlayBGP protocols bgp group underlay-bgp bfd-liveness-detection session-mode
automatic
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.7.2 peer-as
4200000003
set groups UnderlayBGP protocols bgp group underlay-bgp neighbor 192.168.9.2 peer-as
4200000004
set apply-groups UnderlayBGP
```

**NOTE**: If you have additional access layer switches in your network, repeat this configuration procedure for each access switch.

#### SRX Configuration

### Step-by-Step Procedure

1. Configure security settings on the SRX device.

```
set security nat source rule-set trust-to-untrust from zone trust
set security nat source rule-set trust-to-untrust to zone untrust
set security nat source rule-set trust-to-untrust rule source-nat-rule match source-address
0.0.0.0/0
set security nat source rule-set trust-to-untrust rule source-nat-rule then source-nat
interface
set security policies from-zone trust to-zone trust policy trust-to-trust match source-
address any
set security policies from-zone trust to-zone trust policy trust-to-trust match destination-
address any
set security policies from-zone trust to-zone trust policy trust-to-trust match application
any
set security policies from-zone trust to-zone trust policy trust-to-trust then permit
set security policies from-zone trust to-zone untrust policy trust-to-untrust match source-
address any
set security policies from-zone trust to-zone untrust policy trust-to-untrust match
```

```
destination-address any
set security policies from-zone trust to-zone untrust policy trust-to-untrust match
application any
set security policies from-zone trust to-zone untrust policy trust-to-untrust then permit
set security zones security-zone trust host-inbound-traffic system-services ping
set security zones security-zone trust host-inbound-traffic system-services all
set security zones security-zone trust host-inbound-traffic protocols all
set security zones security-zone trust interfaces ge-0/0/1.0
set security zones security-zone trust interfaces ge-0/0/2.0
set security zones security-zone untrust interfaces ge-0/0/4.0 host-inbound-traffic system-
services dhcp
set security zones security-zone untrust interfaces ge-0/0/4.0 host-inbound-traffic system-
services tftp
set interfaces ge-0/0/1 unit 0 family inet address 10.16.5.1/24
set interfaces ge-0/0/2 unit 0 family inet address 10.16.6.1/24
set interfaces ge-0/0/4 unit 0 family inet address 10.204.46.136/20
```

## Configure the Overlay

**IN THIS SECTION**



### Overview

**IN THIS SECTION**

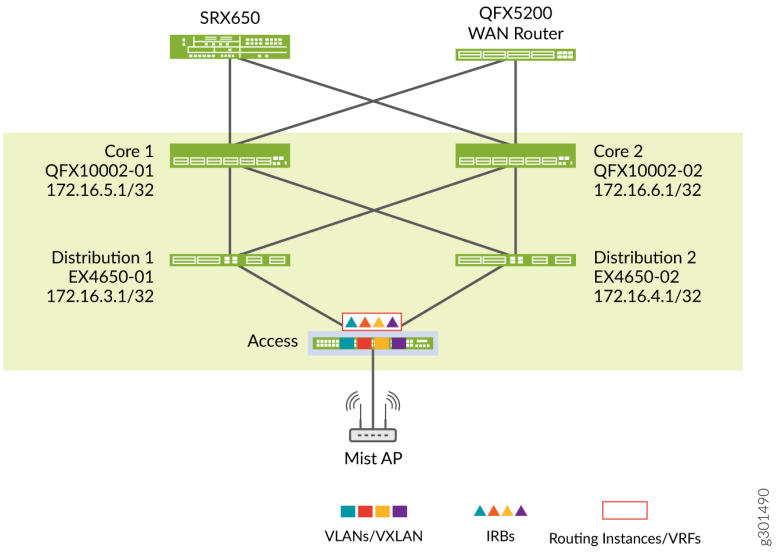
This section shows how to configure the overlay, including configuring IBGP peerings, the VLAN to VXLAN mappings, and the IRB interface configurations for the virtual networks on the access switches.

### Topology

In this example, we have three virtual networks: 1, 2, and 3. The IRB interfaces for these virtual networks are on the access switches. We placed all IRB interfaces in the same routing instance. You can place the IRB interfaces in different routing instances for network segmentation if it is needed in your deployment.

Figure 6 on page 23 shows the overlay virtual network with VLANs.

**Figure 6: Overlay Virtual Network Topology**




## Overlay and Virtual Network Configuration

**IN THIS SECTION**

Use this section to configure the overlay on the core and distribution layer switches.

**Core 1 Configuration**

### Step-by-Step Procedure

1. Set the AS number and configure IBGP neighbors between core and distribution devices. You do not need to configure IBGP neighbors between Core 1 and Core 2 because they receive all BGP updates from Distribution 1 and Distribution 2.

   Configure the core devices as route reflectors to eliminate the need for a full IBGP mesh between all distribution layer switches. This makes the configuration on the distribution layer devices simple and consistent.

```
set routing-options autonomous-system 100
set protocols bgp group cluster-rr type internal
set protocols bgp group cluster-rr local-address 172.16.5.1
set protocols bgp group cluster-rr mtu-discovery
set protocols bgp group cluster-rr family evpn signaling
set protocols bgp group cluster-rr cluster 172.16.5.1
set protocols bgp group cluster-rr multipath
set protocols bgp group cluster-rr bfd-liveness-detection minimum-interval 1000
set protocols bgp group cluster-rr bfd-liveness-detection multiplier 3
set protocols bgp group cluster-rr bfd-liveness-detection session-mode automatic
set protocols bgp group cluster-rr neighbor 172.16.3.1
set protocols bgp group cluster-rr neighbor 172.16.4.1
set protocols bgp group cluster-rr vpn-apply-export
```

#### Core 2 Configuration

### Step-by-Step Procedure

1. Set the AS number and configure IBGP neighbors between core and distribution devices. Configure the core devices as route reflectors to eliminate the need for full mesh IBGP configuration between all distribution and access layer devices.

