

**BP2I – Project LAN DC - VXLAN**

Low-Level Design Document

January 11, 2021

Version 1.2

**Cisco Systems, Inc.  
Corporate Headquarters  
170 West Tasman Drive  
San Jose, CA 95134-1706 USA**[**http://www.cisco.com**](http://www.cisco.com) **Tel: 408 526-4000 Toll Free: 800 553-NETS (6387)  
Fax: 408 526-4100**

Contents

[Contents 2](#_Toc61272694)

[List of Figures and Tables 4](#_Toc61272695)

[About This Low-Level Design Document 5](#_Toc61272696)

[History 5](#_Toc61272697)

[Review 5](#_Toc61272698)

[Document Conventions 5](#_Toc61272699)

[1 Introduction 7](#_Toc61272700)

[1.1 Document Purpose 7](#_Toc61272701)

[1.2 Project Overview 7](#_Toc61272702)

[1.3 Audience 7](#_Toc61272703)

[1.4 Scope 7](#_Toc61272704)

[1.5 Assumptions 7](#_Toc61272705)

[1.6 Related Documents 8](#_Toc61272706)

[1.7 Document scope 8](#_Toc61272707)

[2 Naming Convention 9](#_Toc61272708)

[3 Physical Design Overview 10](#_Toc61272709)

[3.1 Leaf to Spine Connections 10](#_Toc61272710)

[3.2 Leaf to Leaf (VPC Peer-link), Leaf to Servers, Border/service leaf to Firewalls and Gateways Connection 10](#_Toc61272711)

[3.3 Inter Site Connections 12](#_Toc61272712)

[3.3.1 Small Template 12](#_Toc61272713)

[3.3.2 Medium Template 13](#_Toc61272714)

[3.3.3 Large Template 13](#_Toc61272715)

[3.4 Server Connections 14](#_Toc61272716)

[3.5 Services Connections 15](#_Toc61272717)

[3.5.1 Firewall Connections 15](#_Toc61272718)

[3.5.2 Routers Connections 16](#_Toc61272719)

[3.5.3 Load Balancer Connections 16](#_Toc61272720)

[4 VXLAN Design 18](#_Toc61272721)

[4.1 Underlay 18](#_Toc61272722)

[4.1.1 IP Addressing 18](#_Toc61272723)

[4.1.2 Point-to-Point Interfaces 19](#_Toc61272724)

[4.1.3 MTU Considerations and Provisioning 20](#_Toc61272725)

[4.1.4 IP Unicast Routing IS-IS 21](#_Toc61272726)

[4.1.5 Multi-destination traffic 22](#_Toc61272727)

[4.1.6 BFD 25](#_Toc61272728)

[4.2 Overlay 26](#_Toc61272729)

[4.2.1 VXLAN BGP EVPN 27](#_Toc61272730)

[4.2.2 Vxlan Multi-site 31](#_Toc61272731)

[4.2.3 ARP Suppression 35](#_Toc61272732)

[4.2.4 NGOAM 35](#_Toc61272733)

[4.3 VPC VTEPs 36](#_Toc61272734)

[4.3.1 VPC Best Practices 36](#_Toc61272735)

[4.3.2 VPC Considerations for VXLAN EVPN: 38](#_Toc61272736)

[4.3.3 Configuring VPC 38](#_Toc61272737)

[4.3.4 Timeline for vPC recovery 39](#_Toc61272738)

[4.3.5 VXLAN Infra Vlan 40](#_Toc61272739)

[4.4 Classical Ethernet & MST 41](#_Toc61272740)

[4.5 DHCP Relay in VXLAN 41](#_Toc61272741)

[4.6 External Connectivity 43](#_Toc61272742)

[4.6.1 External Connectivity with Static Routing 43](#_Toc61272743)

[4.6.2 External Connectivity with eBGP Routing 44](#_Toc61272744)

[4.7 Service Models 46](#_Toc61272745)

[4.7.1 End Host with Anycast Gateway (AG) 46](#_Toc61272746)

[4.7.2 Service Model with Default Gateway on FW 46](#_Toc61272747)

[4.8 L4-L7 Integration 46](#_Toc61272748)

[4.8.1 Connection with eBGP 47](#_Toc61272749)

[4.8.2 Connection with Static Routing 51](#_Toc61272750)

[4.9 Fabric Management 53](#_Toc61272751)

[4.9.1 Out-of-Band Network 53](#_Toc61272752)

[4.9.2 SSH/Telnet 54](#_Toc61272753)

[4.9.3 NX-OS Programmability & Telemetry 54](#_Toc61272754)

[4.9.4 Authentication, Authorization and accounting (AAA) 56](#_Toc61272755)

[4.9.5 Simple Network Management Protocol (SNMP) 57](#_Toc61272756)

[4.9.6 Logging 57](#_Toc61272757)

[4.9.7 Network Time Protocol (NTP) 58](#_Toc61272758)

[4.10 Security features 58](#_Toc61272759)

[4.10.1 Preventing VLAN Hopping 58](#_Toc61272760)

[4.10.2 Control plane security and protection 59](#_Toc61272761)

[4.10.3 Management security 60](#_Toc61272762)

[4.10.4 Control plane policy 61](#_Toc61272763)

[4.10.5 Data plane security 61](#_Toc61272764)

[4.11 Quality of Service 64](#_Toc61272765)

[4.11.1 Classification at the Ingress VTEP 65](#_Toc61272766)

[4.11.2 Inside the VXLAN Tunnel 66](#_Toc61272767)

[4.11.3 VXLAN to IP 66](#_Toc61272768)

[4.11.4 Decapsulated Packet Priority Selection 67](#_Toc61272769)

[4.12 Fabric Considerations and Preparations 67](#_Toc61272770)

[5 Configuration Templates 68](#_Toc61272771)

[5.1 Access Leaf Configuration Template 68](#_Toc61272772)

[5.2 Border/Service Leaf Configuration Template 77](#_Toc61272773)

[5.3 Spine Configuration Template 83](#_Toc61272774)

[5.4 Super-spine Configuration Template 86](#_Toc61272775)

[6 Appendix A: Acronym Listing or Full Glossary 89](#_Toc61272776)

[7 Appendix B : SFP Modules Matrix Compatibility 91](#_Toc61272777)

[Trademarks and Disclaimers 93](#_Toc61272778)

[Document Acceptance 94](#_Toc61272779)

List of Figures and Tables

[Figure 1 - Spine 9336C-FX2 Leaf 93240YC-FX2 (Leaf to Leaf (VPC Peer-link), Leaf to Servers) 12](#_Toc61272780)

[Figure 2 - Spine 9336C-FX2 Leaf 9336C-FX2 (Border/service leaf to Border/service leaf (VPC Peer-link), Border/service leaf to Firewalls and Gateways) 13](#_Toc61272781)

[Figure 3 - Spine back-to-back connections in stretched small template 13](#_Toc61272782)

[Figure 4 - Spines to Super Spines connection in medium template 14](#_Toc61272783)

[Figure 5 - BGWs to Super Spines connection in large template 15](#_Toc61272784)

[Figure 6 - Server Connections to Access Leaves 16](#_Toc61272785)