```
set routing-options autonomous-system 100
set protocols bgp group cluster-rr type internal
set protocols bgp group cluster-rr local-address 172.16.6.1
set protocols bgp group cluster-rr mtu-discovery
set protocols bgp group cluster-rr family evpn signaling
set protocols bgp group cluster-rr cluster 172.16.5.1
set protocols bgp group cluster-rr multipath
set protocols bgp group cluster-rr bfd-liveness-detection minimum-interval 1000
set protocols bgp group cluster-rr bfd-liveness-detection multiplier 3
set protocols bgp group cluster-rr bfd-liveness-detection session-mode automatic
set protocols bgp group cluster-rr neighbor 172.16.3.1
set protocols bgp group cluster-rr neighbor 172.16.4.1
set protocols bgp group cluster-rr vpn-apply-export
```

#### Distribution 1 Configuration

### Step-by-Step Procedure

1. Configure IBGP neighbors from the distribution switch to the core switches.

```
set routing-options autonomous-system 100
set groups OverlayBGP protocols bgp group cluster-pod1 type internal
set groups OverlayBGP protocols bgp group cluster-pod1 local-address 172.16.3.1
set groups OverlayBGP protocols bgp group cluster-pod1 mtu-discovery
set groups OverlayBGP protocols bgp group cluster-pod1 family evpn signaling
set groups OverlayBGP protocols bgp group cluster-pod1 cluster 172.16.3.1
set groups OverlayBGP protocols bgp group cluster-pod1 multipath
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection minimum-
interval 1000
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection multiplier 3
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection session-mode
automatic
```

```
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.1.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.2.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.5.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.6.1
set groups OverlayBGP protocols bgp group cluster-pod1 vpn-apply-export
set apply-groups OverlayBGP
```

**Distribution 2 Configuration**

### Step-by-Step Procedure

1. Configure IBGP neighbors from the distribution switch to the core switches.

```
set routing-options autonomous-system 100
set groups OverlayBGP protocols bgp group cluster-pod1 type internal
set groups OverlayBGP protocols bgp group cluster-pod1 local-address 172.16.4.1
set groups OverlayBGP protocols bgp group cluster-pod1 mtu-discovery
set groups OverlayBGP protocols bgp group cluster-pod1 family evpn signaling
set groups OverlayBGP protocols bgp group cluster-pod1 cluster 172.16.3.1
set groups OverlayBGP protocols bgp group cluster-pod1 multipath
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection minimum-
interval 1000
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection multiplier 3
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection session-mode
automatic
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.1.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.2.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.5.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.6.1
set groups OverlayBGP protocols bgp group cluster-pod1 vpn-apply-export
set apply-groups OverlayBGP
```

### Access 1 Configuration

### Step-by-Step Procedure

1. Configure the overlay BGP.

```
set routing-options router-id 172.16.1.1
set routing-options autonomous-system 100
set groups OverlayBGP protocols bgp group cluster-pod1 type internal
set groups OverlayBGP protocols bgp group cluster-pod1 local-address 172.16.1.1
set groups OverlayBGP protocols bgp group cluster-pod1 mtu-discovery
set groups OverlayBGP protocols bgp group cluster-pod1 family evpn signaling
set groups OverlayBGP protocols bgp group cluster-pod1 multipath
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection minimum-
interval 1000
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection multiplier 3
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection session-mode
automatic
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.3.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.4.1
set groups OverlayBGP protocols bgp group cluster-pod1 vpn-apply-export
set apply-groups OverlayBGP
```

2. Configure EVPN-VXLAN.

```
set switch-options vtep-source-interface lo0.0
set switch-options route-distinguisher 172.16.1.1:10
set switch-options vrf-target target:65000:1111
set switch-options vrf-target auto
set forwarding-options vxlan-routing next-hop 16384
set forwarding-options vxlan-routing interface-num 6144
set forwarding-options vxlan-routing overlay-ecmp
set protocols evpn encapsulation vxlan
set protocols evpn default-gateway no-gateway-community
set protocols evpn remote-ip-host-routes
set protocols evpn extended-vni-list all
```

3. Configure the VLAN/VXLAN mapping and IRB interfaces. VLAN_1 is used to send management traffic from Mist APs to the Internet. Configure VLAN_2 and VLAN_3 to connect wired and wireless client devices

```
set groups EP-VLAN-1-52 vlans EP-VLAN-1 vlan-id 1
set groups EP-VLAN-1-52 vlans EP-VLAN-1 l3-interface irb.1
set groups EP-VLAN-1-52 vlans EP-VLAN-1 vxlan vni 100001
set groups EP-VLAN-1-52 vlans EP-VLAN-2 vlan-id 2
set groups EP-VLAN-1-52 vlans EP-VLAN-2 l3-interface irb.2
set groups EP-VLAN-1-52 vlans EP-VLAN-2 vxlan vni 100002
set groups EP-VLAN-1-52 vlans EP-VLAN-3 vlan-id 3
set groups EP-VLAN-1-52 vlans EP-VLAN-3 l3-interface irb.3
set groups EP-VLAN-1-52 vlans EP-VLAN-3 vxlan vni 100003
set groups EP-VLAN-1-52 interfaces irb unit 1 family inet address 10.0.1.241/24 preferred
set groups EP-VLAN-1-52 interfaces irb unit 1 family inet6 nd6-stale-time 3600
set groups EP-VLAN-1-52 interfaces irb unit 1 family inet6 address 2001:db8::10:0:1:241/112
preferred
set groups EP-VLAN-1-52 interfaces irb unit 2 family inet address 10.0.2.241/24 preferred
set groups EP-VLAN-1-52 interfaces irb unit 2 family inet6 nd6-stale-time 3600
set groups EP-VLAN-1-52 interfaces irb unit 2 family inet6 address 2001:db8::10:0:2:241/112
preferred
set groups EP-VLAN-1-52 interfaces irb unit 3 family inet address 10.0.3.241/24 preferred
set groups EP-VLAN-1-52 interfaces irb unit 3 family inet6 nd6-stale-time 3600
set groups EP-VLAN-1-52 interfaces irb unit 3 family inet6 address 2001:db8::10:0:3:241/112
preferred
set apply-group EP-VLAN-1-52
```

4. Configure the VRF instance.