[Figure 7 - Active/Standby FWs Connections to Service Leaves 16](#_Toc61272786)

[Figure 8 - Edge Routers Connections to Border Leaves 17](#_Toc61272787)

[Figure 9 - Load Balancers Connections to Service Leaves 18](#_Toc61272788)

[Figure 10 - P2P Link between Leaf1 and Spine1 21](#_Toc61272789)

[Figure 11 - Timeline for vPC recovery 41](#_Toc61272790)

[Figure 12 - Active/Standby Exit Points 46](#_Toc61272791)

[Figure 13 - Firewall interconnection with BGP 48](#_Toc61272792)

[Figure 14 - Firewall interconnection logical diagram 48](#_Toc61272793)

[Figure 15 - Static Routing with Firewalls on different VPC VTEPs 52](#_Toc61272794)

[Figure 16 - Static Routing with Firewalls on different VPC VTEPs (after FW failover) 53](#_Toc61272795)

[Figure 17 - Copy of DSCP from L3 packet to VXLAN Outer Header 66](#_Toc61272796)

[Figure 18 - Copy of CoS from L2 packet to VXLAN Outer Header 66](#_Toc61272797)

[Figure 19 - Marking Inside of the VXLAN Tunnel 67](#_Toc61272798)

[Table 1 – Underlay Platforms 11](#_Toc61272799)

[Table 2 – Leaf Uplinks 11](#_Toc61272800)

[Table 3 – Leaf vPC links 12](#_Toc61272801)

[Table 4 – End Points connections to VXLAN Fabric 18](#_Toc61272802)

[Table 5 – Configuration snippets (elements) 19](#_Toc61272803)

[Table 6 – P2P /31 Links 20](#_Toc61272804)

[Table 7 – MTU Recommendations 21](#_Toc61272805)

[Table 8 – PIM Anycast RP Placement 25](#_Toc61272806)

[Table 9 – EVPN BGP neighborships in each template 28](#_Toc61272807)

[Table 10 – BPDU Filtering Configurations 64](#_Toc61272808)

[Table 11 – Default Settings for VXLAN QOS 67](#_Toc61272809)

About This Low-Level Design Document

|  |  |
| --- | --- |
| Author | Ayman Hariz, Bruno Rahme, Frederic Fratta |
| Change Authority | Professional Services, Customer Experience, Cisco |
| DCP Reference | 1334521 |
| Project ID | 919957 |

History

|  |  |  |  |
| --- | --- | --- | --- |
| Version No. | Issue Date | Status | Reason for Change |
| 0.1 |  | Initial Release | Initial document release |
| 0.2 |  | Pre-Draft | Reworked |
| 0.3 |  | Pre-Draft | After internal review |
| 0.4 |  | Draft | After internal review |
| 0.5 |  | Draft | Add SSL/TLS for NXOS Telemetry |
| 0.65 |  | Draft | Modified after customer review |
| 0.7 |  | Draft | Added Configuration Templates Chapter |
| 0.75 |  | Final | Modified after customer review |
| 0.89 |  | Update | Modified as per SVS lab testing |
| 1.1 |  | Update | Modified as per SVS lab testing |
| 1.2 |  | Update | Review |

Review

|  |  |  |
| --- | --- | --- |
| Reviewer’s Details | Version No. | Date |
| Ayman Hariz, Bruno Rahme | 0.4 | 17/Jul/2020 |
| Ayman Hariz | 0.5 | 22/July/2020 |
| Ayman Hariz, Tural Mammedov | 0.65 | 11/Aug/2020 |
| Ayman Hariz | 0.7 | 18/Aug/2020 |
| Ayman Hariz | 0.75 | 24/Aug/2020 |
| Tural Mammedov, Nicolas Rolin | 0.89 | 13/Nov/2020 |
| Nicolas Rolin, Riadh Habibi, Stephane Romagny | 1.1 | 28/Dec/2020 |
| Stéphane Romagny | 1.2 | 11/Jan/2021 |

Document Conventions

|  |  |
| --- | --- |
|  | Caution—Alerts readers to be careful. In this situation, you might do something that could result in equipment damage or loss of data. |
|  | Note—Alerts readers to take note. Notes contain helpful suggestions or references to material not covered in the document. |
|  | Timesaver—Alerts the reader that they can save time by performing the action described in the paragraph affixed to this icon. |
|  | Tip—Alerts the reader that the information affixed to this icon will help them solve a problem. The information might not be troubleshooting or even an action, but it could be useful information similar to a Timesaver. |
|  | Warning—Alerts readers of a situation that could cause bodily injury. They need to be aware of the hazards involved with electrical circuitry and familiarize themselves with standard practices for preventing accidents. |

# Introduction

## Document Purpose

BNPP has requested Cisco CX Customer Delivery (Cisco CX CD) to create the Low-Level Design for their new and existing Data Center Fabrics based on High-Level Design document as continuation of project delivery.

Cisco CX CD has been engaged for support with the planning, design, lab testing, and implementation support three new design templates for Small, Medium and Large Fabric templates. This document will be the foundation for the development of other documentation like draft JTPoR (Joint Technical Plan of Record), DTP (Detailed Test Plan) and the rest documentation like NMP (Network Migration Plan).

## Project Overview

BNPP has requested Cisco CX Customer Delivery (Cisco CX CD) to provide whole cycle for VLAN EVPN project delivery within Cisco Nexus 9000 Family.

## Audience

This document is intended for members of the Cisco and BNPP architecture and engineering teams who have direct responsibility for the network infrastructure inside the datacenters. In addition, it is advisable that those IT persons responsible for managing servers and application environments hosted inside the datacenters also review this document and provide comments and feedback where applicable.

## Scope

Draft the High-Level Design (HLD) Document, which is limited to the following:

* + New technical objectives and requirements fulfilment
  + Definition of design recommendations
  + Network logical and physical topology
  + IP protocols
  + Switching and routing
  + High-availability platform features/protocols
  + Security infrastructure
  + Hardware platform recommendations
  + Cisco platform configuration templates for the aforementioned protocols and features
  + Software release recommendations based on features and/or functionality

## Assumptions

Assumptions have been taken into consideration during whole project and within all deliverables:

* Design documents (HLD, LLD) will consider 3 scenarios according to fabric size (large, medium, and small templates).
* Design documentation will consider utilization of DCNM (design of DCNM itself is out of scope).
* Design does not cover QoS definition and considers only out for the box QOS (limited to 3 classes of traffic).
* Cisco will create High-Level Migration Plan with a strategy for migration.
* Cisco will create MoP template that will be used by Cisco lab for migration procedures validation.
* MoP template will not cover the whole migration windows, but it will be used to validate the main concept regarding the migration in Cisco lab.
* NMP and MoP template will cover fabric migration from old to new with 2 Spines, 4 Leaves, and 2 Border Leaves in each DC.
* NMP and MoP will cover the use case of migrating existing DFI-B fabric based on N7k/N5k to the new N9k fabric.
* Fabric will be based on N9K family product and on VXLAN EVPN technology only.
* DCNM and NIR used only for telemetry based on this assumption and in leveraging SNMP as well.
* BoM definition and validation will cover the initial deployment for a dedicated fabric for PRES-R with a maximum of 20 switches and made of Spines, Leaves, L2 Border Leaves and L3 Border Leaves.
* Cisco lab will be up to 16 N9k devices and 2 L3 devices to simulate a L3 backbone.
* For migration validation, Cisco lab will include up to 2 N7K, 4 N5k and 4 N2K.
* Tests execution will be carried out with BNPP and from a Cisco office.
* Runbook will cover up to 6 operational use cases agreed with BNP (i.e. Adding new Leaves to Fabric, definition of new network and routing to second site, definition of new local connection to end hosts, Configuring multihoming).
* Runbook will include elementary/daily operations, maintenance and monitoring procedures agreed with Customer during Requirements Gathering Workshop.

## Related Documents

This document and the rest documents of whole delivery refer to the next existing documents:

* Kick-off Power Point Document – filename: 919957 BP2I LAN DC kick off - V0.1.pptx
* Services Proposal Document – filename: 919957\_19866942\_BNP PARIBAS\_BNPP BP2i Project LAN DC\_proposal v7.docx
* High-Level Design Document – filename: DC 2015 BNPP - HLD Intra v1.3
* Low-Level Design Document – filename: BNPP-DC2015\_LLD-DFI-v1.4
* CRD Power Point format – BNPP VXLAN CRD Initial Meeting v0.7.pptx
* CRD Word format – BNPP VXLAN CRD v0.5.docx
* HLD PDF format – BNPP VXLAN HLD v0.85.docx.

## Document scope

The scope of this document will cover the new Next Generation Data Center (NGDC) template-based design architecture. The LLD will include the features and functions that will satisfy the stated technical objectives of the project.

This document will focus on DC blocks, which will be replicated by BNPP in multiple location or sizes, based on the network needs (like number of server ports, security restrictions, and numbers of DC etc.)

The scope of the actual design is to build new three templates (Small, Medium, and Large) and integrate one of them it into the existing network as first step.

# Naming Convention

In this chapter we will have the naming conventions standards for BNPP.

Below you can find the engineering rules that we have defined for the new fabrics.

**{DeviceType}{Location}{Room}-{FabricRole}-{Metier}-{Function}{PodID}{MemberID}**

Example :

RBAN11-PREZ-PFPI-SP01**X**

Device type: W=Switch, R=Router

Location: NORM for Marne Nord DC, EST for Marne Est DC, BAN for Bastogne Nord and BAS for Bastogne Sud.

Room number: 5 or 6 in NORD DC, 11 or 12 in EST DC and Bastogne DCs.

Fabric Role : PREZ, DC, T, CIS, CLD, CYB, LEG

ID Fabric/metier (**optional**): PFPI, ASSU…

Function: BL=Border Leaf, SL=Service Leaf, AL=Access Leaf, SP=Spine, SS=Super Spine

Pod Number: 01-99, unique per fabric.

Member ID: “X” for primary, “Y” for secondary.

# Physical Design Overview

This section describes Cisco Nexus 9000 Hardware used in NGDC Switching Fabric which is based on Cisco Cloud Scale Family. It is very important to understand what capabilities the platform has and align the Design Conception accordingly to avoid any interoperability and compatibility issues between devices and within the single device for features implemented.

The following platforms will be used by BNPP for the Leafs, Border Gateways, Spines and Super Spines in the respective templates.

Table 1 – Underlay Platforms

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Platform | Specifications | Access Leaf | Border/Service Leaf | Border Gateway | Spine | Super Spines |
| Nexus 93240YC-FX2 | 48 x 1/10/25G & 12 x 40/100G |  |  |  |  |  |
| Nexus 93180YC-FX | 48 x 1/10/25G & 6 x 40/100G |  |  |  |  |  |
| Nexus 93108TC-FX | 48 x 100M/1/10GBASE-T & 6 x 40/100 |  |  |  |  |  |
| Nexus 9336C-FX2 | 36 x 25/40/100G |  |  |  |  |  |

## Leaf to Spine Connections

Leaf to Spine connections in all templates will be 100G. Each device form leaf level should be connected to the same port on spine level devices. For example, leaf #1 should be connected to e1/1 ports on each spine, leaf #2 should be connected to e1/2 on each spine, etc. On Leaf switches first available port should be used for uplink.

Depending on the Leaf switch platform, certain interfaces only can be used for 100G uplinks as seen in the table below:

Table 2 – Leaf Uplinks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Leaf Platform | Specifications | 100G Interfaces | Planned uplink ports | Description |
| Nexus 93240YC-FX2 | 48 x 1/10/25G & 12 x 40/100G | e1/49 - 60 | e1/49 - 58 | Maximum 10 spines |
| Nexus 93180YC-FX | 48 x 1/10/25G & 6 x 40/100G | e1/49 - 54 | e1/49 - 52 | Maximum 4 spines |
| Nexus 93108TC-FX | 48 x 100M/1/10GBASE-T & 6 x 40/100 | e1/49 - 54 | e1/49 - 52 | Maximum 4 spines |
| Nexus 9336C-FX2 | 36 x 25/40/100G | e1/1 - 36 | e1/1 – 10 | Maximum 10 spines |

* Last two uplink ports ono Leaf switches will be used for vPC peer-link.

## Leaf to Leaf (VPC Peer-link), Leaf to Servers, Border/service leaf to Firewalls and Gateways Connection

VPC peer-links between Leaf switches will utilize 2x 100G for each VPC pair. Last ports on leaf devices should be used for vPC peer link.

Table 3 – Leaf vPC links

|  |  |  |  |
| --- | --- | --- | --- |
| Leaf Platform | Specifications | 100G Interfaces | Planned vPC ports |
| Nexus 93240YC-FX2 | 48 x 1/10/25G & 12 x 40/100G | e1/49 - 60 | e1/59 - 60 |
| Nexus 93180YC-FX | 48 x 1/10/25G & 6 x 40/100G | e1/49 - 54 | e1/53 - 54 |
| Nexus 93108TC-FX | 48 x 100M/1/10GBASE-T & 6 x 40/100 | e1/49 - 54 | e1/53 - 54 |
| Nexus 9336C-FX2 | 36 x 25/40/100G | e1/1 - 36 | e1/35 - 36 |

Figure 1 - Spine 9336C-FX2 Leaf 93240YC-FX2 (Leaf to Leaf (VPC Peer-link), Leaf to Servers)

A close up of a map

Description automatically generated

* Each server should be dual attached to the VXLAN fabric. Therefore, ports will be mirrors on both leaves where servers will be attached. For example, server#1 will be connected to e1/1 on leaf#1 and e1/1 on leaf#2. In case of single attached server the mirrored port will be reserved on the other leaf (leaf that doesn’t have connection to the single-attached server) and port-channel will be configured (port-channel from server toward leaves and virtual port-channel from vPC pair leaves towards to server).
* VPC peer-links are using last available ports. Interconnection between Leaves and Spines is using first available port (on uplink block) on Leaf to first available port on Spine.