```
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 instance-type vrf
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface lo0.1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.2
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.3
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 route-distinguisher
172.16.1.1:1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 vrf-target target:100:1
set apply-group VRF-TYPE-2-BD-1-52
```

5. Configure the ports for the Mist Access Points as trunk ports. This allows you to use multiple SSID and VLANs on the port. VLAN_1 is used to send management traffic from Mist APs to the Internet. Configure VLAN_2 and VLAN_3 to connect wired and wireless client devices.

```
set routing-options static route 0.0.0.0/0 next-hop 172.31.25.2
set interfaces mge-0/0/25 mtu
9200
set interfaces mge-0/0/25 unit 0 family ethernet-switching interface-mode trunk
set interfaces mge-0/0/25 unit 0 family ethernet-switching vlan members vlan_1
set poe interface mge-0/0/25
set interfaces irb mtu 9200
```

6. Configure 802.1x authentication for the wired clients.

```
set access profile 802.1X-PROFILE-1 accounting-order radius
set access profile 802.1X-PROFILE-1 authentication-order radius
set access profile 802.1X-PROFILE-1 radius authentication-server 10.204.32.234
set access profile 802.1X-PROFILE-1 radius accounting-server 10.204.32.234
set access profile 802.1X-PROFILE-1 radius options nas-identifier access-switch-1
set access profile 802.1X-PROFILE-1 accounting order radius
set access profile 802.1X-PROFILE-1 accounting address-change-immediate-update
set groups DOT1X protocols dot1x authenticator authentication-profile-name 802.1X-PROFILE-1
set groups DOT1X protocols dot1x authenticator no-mac-table-binding
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 authentication-order
dot1x
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 authentication-order
mac-radius
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 supplicant multiple
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 transmit-period 30
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 mac-radius
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 reauthentication 3600
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 server-timeout 30
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 maximum-requests 10
set apply-groups DOT1X
```

### Access 2 Configuration

### Step-by-Step Procedure

1. Configure the overlay BGP

```
set routing-options router-id 172.16.2.1
set routing-options autonomous-system 100
set groups OverlayBGP protocols bgp group cluster-pod1 type internal
set groups OverlayBGP protocols bgp group cluster-pod1 local-address 172.16.2.1
set groups OverlayBGP protocols bgp group cluster-pod1 mtu-discovery
set groups OverlayBGP protocols bgp group cluster-pod1 family evpn signaling
set groups OverlayBGP protocols bgp group cluster-pod1 multipath
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection minimum-
interval 1000
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection multiplier 3
set groups OverlayBGP protocols bgp group cluster-pod1 bfd-liveness-detection session-mode
automatic
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.3.1
set groups OverlayBGP protocols bgp group cluster-pod1 neighbor 172.16.4.1
set groups OverlayBGP protocols bgp group cluster-pod1 vpn-apply-export
set apply-groups OverlayBGP
```

2. Configure EVPN-VXLAN.

```
set switch-options vtep-source-interface lo0.0
set switch-options route-distinguisher 172.16.1.1:10
set switch-options vrf-target target:65000:1111
set switch-options vrf-target auto
set forwarding-options vxlan-routing next-hop 16384
set forwarding-options vxlan-routing interface-num 6144
set forwarding-options vxlan-routing overlay-ecmp
set protocols evpn encapsulation vxlan
set protocols evpn default-gateway no-gateway-community
set protocols evpn remote-ip-host-routes
set protocols evpn extended-vni-list all
```

3. Configure the VLAN/VXLAN mapping and IRB.

```
set groups EP-VLAN-1-52 vlans EP-VLAN-1 vlan-id 1
set groups EP-VLAN-1-52 vlans EP-VLAN-1 l3-interface irb.1
set groups EP-VLAN-1-52 vlans EP-VLAN-1 vxlan vni 100001
set groups EP-VLAN-1-52 vlans EP-VLAN-2 vlan-id 2
set groups EP-VLAN-1-52 vlans EP-VLAN-2 l3-interface irb.2
set groups EP-VLAN-1-52 vlans EP-VLAN-2 vxlan vni 100002
set groups EP-VLAN-1-52 vlans EP-VLAN-3 vlan-id 3
set groups EP-VLAN-1-52 vlans EP-VLAN-3 l3-interface irb.3
set groups EP-VLAN-1-52 vlans EP-VLAN-3 vxlan vni 100003
set groups EP-VLAN-1-52 interfaces irb unit 1 family inet address 10.0.1.242/24 preferred
set groups EP-VLAN-1-52 interfaces irb unit 1 family inet6 nd6-stale-time 3600
set groups EP-VLAN-1-52 interfaces irb unit 1 family inet6 address 2001:db8::10:0:1:242/112
preferred
set groups EP-VLAN-1-52 interfaces irb unit 2 family inet address 10.0.2.242/24 preferred
set groups EP-VLAN-1-52 interfaces irb unit 2 family inet6 nd6-stale-time 3600
set groups EP-VLAN-1-52 interfaces irb unit 2 family inet6 address 2001:db8::10:0:2:242/112
preferred
set groups EP-VLAN-1-52 interfaces irb unit 3 family inet address 10.0.3.242/24 preferred
set groups EP-VLAN-1-52 interfaces irb unit 3 family inet6 nd6-stale-time 3600
set groups EP-VLAN-1-52 interfaces irb unit 3 family inet6 address 2001:db8::10:0:3:242/112
preferred
set apply-group EP-VLAN-1-52
```

4. Configure the VRF instance.