Figure 2 - Spine 9336C-FX2 Leaf 9336C-FX2 (Border/service leaf to Border/service leaf (VPC Peer-link), Border/service leaf to Firewalls and Gateways)

A close up of a map

Description automatically generated

## Inter Site Connections

### Small Template

In the Small Template Stretched fabric, spines from both sites will have 100G back-to-back connections, last available port on Spine DC1 to last available port on Spine DC2, as illustrated in the diagram below:

Figure 3 - Spine back-to-back connections in stretched small template

A picture containing drawing

Description automatically generated

### Medium Template

In the Medium Template, a spine in each site will connect to all super spine switches in all sites using 100G links connection, last available port on Spine to first available port to Super-Spine, as illustrated in the diagram below:

Figure 4 - Spines to Super Spines connection in medium template

A picture containing drawing

Description automatically generated

### Large Template

In the Large Template, Border Gateways in each site will connect to super spine switches using 100G links connection as illustrated in the diagram below:

Figure 5 - BGWs to Super Spines connection in large template

A picture containing text, map

Description automatically generated

## Server Connections

We have two type of server connections to the Access Leaf switches. One type has dual/redundant (VPC) connection to the switches and the other has only a single connection.

Figure 6 - Server Connections to Access Leaves

A screenshot of a cell phone

Description automatically generated

## Services Connections

### Firewall Connections

Active/Standby FWs will be connected through VPC to the VXLAN Fabric. Active/Standby FWs are connected to different VPC Pairs (different VTEPs).

Below diagram is just an example to illustrate the FWs connectivity, number of spines and inter-site connection will be different based on the template used. The interfaces on Service Leaves connecting to FWs can be different as well.

Figure 7 - Active/Standby FWs Connections to Service Leaves

A close up of a map

Description automatically generated

* For Inter-site connection details, please refer to section [3.3](#_Inter_Site_Connections) in this document.

### Routers Connections

Routers will be connected to the VXLAN fabric using full mesh P2P links, they can connect to the same or different VTEPs (different sites)

Below diagram is just an example to illustrate the Routers connectivity, number of spines will be different based on the template used. The interfaces on Border Leaves connecting to Routers can be different as well.

Figure 8 - Edge Routers Connections to Border Leaves

A picture containing text, map

Description automatically generated

### Load Balancer Connections

Load Balancers will be connected through VPC to the VXLAN Fabric. LBs will be connected through VPC to the VXLAN Fabric. LBs are connected to different VPC Pairs (different pair of VTEPs).

Below diagram is just an example to illustrate the LBs connectivity, number of spines and inter-site connection will be different based on the template used. The interfaces on Service Leaves connecting to LBs can be different as well.

Figure 9 - Load Balancers Connections to Service Leaves

A close up of a map

Description automatically generated

* For Inter-site connection details, please refer to section [3.3](#_Inter_Site_Connections) in this document.

The below table summarizes end point connections type and speed to the VXLAN Fabric:

Table 4 – End Points connections to VXLAN Fabric

|  |  |  |  |
| --- | --- | --- | --- |
| Device | Leaf Type | Connection Type | Link Speed |
| Servers/endpoints | Access Leaf | VPC or Orphan Ports | 1/10/25/40G\* |
| Firewalls | Service Leaf | VPC | 1/10/25/40G\* |
| Load Balancers | Service Leaf | VPC | 1/10/25/40G\* |
| External Routers | Border Leaf | 2x Single P2P to each VTEP | 1/10/25/40G\* |

\* For 40G connection links, Nexus 9336C-FX2 switches should be used.

# VXLAN Design

This chapter describes in detail the proposed solution design for all the templates and demonstrates the difference between the templates in perspective of technologies used.

* All references regarding configuration and scalabilities in this chapter refer to Cisco Nexus 9000 Series NX-OS Verified Scalability Guide, Release 9.3(3) and Cisco Nexus 9000 Series NX-OS VXLAN Configuration guide, Release 9.3(x).

All templates have common configuration snippets, but for example Large Fabric template requires specific configuration for BGW based on Cisco VXLAN EVPN Multi-site Solution. Small and Medium templates are common and utilize Single Fabric deployment design. To apply any Fabric template for specific network block it is mandatory determining specific requirements and apply appropriate configuration snippets listed in table below. Not all the Configuration snippets (elements) are mandatory.

Table 5 – Configuration snippets (elements)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Number | Configuration snippet / element | Chapter | Template | | |
| **Small** | **Medium** | **Large** |
| **1** | **Underlay** | 4.1 |  |  |  |
| 1.1 | IP Addressing and planning | 4.1.1 | + | + | + |
| 1.2 | Point to Point Interfaces | 4.1.2 | + | + | + |
| 1.3 | MTU Considerations and provisioning | 4.1.3 | + | + | + |
| 1.4 | IP Unicast Routing (IS-IS) | 4.1.4 | + | + | + |
| 1.5 | Multi destination traffic | 4.1.5 | + | + | + |
| 1.6 | BFD | 4.1.6 | Optional | Optional | Optional |
| **2** | **Overlay** | 4.2 |  |  |  |
| 2.1 | VXLAN BGP EVPN | 4.2.1 | + | + | + |
| 2.2 | VXLAN Multisite | 4.2.2 |  |  | + |
| 2.3 | ARP Supression | 4.2.3 | + | + | + |
| 2.4 | NGOAM | 4.2.4 | + | + | + |
| 2.5 | VPC VTEPs | 4.3 | + | + | + |
| 2.6 | Classical Ethernet & MST | 4.4 | + | + | + |
| 2.8 | DHCP Relay | 4.5 | + | + | + |
| **3** | **External Connectivity** | 4.6 | + | + | + |
| **4** | **Service Models** | 4.7 | + | + | + |
| **5** | **L4-L7 Integration** | 4.8 | + | + | + |
| **6** | **Fabric Management** | 4.9 | + | + | + |
| **7** | **Security Features** | 4.10 | + | + | + |
| **8** | **Quality of Service** | 4.11 | + | + | + |

## Underlay

The primary purpose of the underlay in the VXLAN EVPN fabric is to advertise the reachability of Virtual Tunnel End Points (VTEPs) and BGP peering addresses.

### IP Addressing

The VXLAN IP fabric underlay supports IPv4 address family and IPv6 Underlay (VXLANv6) starting from NX-OS 9.3(1). IPv6 Underlay (VXLANv6) has restrictions which are listed in [Chapter: Configuring VXLAN with IPv6 in the Underlay (VXLANv6)](https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/93x/vxlan/configuration/guide/b-cisco-nexus-9000-series-nx-os-vxlan-configuration-guide-93x/b-cisco-nexus-9000-series-nx-os-vxlan-configuration-guide-93x_chapter_010011.html#id_106255) of Cisco Nexus 9000 Series NX-OS VXLAN Configuration Guide, Release 9.3(x).

As requirements and preferences of BNPP teams Underlay will utilize IPv4 address family in early deployments and will consider IPv6 Underlay in future. Point-to-point (P2P) numbered network links will be used.

For each point-to-point link, as an example between the leaf switch nodes and the spine switch nodes, BNPP will use a /31 mask. An IS-IS point-to-point numbered network is only between two switches (interfaces), and there is no need for a broadcast or network address. So, a /31 network will suffice for all fabric templates.