```
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 instance-type vrf
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface lo0.1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.2
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.3
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.4
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 route-distinguisher
172.16.2.1:1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 vrf-target target:100:1
set groups VRF-TYPE-2-BD-1-52 interfaces lo0 unit 1 family inet
set apply-group VRF-TYPE-2-BD-1-52
```

5. Configure the ports for the Mist Access Points as trunk ports. This allows you to support multiple SSID and VLANs on the port. VLAN_1 is used to send management traffic from Mist APs to the Internet. Configure VLAN_2 and VLAN_3 to connect wired and wireless client devices.

```
set routing-options static route 0.0.0.0/0 next-hop 172.31.25.2
set interfaces mge-0/0/25 mtu
9200
set interfaces mge-0/0/25 unit 0 family ethernet-switching interface-mode trunk
set interfaces mge-0/0/25 unit 0 family ethernet-switching vlan members vlan_1
set poe interface mge-0/0/25
set interfaces irb mtu 9200
```

6. Configure 802.1x authentication for the wired clients.

```
set access profile 802.1X-PROFILE-1 accounting-order radius
set access profile 802.1X-PROFILE-1 authentication-order radius
set access profile 802.1X-PROFILE-1 radius authentication-server 10.204.32.234
set access profile 802.1X-PROFILE-1 radius accounting-server 10.204.32.234
set access profile 802.1X-PROFILE-1 radius options nas-identifier access-switch-2
set access profile 802.1X-PROFILE-1 accounting order radius
set access profile 802.1X-PROFILE-1 accounting address-change-immediate-update
set groups DOT1X protocols dot1x authenticator authentication-profile-name 802.1X-PROFILE-1
set groups DOT1X protocols dot1x authenticator no-mac-table-binding
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 authentication-order
dot1x
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 authentication-order
mac-radius
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 supplicant multiple
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 transmit-period 30
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 mac-radius
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 reauthentication 3600
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 server-timeout 30
set groups DOT1X protocols dot1x authenticator interface mge-1/0/44.0 maximum-requests 10
set apply-groups DOT1X
```

**NOTE**: If you have additional access layer switches in your network, repeat this configuration procedure for each access switch.

# Verification

IN THIS SECTION



## Procedure

### Step-by-Step Procedure

## Overview

Log in to each device and verify that the EVPN-VXLAN fabric has been configured.

## Verification

IN THIS SECTION

**Distribution 1: Verifying BGP Sessions**

### Purpose

Verify the state of the BGP sessions with the core and access devices.

### Action

Verify that BGP sessions are established with the core devices and access devices. The IP addresses for the core devices are 172.16.5.1 and 172.16.6.1 and the IP addresses for the access devices are 172.16.1.1 and 172.16.2.1

```
user@distribution-1> show bgp summary
Threading mode: BGP I/O
Groups: 3 Peers: 11 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
bgp.evpn.0
                     931        914          0          0          0          0
inet.0
                      23         14          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State|#Active/
Received/Accepted/Damped...
172.16.1.1              100        379        318       0       0       35:40 Establ
  bgp.evpn.0: 273/273/273/0
  default-switch.evpn.0: 5/5/5/0
  __default_evpn__.evpn.0: 0/0/0/0
  VRF-ep-type5-2.evpn.0: 9/9/9/0
172.16.2.1              100        301        357       0       0       35:32 Establ
  bgp.evpn.0: 624/624/624/0
  default-switch.evpn.0: 5/5/5/0
  __default_evpn__.evpn.0: 0/0/0/0
172.16.5.1              100        569        540       0       0       29:53 Establ
  bgp.evpn.0: 17/17/17/0
  default-switch.evpn.0: 14/14/14/0
  __default_evpn__.evpn.0: 1/1/1/0
  VRF-ep-type5-2.evpn.0: 2/2/2/0
172.16.6.1              100        476        541       0       0       30:21 Establ
  bgp.evpn.0: 0/17/17/0
  default-switch.evpn.0: 0/14/14/0
  __default_evpn__.evpn.0: 0/1/1/0
  VRF-ep-type5-2.evpn.0: 0/2/2/0
192.168.1.1      4200000001         77         75       0       0       31:51 Establ
```

```
  inet.0: 2/3/3/0
192.168.2.1      4200000001         84         83       0       0       35:42 Establ
  inet.0: 2/3/3/0
192.168.3.1      4200000001         85         84       0       0       35:43 Establ
  inet.0: 2/3/3/0
192.168.7.1      4200000002         83         86       0       0       35:39 Establ
  inet.0: 2/3/3/0
192.168.8.1      4200000002         83         86       0       0       35:35 Establ
  inet.0: 2/3/3/0
192.168.11.2     4200000005         76         74       0       0       30:04 Establ
  inet.0: 2/5/5/0
192.168.12.2     4200000006         72         72       0       0       30:25 Establ
```

### Meaning

BGP is up on both the distribution and core devices. The IBGP sessions are established with the loopback interfaces of the core and access devices using MP-IBGP with EVPN signaling to form the overlay that exchanges EVPN routes.

#### Distribution 2: Verifying BGP Sessions

### Purpose

Verify the state of the BGP sessions with the core and access devices.

### Action

Verify that BGP sessions are established with the core devices and access devices. The IP addresses for the core devices are 172.16.5.1 and 172.16.6.1 and the IP addresses for the access devices are 172.16.1.1 and 172.16.2.1.

```
user@distribution-2> show bgp summary
Threading mode: BGP I/O
Groups: 3 Peers: 11 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
bgp.evpn.0
                     931        914          0          0          0          0
inet.0
                      26         14          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State|#Active/
Received/Accepted/Damped...
```

```
172.16.1.1              100        390        321        0       0       36:52 Establ
  bgp.evpn.0: 273/273/273/0
  default-switch.evpn.0: 5/5/5/0
  __default_evpn__.evpn.0: 0/0/0/0
  VRF-ep-type5-2.evpn.0: 9/9/9/0
172.16.2.1              100        303        364        0       0       36:32 Establ
  bgp.evpn.0: 624/624/624/0
  default-switch.evpn.0: 5/5/5/0
  __default_evpn__.evpn.0: 0/0/0/0
172.16.5.1              100        555        555        0       0       31:46 Establ
  bgp.evpn.0: 17/17/17/0
  default-switch.evpn.0: 14/14/14/0
  __default_evpn__.evpn.0: 1/1/1/0
  VRF-ep-type5-2.evpn.0: 2/2/2/0
172.16.6.1              100        555        556        0       0       31:49 Establ
  bgp.evpn.0: 0/17/17/0
  default-switch.evpn.0: 0/14/14/0
  __default_evpn__.evpn.0: 0/1/1/0
  VRF-ep-type5-2.evpn.0: 0/2/2/0
192.168.4.1      4200000001         79         77       0       0       33:22 Establ
  inet.0: 2/4/4/0
192.168.5.1      4200000001         88         85       0       0       36:53 Establ
  inet.0: 2/4/4/0
192.168.6.1      4200000001         87         85       0       0       36:49 Establ
  inet.0: 2/4/4/0
192.168.9.1      4200000002         88         91       0       0       36:49 Establ
  inet.0: 2/4/4/0
192.168.10.1     4200000002         88         89       0       0       36:45 Establ
  inet.0: 2/4/4/0
192.168.13.2     4200000005         74         75       0       0       31:48 Establ
  inet.0: 2/2/2/0
192.168.14.2     4200000006         76         77       0       0       31:53 Establ
  inet.0: 2/4/4/0
```

### Meaning

BGP is up on both the distribution and core devices. The IBGP sessions are established with the loopback interfaces of the core and access devices using MP-IBGP with EVPN signaling to form the overlay layer and exchange EVPN routes.

#### Access 1: Verifying EVPN Database Information

### Purpose

Verify that the EVPN database is correctly populated.

### Action

Verify that the EVPN database is installing MAC address information for locally attached hosts and receiving advertisements from the other leaf devices with information about remote hosts.

```
user@access-1> show evpn database
Instance: default-switch
VLAN  DomainId  MAC address        Active source                  Timestamp        IP address
     100001     0c:59:9c:ea:22:dc  172.16.2.1                     Dec 14 01:55:44  10.0.1.242

2001:db8::10:0:1:242

fe80::e59:9c00:1ea:22dc
     100001     94:bf:94:8e:9b:6d  irb.1                          Dec 13 04:48:00  10.0.1.241

2001:db8::10:0:1:241

fe80::96bf:9400:18e:9b6d
     100001     ba:04:00:00:00:01  mge-0/0/44.0                   Dec 14 20:11:42
10.0.1.1
     100001     ba:04:00:00:00:02  mge-0/0/44.0                   Dec 14 20:11:42  10.0.1.2
     100003     94:bf:94:8e:9b:6d  irb.3                          Dec 13 04:48:00  10.0.3.241

2001:db8::10:0:3:241
     100003     ba:04:01:00:00:01  mge-0/0/44.0                   Dec 14 20:11:43
10.0.3.1

...
```

#### Access 1: Verifying Local Switching Table Information

### Purpose

Verify that the local switching table is correctly populated. For this example, we are interested in the devices and routes for VLAN_2.

### Action

Verify that the local switching table is installing MAC address information for locally attached hosts and receiving advertisements from the other leaf devices with information about remote hosts.

```
user@access-1> show ethernet-switching table vlan-name EP-VLAN-1
MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static
           SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)


Ethernet switching table : 201 entries, 201 learned
Routing instance : default-switch
   Vlan                MAC                 MAC      Logical                SVLBNH/      Active
   name                address             flags    interface              VENH Index   source
   EP-VLAN-1           0c:59:9c:ea:22:dc   D        vtep.32770
172.16.2.1
   EP-VLAN-1           ba:04:00:00:00:01   D        mge-0/0/44.0
   EP-VLAN-1           ba:04:00:00:00:02   D        mge-0/0/44.0
```

```
user@access-1> show ethernet-switching table vlan-name EP-VLAN-2
MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static
           SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)
Ethernet switching table : 201 entries, 201 learned
Routing instance : default-switch
   Vlan                MAC                 MAC      Logical                SVLBNH/      Active
   name                address             flags    interface              VENH Index   source
   EP-VLAN-2           0c:59:9c:ea:22:dc   D        vtep.32770
172.16.2.1
   EP-VLAN-2           bc:04:00:00:00:01   D        vtep.32770
172.16.2.1
   EP-VLAN-2           bc:04:00:00:00:02   D        vtep.32770
172.16.2.1
```

### Meaning

The output above confirms that the local switching table is correctly learning and installing MAC addresses for all endpoints. It shows the relationship between MAC addresses, the VLANs that they are associated with (VLANs 1, 2, and 3), and their next-hop interface.

#### Access 2: Verifying EVPN Database Information

### Purpose

Verify that the EVPN database is correctly populated.

### Action

Verify that the EVPN database is installing MAC address information for locally attached hosts and receiving advertisements from the other leaf devices with information about remote hosts.