This IP addressing approach requires to reserve the following IP addresses:

* Loopback 0, this loopback interface is required on every network element within fabric for IGP’s router ID. Technically it will not be used by IGP for any purposes, there is only match between router ID and IP address of this loopback.
* Loopback 1, this loopback utilizes one IP address as Virtual IP address (VIP) shared between two vPC peers and one Physical IP address (PIP) as unique VTEP identifier. This interface is required for all Leaves and BGWs (BGWs IP addressing will be considered in detail later in this document).
* Loopback <y> & <y+1>, these loopbacks are used for RP Anycast mode on set of Spines/SuperSpines if multicast is the mechanism of BUM traffic delivery.
* Interface vlan <x>, this Infra VLAN for Underlay backup routing over Peer Link between vPC Peers.
* In this document, examples of assigning IP addresses will be used, but those examples do not represent the IP addressing that will be used in the actual deployment.

### Point-to-Point Interfaces

BNPP will use /31 subnets for the underlay interconnections. Each link will have a unique /31 subnet with one IP addresses assigned to each side of the link.

The below table shows where the /31 subnet can be used:

Table 6 – P2P /31 Links

|  |  |  |
| --- | --- | --- |
| Device A | Device B | Template |
| Access Leaf, Border Leaf, Service Leaf | Spine | All Templates |
| Spine site A | Spine site B | Small Template Stretched Fabric |
| Spine | Super Spine | Medium Template |
| Spine | Border Gateway | Large Template |
| Border Gateway | Super Spine | Large Template |
| Super Spine Site A | Super Spine Site B | Large Template |

Below is an example of /31 P2P link configuration between a Leaf and a Spine.

***Configuration 1*– P2P Configuration**

**! Configuration on Leaf Interface**

LEAF1(config)#Interface e1/54

LEAF1(config-if)#Description \_Uplink\_To\_Spine

LEAF1(config-if)#ip address 209.165.201.1/31

**! Configuration on Spine Interface**

SPINE1(config)#Interface e1/1

SPINE1(config-if)#Description \_Link\_To\_Leaf

SPINE1(config-if)#ip address 209.165.201.2/31

Figure 10 - P2P Link between Leaf1 and Spine1

**A close up of a logo

Description automatically generated**

### MTU Considerations and Provisioning

Below table shows the MTU value recommendations for BNPP

Table 7 – MTU Recommendations

|  |  |
| --- | --- |
| Type of Network | MTU Recommendation |
| Fabric - Underlay L3 | 9100 |
| Leaf - External L2/L3 Devices | max 9000 |

* Per [rfc7348](https://tools.ietf.org/html/rfc7348), VTEPs MUST NOT fragment VXLAN packets.

#### MTU between fabric devices

An MTU of 9100 bytes on each interface on the path between the VTEPs accommodates maximum server MTU + VXLAN overhead. Most datacenter server NICs support up to 9000 bytes. So, no fragmentation is needed for VXLAN traffic.

***Configuration 2*– MTU Between Fabric Devices**

interface x/x

description UNDERLAY interface

mtu 9100

#### MTU on Leaf switch (facing fabric external devices)

Most servers utilize a maximum MTU of 9000 so the recommendation is to limit a maximum MTU of 9000 for external devices that require larger than standard 1500 bytes packets.

***Configuration 3*– MTU on Leaf switch (External Connections)**

interface x/x

description OVERLAY interface toward servers

mtu *<1500-9000>*

!

interface vlan 1000  
description OVERLAY interface toward servers

mtu *9000*

!

#### TCP MSS on all switches

TCP MSS will cover the TCP session size from devices to devices to avoid fragmentation. In Nexus 9000 switches the default TCP MSS value is 536 bytes. In case of VXLAN fabric the goal is to provide TCP MSS value of 8960 bytes to allow all BGP messages to have IP MTU of 9000 bytes. Additionally, this command is global to the switches and will provide MSS value of 8960 bytes for every TCP connection on affected switches.

!  
configuration terminal

ip tcp mss 8960

!

### IP Unicast Routing IS-IS

Based on BNPP’s preference, IS-IS has been selected as IGP in the Underlay. This approach is matching Cisco’s best practices and recommendations.

Since IS-IS uses Connectionless Network Service (CLNS) and is independent of the IP, full SPF calculation is avoided when a link changes. This benefit, in its turn, supports the advantage of IS-IS.

Level 1 IS-IS in the Fabric — Cisco has validated the use of IS-IS Level 1 only configuration on all nodes in the programmable fabric. The fabric is considered a stub network where every node needs an optimal path to every other node in the fabric. Cisco NX-OS IS-IS implementation scales well to support several nodes in a fabric, hence there is no anticipation of having to break up the fabric into multiple IS-IS domains. An IS-IS level 2 only configuration would have been an equally valid choice and might be a supported option in the future, however, note that this configuration has not been validated by Cisco quality assurance department.

As discussed above IS-IS – P2P link scenario with /31 mask will be considered in deployment. A sample P2P configuration between Access Leaf and Spine switch is given below.

Configuration 4– IS-IS configuration on Access Leaf switch

feature isis

router isis UNDERLAY

log-adjacency-changes

net 49.0001.0010.0100.1074.00

is-type level-1

set-overload-bit on-startup 60

The same configuration should be implemented on elements:

* Access Leaves, Border Leaves (towards VXLAN Underlay only)
* Spines and Super Spines
* BGW.

Setting the overload bit – you can configure a Cisco Nexus switch to signal other devices not to use the switch as an intermediate hop in their shortest path first (SPF) calculations. You can optionally configure the overload bit temporarily on startup. In the above example, the *set-overload-bit* command is used to set the overload bit on startup to 60 seconds.

Setting “log-adjacency-changes” - Sends a system message whenever an IS-IS neighbor changes the state.

For each Access Leaf, we must configure a loopback interface and a P2P interface configuration to connect to Spine. A sample P2P configuration between an Access Leaf switch interface and a Spine switch interface is given below.

Configuration 5– IS-IS P2P interface configuration (Access Leaf switch sample)

interface Ethernet 1/41

description Link to Spine S1

mtu 9100

ip address 209.165.201.1/31

ip router isis UNDERLAY

The same configuration should be implemented between elements:

* Access Leaves and Border Leaves to Spines
* BGW – Spines, BGW – Super Spines
* Spines – Super Spines.
* Spines – Spines

Configure a loopback interface so that it can be used as the IS-IS router ID of every network element.

Configuration 6– IS-IS loopback interface configuration

interface loopback 0

ip address 10.1.1.74/32

ip router isis UNDERLAY

The IS-IS instance is tagged as UNDERLAY for better recall.

#### ECMP

Configures the maximum number of equal-cost paths that IS-IS maintains in the route table. The range is from 1 to 64. The default is 8. For Small and Large Fabric template the default value is enough due to performance requirements and VXLAN EVPN multi-site scalabilities, but for Medium Fabric template it could be increased to 18 or 32 depending on the number of Spines and Super-Spines.

Configuration 7 – IS-IS ECMP

router isis UNDERLAY

maximum-paths <number>

### Multi-destination traffic

Multi-destination traffic comprises traffic types such as broadcast, unknown unicast, and multicast (BUM). The underlay must provide a way to carry BUM traffic between the edge devices. VXLAN has two different ways of handling BUM traffic:

* Unicast mode, known as ingress replication, or head-end replication, is one way to create a multicast-independent underlay.
* Multicast mode that employs IP multicast in the underlay is the other option.