```
user@access-2> show evpn database
Instance: default-switch
VLAN  DomainId  MAC address        Active source                  Timestamp        IP address
 100002     0c:59:9c:ea:22:dc  irb.2                          Dec 14 09:31:03  10.0.2.242

2001:db8::10:0:2:242
     100002     94:bf:94:8e:9b:6d  172.16.1.1                     Dec 14 09:52:33  10.0.2.241

2001:db8::10:0:2:241
     100002     bc:04:00:00:00:01  mge-0/0/44.0                   Dec 15 04:12:22  10.0.2.1
     100002     bc:04:00:00:00:02  mge-0/0/44.0                   Dec 15 04:12:22  10.0.2.2
     100004     0c:59:9c:ea:22:dc  irb.4                          Dec 14 09:31:03  10.0.4.242

2001:db8::10:0:4:242
     100004     94:bf:94:8e:9b:6d  172.16.1.1                     Dec 14 09:52:33  10.0.4.241

2001:db8::10:0:4:241
     100004     bc:04:01:00:00:01  mge-0/0/44.0                   Dec 15 04:12:23  10.0.4.1
     100004     bc:04:01:00:00:02  mge-0/0/44.0                   Dec 15 04:12:23  10.0.4.2
```

#### Access 2: Verifying Local Switching Table Information

### Purpose

Verify that the local switching table has been populated correctly.

### Action

Verify that the local switching table is installing MAC address information for locally attached hosts and receiving advertisements from the other leaf devices with information about remote hosts.

```
user@access-2> show ethernet-switching table vlan-name EP-VLAN-2
MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static
           SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)


Ethernet switching table : 201 entries, 201 learned
Routing instance : default-switch
   Vlan                MAC                 MAC      Logical                SVLBNH/      Active
   name                address             flags    interface              VENH Index   source
   EP-VLAN-2           94:bf:94:8e:9b:6d   D        vtep.32769
172.16.1.1
   EP-VLAN-2           bc:04:00:00:00:01   D        mge-0/0/44.0
   EP-VLAN-2           bc:04:00:00:00:02   D        mge-0/0/44.0
```

### Meaning

The output above confirms that the local switching table is correctly learning and installing MAC addresses for all endpoints associated with VLAN_2. It shows the relationship between MAC addresses, VLANs that they are associated with , and their next-hop interface.

# How to Configure DHCP in an IP Clos Network

**IN THIS SECTION**

## Requirements

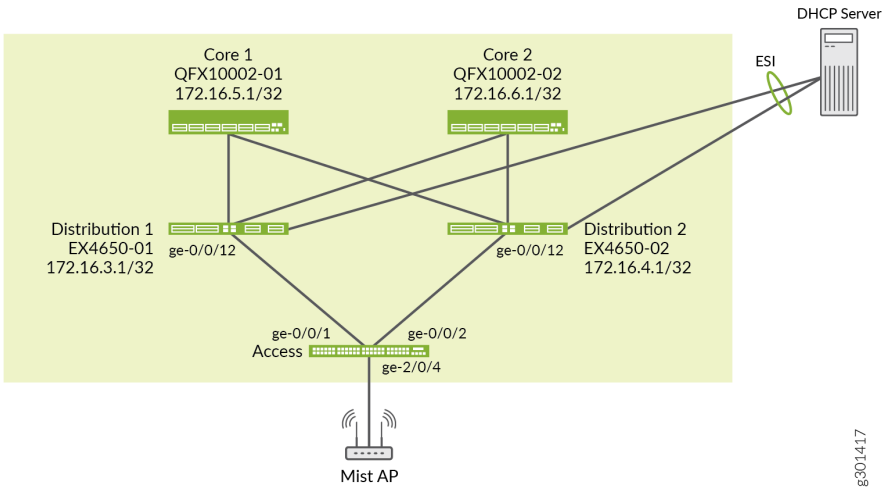
Configure DHCP on the following devices that you configured in the "How to Configure an IP Clos Fabric for a Campus Network" on page 11 configuration example. This configuration example uses the following device:

- An EX4600 switch running Junos OS Release 20.2R3 as a DCHP server. The DHCP server may be an external device in your network.

## Overview

Use this section to configure DHCP on the network. To avoid flooding the network with DHCP discover packets, configure DHCP on an interface in a VRF routing instance. Both distribution devices are configured with EVPN multihoming (also called ESI-LAG) to act as a DHCP relay to a Layer 3 reachable external DHCP server. This provides redundant connectivity to maintain DHCP services if one of the distribution devices fail. The link between the server and the distribution layer is an ESI. shows the virtual network topology with a DHCP server. Figure 7 on page 41 shows the virtual network topology with a DHCP server.

**Figure 7: Overlay Virtual Network Topology with a DHCP Server**



# Configuration

IN THIS SECTION



## CLI Quick Configuration

## DHCP Relay Configuration for Access Switches.

### Step-by-Step Procedure

1. On the access switch, configure DHCP Relay.

```
set groups dhcp-relay forwarding-options dhcp-relay forward-only
set groups dhcp-relay forwarding-options dhcp-relay forward-only-replies
set groups dhcp-relay forwarding-options dhcp-relay group test interface vme.0 exclude
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
group v6_relay active-server-group v6_server_group
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
group v6_relay forward-only
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
group v6_relay interface irb.1
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
group v6_relay interface irb.2
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
group v6_relay interface vme.0 exclude
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
group v6_relay interface em0.0 exclude
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay dhcpv6
server-group v6_server_group abcd::80:01:01:02
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay forward-
only
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay forward-
only-replies
```

```
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay server-
group server_group_1 172.16.38.1
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 active-server-group server_group_1
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 forward-only
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 route-suppression destination
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 interface irb.1
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 interface irb.2
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 interface em0.0 exclude
set groups dhcp-relay routing-instances VRF-ep-type2-1 forwarding-options dhcp-relay group
dhcp_relay_1 interface vme.0 exclude
set apply-groups dhcp-relay
```

2. On the access switch, verify that the route to the DHCP server is reachable via an EVPN type 5 route.

```
user@access-1>show route table VRF-ep-type2-1.evpn.0

VRF-ep-type2-1.evpn.0: 13 destinations, 17 routes (13 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

5:172.16.3.1:3001::0::172.16.38.0::24/248
                   *[BGP/170] 01:41:21, localpref 100, from 172.16.3.1
                      AS path: I, validation-state: unverified
                    >  to 192.168.3.2 via ge-2/0/2.0
                    [BGP/170] 01:41:24, localpref 100, from 172.16.4.1
                      AS path: I, validation-state: unverified
                    >  to 192.168.3.2 via ge-2/0/2.0
5:172.16.4.1:3001::0::172.16.38.0::24/248
                   *[BGP/170] 01:41:24, localpref 100, from 172.16.4.1
                      AS path: I, validation-state: unverified
                       to 192.168.4.2 via ge-0/0/1.0
                    >  to 192.168.6.2 via ge-2/0/1.0
                    [BGP/170] 01:41:21, localpref 100, from 172.16.3.1
                      AS path: I, validation-state: unverified
                       to 192.168.4.2 via ge-0/0/1.0
                    >  to 192.168.6.2 via ge-2/0/1.0
```

```
5:172.16.3.1:3001::0::abcd::80:1:1:0::112/248
                   *[BGP/170] 01:41:21, localpref 100, from 172.16.3.1
                      AS path: I, validation-state: unverified
                    >  to 192.168.3.2 via ge-2/0/2.0
                    [BGP/170] 01:41:24, localpref 100, from 172.16.4.1
                      AS path: I, validation-state: unverified
                    >  to 192.168.3.2 via ge-2/0/2.0
5:172.16.4.1:3001::0::abcd::80:1:1:0::112/248
                   *[BGP/170] 01:41:24, localpref 100, from 172.16.4.1
                      AS path: I, validation-state: unverified
                    >  to 192.168.4.2 via ge-0/0/1.0
                       to 192.168.6.2 via ge-2/0/1.0
                    [BGP/170] 01:41:21, localpref 100, from 172.16.3.1
                      AS path: I, validation-state: unverified
                    >  to 192.168.4.2 via ge-0/0/1.0
                       to 192.168.6.2 via ge-2/0/1.0
```

3. On the access switch, configure VRF support to advertise EVPN type 5 routes to the distribution layer.

```
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 protocols evpn ip-prefix-
routes advertise direct-nexthop
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 protocols evpn ip-prefix-
routes encapsulation vxlan
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 protocols evpn ip-prefix-
routes vni 203001
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 instance-type vrf
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface lo0.1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.1
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.2
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.3
set groups VRF-TYPE-2-BD-1-52 routing-instances VRF-ep-type2-1 interface irb.4
```

4. On the distribution devices, configure an ESI-LAG interface between the distribution devices and the DHCP server.

   *Distribution 1*

```
set interfaces xe-0/0/5 ether-options 802.3ad ae1
set interfaces ae1 esi 00:00:00:ff:00:01:00:01:00:02
```

```
set interfaces ae1 esi all-active
set interfaces ae1 aggregated-ether-options minimum-links 1
set interfaces ae1 aggregated-ether-options lacp active
set interfaces ae1 aggregated-ether-options lacp periodic fast
set interfaces ae1 aggregated-ether-options lacp system-id 00:00:00:99:99:01
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk
set interfaces ae1 unit 0 family ethernet-switching vlan members v3001
set interfaces irb unit 3001 virtual-gateway-accept-data
set interfaces irb unit 3001 family inet address 172.16.38.2/24 virtual-gateway-address
172.16.38.3
set interfaces irb unit 3001 family inet6 address abcd::80:01:01:03/112 preferred
set interfaces irb unit 3001 family inet6 address abcd::80:01:01:03/112 virtual-gateway-
address abcd::80:01:01:01
set interfaces irb unit 3001 virtual-gateway-v4-mac 00:00:5e:00:00:04
set interfaces irb unit 3001 virtual-gateway-v6-mac 00:00:5e:00:00:04
set vlans v3001 vlan-id 3001
set vlans v3001 l3-interface irb.3001
set vlans v3001 vxlan vni 103001
set groups type5-dhcp-server interfaces lo0 unit 3001 family inet
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 routing-options rib VRF-ep-
type5-2.inet6.0 multipath
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 routing-options multipath
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 protocols evpn ip-prefix-routes
advertise direct-nexthop
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 protocols evpn ip-prefix-routes
encapsulation vxlan
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 protocols evpn ip-prefix-routes
vni 203001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 instance-type vrf
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 interface lo0.3001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 interface irb.3001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 route-distinguisher
172.16.3.1:3001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 vrf-target target:100:1
set apply-groups dhcp-relay-fwd
```

*Distribution 2*