#### Unicast Mode

In unicast mode, the data packets are replicated by the ingress VTEP and sent to the respective neighboring VTEPs that are part of the same VNI. Because the ingress VTEP has to address all the neighboring VTEPs, it inherently knows about them and has the burden of replicating and sending the BUM traffic to these VTEPs. In other words, if N VTEPs have membership in a given VNI, for every multi-destination packet in that VNI, a VTEP has to generate N–1 copies of the packet. Each copy is VXLAN encapsulated, with the outer DIP set to that of the destination VTEP to which it must be delivered. From the underlay’s as well as the destination VTEP’s point of view, this packet is simply treated as a regular VXLAN-encapsulated unicast packet.

There are two configuration options for unicast mode. The first option is static where a list of interested VTEPs is configured at each VTEP. The other option is to configure the list dynamically, using a control protocol, in this case BGP EVPN, to distribute the interest information within the network. Static configuration reduces scalability because any new VTEP or VNI added to the network requires reconfiguration at each of the neighboring VTEPs and it is not recommended.

BGP EVPN supports unicast mode. For example, configuration snippet below, the BGP EVPN control protocol is leveraged with Route type 3 (inclusive multicast) messages to distribute VTEP and VNI membership information. This does not mean that BGP EVPN is doing the job of ingress replication, but instead it simply distributes the VTEP and VNI membership information over the control protocol. Replication of data traffic is a data plane operation performed by the VTEP. As a result, the ingress or head-end replication operation must be supported by the forwarding functionality that is embedded at the VTEPs.

Configuration 8 – Ingress Replication Configuration with EVPN

interface NVE 1

source-interface loopback1

member vni 10001

ingress-replication protocol bgp

**!if all VNIs will use ingress replication, enable ingress replication globally**

interface NVE 1

global ingress-replication protocol bgp

#### Multicast Mode (PIM ASM)

Multicast Mode, known as Multicast Replication, is the second approach using an IP multicast network in the underlay in order to address this multicast replication inside the Fabric.

In this mode, a VNI is mapped to a multicast group, a tree will be made in order to connect the source to all the receivers. When the source VTEP (leaf or BGW) receives one BUM packet, it will send only one packet upstream to one spine and will replicate this packet regarding the multicast tree built by the IP multicast network in underlay reaching all the VTEP that have this VNI configured.

This mode needs a new protocol inside the fabric (PIM ASM, PIM BiDir or PIM SSM), troubleshooting must be made on ISIS and IP multicast network for all the BUM traffic contrary to the Unicast Mode.

##### Enabling PIM ASM

To enable PIM ASM, it is required to enable feature PIM per each VTEP (Leaf, Spine, Super Spines, BGW) in Small and Medium templates as per snipper below. Apart from this PIM should be enabled on underlay interfaces in-between all devices within fabric.

Configuration 9– Enabling PIM

feature pim

!

interface Ethernetx/y

description Link to Spine/Super Spine/Leaf/BGW

ip pim sparse-mode

##### Configuring PIM Anycast RP

With PIM ASM, the particularity of Anycast RP is the active-active function, the LHR (Last Hop Router or receivers) will built the shared tree to the closest RP regarding the routing table and PIM algorithm. If there is two RPs inside the network, then two independent shared trees can be built.

Table 8 – PIM Anycast RP Placement

|  |  |
| --- | --- |
| Fabric Template | Anycast RPs Placement |
| Small Template | 2 RPs configured on 2 Spines (one in each site if stretched) |
| Medium Template | 2 RPs on 2 Super Spine switches for the whole fabric |
| Large Template | 2 RPs configured on 2 Spines in each site |

Configuration 10– PIM Anycast RP Configuration Example

**! Loopback Interface Configuration on each RP, enable PIM on Lo0 IGP Interface**

interface loopback0

description IGP Loopback Router\_ID

ip address 10.10.10.x/32

ip pim sparse-mode

**! Loopback Interface Configuration (Anycast RP) on all RPs**

interface loopback 254

description Anycast RP

ip address 10.10.10.254/32

ip pim sparse-mode

**! Anycast-RP Configuration on all RPs**

ip pim rp-address 10.10.10.254 group-list 224.0.0.0/8

ip pim anycast-rp 10.10.10.254 10.10.10.1 (Spine-1 IP)

ip pim anycast-rp 10.10.10.254 10.10.10.2 (Spine-2 IP)

**! Configure RP for mcast group on all Leafs**

ip pim rp-address 10.10.10.254 group-list 224.0.0.0/8

##### Multicast Grouping for VXLAN

There are three main approaches to map L2VNIs to Multicast groups:

— One-to-One L2VNI to Multicast Group mapping

— VRF-based L2VNI to Multicast Group mapping

— Odd & Even L2VNI to Multicast Group mapping

In this case of one to one L2VNI and multicast group mapping, number of VLANs is equal to number of L2VNIs. Hence number of VLANs could be easily limited and it is suboptimal approach in case of BNPP scale requirements. Below example of configuration for reference purposes, two VNIs are utilizing two separate multicast groups.

Configuration 11– One-to-One L2VNI to Multicast Group mapping

interfcae nve 1

member vni 10011

mcast-group 225.1.1.11

member vni 10012

mcast-group 225.1.1.12

There is another approach to map L2VNIs, stitched to specific VRF, to one separate multicast group. In this case if number of VLANs are significantly higher than number of VRFs and VLANs are spread across multiply VRFs then this approach become more attractive in perspective scale and ability to separate BUM traffic flows. This option is widely used and recommended for deployments. Below example of configuration, two VNIs (2000401 & 2000402) of three VNIs are utilizing same separate multicast group (225.1.1.12) based on common VRF.

Configuration 12– VRF-based L2VNI to Multicast Group mapping

interfcae nve 1

member vni 2000201

mcast-group 225.1.1.11

member vni 2000401

mcast-group 225.1.1.12

member vni 2000402

mcast-group 225.1.1.12

Apart from previous option there is variant to split L2VNIs to different multicast groups based on parity of L2VNI number. Odd L2VNIs are mapped to multicast group number one and Even L2VNIs are mapped to a second multicast group. This approach could be extended by adding four or more multicast groups and split based on parity. Parity based L2VNI separation is used widely and recommended as well as previous option. Below example of configuration for parity-based mapping, VNIs 2000201 & 2000401 are mapped to 225.1.1.11 group as odd and VNI 2000402 is mapped to 225.1.1.12 group as even.

Configuration 13– Odd & Even L2VNI to Multicast Group mapping

interfcae nve 1

member vni 2000201

mcast-group 225.1.1.11

member vni 2000401

mcast-group 225.1.1.11

member vni 2000402

mcast-group 225.1.1.12

It is possible to mix the approaches having for the Vlans/L2VNIs with regular BUM traffic a “L3NVI-based Multicast mapping” using a defined Multicast Subnet IP Addresses reserved, and for the Vlans/L2VNIs where BUM traffic is important a “One-to-One L2VNI to Multicast Group Mapping” with a dedicated Multicast Subnet IP Addresses.

### BFD

BFD is a detection protocol designed to provide fast forwarding-path failure detection times. BFD provides sub-second failure detection between two adjacent devices and can be less CPU-intensive than protocol hello messages because some of the BFD load can be distributed onto the data plane on supported modules.

There are two places to enable BFD for IS-IS and PIM protocols what will give faster IGP and multicast protocol convergence improvement. BNPP has no requirements for this protocol but it is recommended to utilize it where it is possible in order make network more resilient.

* On Cisco Nexus 9000 9.3.x, BFD Echo Function is enabled by default.

#### BFD IS-IS

It is recommended to enable BFD in IS-IS IGP on the Leaf-Spine as well as Spine – BGW, BGW – Super Spine (DCI devices) and Super Spine – Super Spine (Inter DCI) links between the DCs. As per design in every template point-to-point subnet will be used on L3 post-channels and it is supposed to utilized BFD on these links.

Example configuration for IS-IS where BFD is enabled under IPv4 and an IPv6 address family.

Configuration 14– BFD IS-IS

router isis UNDERLAY

bfd

!

router isis UNDERLAY

address-family ipv4 unicast

bfd

#### BFD PIM

Protocol Independent Multicast (PIM) uses a hello mechanism for neighbor discovery and failure detection. The minimum failure detection time in PIM is 3.5 times the PIM hello interval. With default 30-second PIM Hello interval in VXLAN EVPN Multi-Site fabric, Hold-Time (Idle Timeout) is 105 seconds. You can achieve faster failure detection by lowering the interval between PIM Hello messages sent from the interface. However, lower intervals increase the load on the protocol and can increase CPU and memory utilization impacting Control-Plane performance system-wide. Lower intervals can also cause PIM neighbors to expire unpredictably as the neighbor expiry can occur before hello messages received are processes by the switch Control Plane.

The BFD Support for Multicast (PIM) feature, also known as PIM BFD, registers PIM as a client of BFD. PIM can then utilize BFD to initiate a session with an adjacent PIM node to support BFD's fast adjacency failure detection. BFD for PIM in a VXLAN Underlay is a recommended Best Practice to achieve fast failover.

Configuration is specified for a single physical port in Underlay, same configuration must be applied on a peer side, respectively. Configuration of PIM BFD will take place on the same interfaces on which Underlay IPv4 eBGP peering is established, configurations for “no ip redirects”, “no ipv6 redirects” and “bfd interval …” should have been applied on that interfaces already and are included here for consistency.

Configuration 15– PIM BFD for Underlay

ip pim bfd

!

interface <slot>/<port>

no ip redirects

no ipv6 redirects

bfd interval 300 min\_rx 300 multiplier 3

ip pim bfd-instance

#### BFD on L3 Port-channel (Per-Link)

You can configure the BFD session parameters for all BFD sessions on a port channel. If per-link mode is used for Layer 3 port channels, BFD creates a session for each link in the port channel and provides an aggregate result to client protocols. For example, if the BFD session for one link on a port channel is up, BFD informs client protocols, such as OSPF, that the port channel is up.

* When the BFD per-link mode is configured, the BFD echo function is not supported. You must disable the BFD echo function using the no bfd echo command before configuring the bfd per-link command.
* Ensure that you enable LACP on the port channel before you enable BFD.

Configuration 16– Per-Link BFD

interface port-channel x

no bfd echo

no ip redirects

no ipv6 redirects

bfd per-link

## Overlay

This chapter contains primary section for BGP EVPN as Overlay for all fabric templates and special section BGW for Large fabric template. Also, NGOAM feature is part of Overlay as a tool for overlay verification.

### VXLAN BGP EVPN

This chapter is related to VXLAN BGP EVPN provisioning within Small and Medium fabric templates and for Large fabric template (within each CLOS fabric – VXLAN site). In chapter [BGP Route Reflectors](#_BGP_Route_Reflectors), Two route reflectors in each site (in all templates) will be configured. RRs will be on two spine switches in each site.

[Vxlan Multisite](#_Vxlan_Multi-site) chapter reviews configuration of BGW and inter BGW control plane.

Prerequisites to enable VXLAN BGP EVPN on devices with VTEP and without VTEP functionality:

* Adjust MTU (if required)
* Create IP address plan and provision IP addresses on appropriate interfaces (L3 interfaces, Port-Channels and loopbacks)
* Provision IGP – IS-IS
* Provision multicast – PIM ASM (optional)
* Enable BFD (optional)

To Enable VXLAN BGP EVPN on Access Leaf, Service Leaf and Border Leaf L2/L3 (VTEPs) it is required to provision:

* Enablement VXLAN and the EVPN
* Configuring VLAN and VXLAN VNI
* Configuring VRF for VXLAN Routing (provision if required, depends on use case)
* Configuring SVI for Hosts for VXLAN Routing (provision if required, depends on use case)
* Configuring VRF Overlay VLAN for VXLAN Routing (provision if required, depends on use case)
* Configuring the NVE Interface and VNIs
* Configuring BGP on the VTEPs (Leaf and BGW)
* Suppressing ARP

To Enable VXLAN BGP EVPN on Spines and Super Spines it is required to provision:

* Configuring BGP on the Spines without VTEP functionality

Each template will have different BGP peerings, please refer to the below table which summarizes the BGP neighborships needed for each template.

Table 9 – EVPN BGP neighborships in each template

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BGP Peer A | BGP Peer B | Site | Small Template | Medium Template | Large Template |
| Leafs & BGWs | Spines | Same site |  |  |  |
| RR Spines Site 1 | RR Spines Site n | All sites |  |  |  |
| BGW Site 1\*\* | BGW Site n | All sites |  |  |  |

\*\* If Route Server exists in Large Template, then BGP peer B will be all the route servers. Please refer to Vxlan Multi-site chapter.

#### BGP EVPN on Leaves

First need to enable VXLAN and the EVPN on each network element with VTEP functionality (Leaves and BGWs).

Configuration 17 – Enablement VXLAN and the EVPN

**! Enable VLAN-based VXLAN**

feature vn-segment-vlan-based

**! Enable VXLAN**

feature nv overlay

**! Enable the EVPN control plane for VXLAN**

nv overlay evpn

To provision L2VNI and VLAN we need to assign VXLAN VNI identifier which could contain VLAN ID number inside itself.

For BNPP, L2VNI will be 20000 + VLAN ID. For example, VLAN ID 201 and VXLAN L2VNI 20201. This configuration must be heterogeneous across all Leaves within fabric.

For L2VNI, BNP will use the below RT and RD values:

RD = RID:L2VNI

RT = ASN:L2VNI \* BNP will use only 2 Bytes ASN

Configuration 18– Configuring VLAN and VXLAN VNI

vlan <vlan-number>

vn-segment <vn-number>

!

evpn

vni <20000+VLANID> l2

rd *RID:L2VNI*

route-target both *ASN:L2VNI*

!

For L3VNIs, it will be equal to 30000+vrf-id. vrf-id is internal for BNP.

Below values for RD and RT will be used:

RD = RID:L3VNI

RT = ASN:L3VNI \* BNP will use only 2 Bytes ASN

Provision VRF for VXLAN Routing as per snippet below.