```
set interfaces xe-0/0/5 ether-options 802.3ad ae1
set interfaces ae1 esi 00:00:00:ff:00:01:00:01:00:02
set interfaces ae1 esi all-active
set interfaces ae1 aggregated-ether-options minimum-links 1
```

```
set interfaces ae1 aggregated-ether-options lacp active
set interfaces ae1 aggregated-ether-options lacp periodic fast
set interfaces ae1 aggregated-ether-options lacp system-id 00:00:00:99:99:01
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk
set interfaces ae1 unit 0 family ethernet-switching vlan members v3001
set interfaces irb unit 3001 virtual-gateway-accept-data
set interfaces irb unit 3001 family inet address 172.16.38.4/24 virtual-gateway-address
172.16.38.3
set interfaces irb unit 3001 family inet6 address abcd::80:01:01:04/112 preferred
set interfaces irb unit 3001 family inet6 address abcd::80:01:01:04/112 virtual-gateway-
address abcd::80:01:01:01
set interfaces irb unit 3001 virtual-gateway-v4-mac 00:00:5e:00:00:04
set interfaces irb unit 3001 virtual-gateway-v6-mac 00:00:5e:00:00:04
set vlans v3001 vlan-id 3001
set vlans v3001 l3-interface irb.3001
set vlans v3001 vxlan vni 103001
set groups type5-dhcp-server interfaces lo0 unit 3001 family inet
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 routing-options rib VRF-ep-
type5-2.inet6.0 multipath
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 routing-options multipath
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 protocols evpn ip-prefix-routes
advertise direct-nexthop
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 protocols evpn ip-prefix-routes
encapsulation vxlan
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 protocols evpn ip-prefix-routes
vni 203001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 instance-type vrf
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 interface lo0.3001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 interface irb.3001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 route-distinguisher
172.16.4.1:3001
set groups type5-dhcp-server routing-instances VRF-ep-type5-2 vrf-target target:100:1
set apply-groups type5-dhcp-server
```

5. Finally, we configure the EX4600 switch to act as the DHCP server.

```
set groups DHCP_SERVER system services dhcp-local-server dhcpv6 group v6group interface
irb.3001
set groups DHCP_SERVER system services dhcp-local-server group DHCP_Server interface irb.3001
set groups DHCP_SERVER system services dhcp-local-server group DHCP_Server interface em0.0
exclude
set groups DHCP_SERVER interfaces irb unit 3000 family inet address 200.11.184.10/24 preferred
```

```
set groups DHCP_SERVER routing-options rib inet6.0 static route 2001:db8::10:0:2:0/112 next-
hop abcd::80:1:1:1
set groups DHCP_SERVER routing-options static route 10.0.2.0/24 next-hop 172.16.38.3
set groups DHCP_SERVER routing-options static route 10.0.1.0/24 next-hop 172.16.38.3
set groups DHCP_SERVER routing-options static route 172.16.1.0/24 next-hop 172.16.38.3
set groups DHCP_SERVER access address-assignment pool p1 family inet network 10.0.1.0/24
set groups DHCP_SERVER access address-assignment pool p1 family inet range p1_range low
10.0.1.10
set groups DHCP_SERVER access address-assignment pool p1 family inet range p1_range high
10.0.1.220
set groups DHCP_SERVER access address-assignment pool p1 family inet dhcp-attributes maximum-
lease-time infinite
set groups DHCP_SERVER access address-assignment pool p1 family inet dhcp-attributes router
10.0.1.254
set groups DHCP_SERVER access address-assignment pool p2 family inet network 10.0.2.0/24
set groups DHCP_SERVER access address-assignment pool p2 family inet range p2_range low
10.0.2.10
set groups DHCP_SERVER access address-assignment pool p2 family inet range p2_range high
10.0.2.220
set groups DHCP_SERVER access address-assignment pool p2 family inet dhcp-attributes maximum-
lease-time infinite
set groups DHCP_SERVER access address-assignment pool p2 family inet dhcp-attributes router
10.0.2.254
set groups DHCP_SERVER access address-assignment pool v6-p1 family inet6 range v6_p1_range
low 2001:db8::10:0:2:10/112
set groups DHCP_SERVER access address-assignment pool v6-p1 family inet6 range v6_p1_range
high 2001:db8::10:0:2:100/112
```

6. On the EX4600 switch acting as the DHCP server, verify that the DHCP clients are successfully being assigned IP addresses.

```
user@ex4600-03> show dhcp server binding

IP address        Session Id  Hardware address   Expires     State      Interface
10.0.2.10         2           00:10:94:00:3c:f1  537735079   BOUND      irb.3001
10.0.2.11         5           00:10:94:00:3c:f2  537735079   BOUND      irb.3001
10.0.2.12         6           00:10:94:00:3c:f3  537735079   BOUND      irb.3001
10.0.2.13         7           00:10:94:00:3c:f4  537735079   BOUND      irb.3001
10.0.2.14         8           00:10:94:00:3c:f5  537735079   BOUND      irb.3001
10.0.2.15         9           00:10:94:00:3c:f6  537735079   BOUND      irb.3001
10.0.2.16         10          00:10:94:00:3c:f7  537735079   BOUND      irb.3001
10.0.2.17         11          00:10:94:00:3c:f8  537735079   BOUND      irb.3001
```

```
10.0.2.18         12          00:10:94:00:3c:f9  537735079   BOUND      irb.3001
10.0.2.19         13          00:10:94:00:3c:fa  537735079   BOUND      irb.3001
```