Configuration 19– Configuring VRF for VXLAN Routing

vrf context <context-name>

vni <*30000+vrf-id*>

rd *RID:L3VNI*

address-family ipv4 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

address-family ipv6 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

!

Configuring SVI for Hosts for VXLAN Routing

Configuration 20– Configuring SVI for Hosts for VXLAN Routing

vlan <vlan-number>

vn-segment <vn-number>

!

interface vlan <vlan-number>

vrf member <vrf-name>

ip address <ip-address> <netmask>

!

Configuring VRF Overlay VLAN for VXLAN Routing

Configuration 21– Configuring VRF Overlay VLAN for VXLAN Routing

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

vlan <vlan-number>

vn-segment <vn-segment>

!

**! Configure VRF overlay VLAN/SVI for the VRF**

interface Vlan101

no shutdown

vrf member vxlan-900001

ip forward

**!The system vlan nve-overlay id command should be used for a VRF or a Layer-3 VNI (L3VNI) only. Do not use this command for regular VLANs or Layer-2 VNIs (L2VNI).**

system vlan nve-overlay id <>

**! Create VRF and configure VNI**

vrf context vxlan-900001

vni 900001

Configuring Anycast Gateway for VXLAN Routing.

The Anycast Gateways on all the edge devices of the VXLAN fabric share the same MAC address for the gateway service. The same MAC address is shared across all the different IP subnets, and each subnet has its own unique default gateway IP.

Every VXLAN fabric should have individual MAC address. Sample MAC address per fabric could the following: cafe.0000.00xx (where xx is the number of the fabric).

Configuration 22– Configuring Anycast Gateway for VXLAN Routing

**! Configure distributed gateway virtual MAC address**

**! All VTEPs should have the same virtual MAC address**

fabric forwarding anycast-gateway-mac <mac-address>

!

interface vlan <vlan-number>

fabric forwarding mode anycast-gateway

!

Configuring the NVE Interface and VNIs

Configuration 23 - Configuring the NVE Interface and VNIs

interface nve-interface

**! This defines BGP as the mechanism for host reachability advertisement**

host-reachability protocol bgp

**! Add Layer-3 VNIs, one per tenant VRF, to the overlay**

member vni <vni-number> associate-vrf

**! Add Layer 2 VNIs to the tunnel interface**

member vni <vni-number>

**! Configure the mcast group on a per-VNI basis**

mcast-group <multicast-group-address>

!

* Command “global mcast-group ip-address {L2 | L3}” configures the mcast group on a per-NVE interface basis. This applies to all Layer 2 VNIs. This type of configuration is not recommended to BNPP.

Configuring BGP on the VTEPs (Leaves and BGW)

Configuration 24 - Configuring BGP on the VTEPs (Leaves and BGW)

feature bgp

**! Use only private BGP ASN range 64512-65534 as per RFC 6996**

router bgp <ASN-number>

log-neighbor-changes

address-family l2vpn evpn

**! Requires advertise virtual-rmac in NVE**

advertise-pip

**! Recommended to provision the same as IP address of interface loopback 0 for IGP**

router-id <address>

**! Provision Spines or Super Spines as RR depending on template**

**! ASN-number shuld remain the same and consistent within each fabric, except Large fabric; In Large Fabric template each VXLAN site has to have own private ASN-number**

neighbor <RR-address> remote-as <ASN-number>

description RR on Spine <x>

**! Use loopback 0 as source interface for BGP peering**

update-source loopback0

password 0 <pwd>

**! Configure address family Layer 2 VPN EVPN under the BGP neighbor.**

address-family l2vpn evpn

send-community

send-community extended

vrf <vrf-name>

address-family ipv4 unicast

**! Configure iBGP Multipath Load Sharing (Leaf will select multiple iBGP paths as the best paths to a destination.The best paths or multipaths are then installed in the IP routing table)**

maximum-paths ibgp 2

advertise l2vpn evpn

**! In case you're willing to advertise Type 5, you need network/redistribute statements. This will be reviewed in detail later in chapters 4.6 and 4.8;**

network <ip-network/subnet>

!

address-family ipv6 unicast

advertise l2vpn evpn

!

!

* To disable IPv6 NLRI advertisement for a VRF toward the EVPN, disable the VNI in NVE by entering the *no member vni <vni-number> associate-vrf* command in interface nve1. The *<vni-number>* is the VNI associated with that particular VRF.

#### BGP EVPN on Spines

Configuring BGP on the Spines without VTEP functionality.

Configuration 25– Configuring BGP on the Spines without VTEP functionality

feature bgp

**! Use only private BGP ASN range 64512-65534 as per RFC 6996**

router bgp <ASN-number>

log-neighbor-changes

address-family l2vpn evpn

**! Recommended to provision the same as IP address of interface loopback 0 for IGP**

router-id <address>

**! Use IP address of interface loopback 0 of each Leaf for neighboring**

neighbor <address> remote-as <ASN-number>

**! Use loopback 0 as source interface for BGP peering**

update-source loopback0

**! Never use BFD on L2VPN EVPN**

**! no bfd**

password 0 <pwd>

address-family l2vpn evpn

send-community

send-community extended

!

#### BGP Route Reflectors

Two route reflectors in each site (in all templates) will be configured. RRs will be on two spine switches in each site.

**Configuration 26 - Configuring Route Reflector Spines in all templates**

router bgp <ASN-number>

address-family l2vpn evpn

**! Use IP address of interface loopback 0 of each Leaf for neighboring**

neighbor <leaf-address>

description RR Client - Leaf <x>

**! Route-reflecetor-client is only required if neighbor is a Leaf switch**

route-reflector-client

!

### Vxlan Multi-site

This section solely dedicated for Vxlan Multi-site functionality, which is required only in Large fabric template, but before proceeding there are prerequisites for each CLOS fabric (VXLAN site) which need to add to multi-site fabric:

* Adjust MTU (if required)
* Create IP address plan and provision IP addresses on appropriate interfaces (L3 interfaces, Port-Channels and loopbacks)
* Provision IGP – IS-IS for Intra-site and for Inter-site communication
* Provision multicast – PIM ASM
* Enable BFD (optional)

#### Guidelines and Limitations for VXLAN EVPN Multi-Site

VXLAN EVPN Multi-Site has the following configuration guidelines and limitations:

* The following switches support VXLAN EVPN Multi-Site: Cisco Nexus 9336C-FX2 and 93240YC-FX2 platform switches.
* The Multi-Site border gateway allows the co-existence of Multi-Site extensions (Layer 2 unicast/multicast and Layer 3 unicast) as well as Layer 3 unicast and multicast external connectivity.
* Anycast mode can support up to six border gateways per site.
* Beginning with Cisco NX-OS Release 9.2(1), Border Gateways (BGWs) in a vPC topology are supported.
* Support for Multicast Flood Domain between inter-site/fabric border gateways is not supported.
* Multicast Underlay between sites is not supported.
* iBGP EVPN Peering between border gateways of different fabrics/sites is not supported.
* Anycast mode can only support Layer 3 services that are attached to local interfaces.
* In Anycast mode, BUM is replicated to each border-leaf and DF election, between Border Leafs of a particular site decides which border leaf would forward the traffic inter-site traffic (Fabric to DCI and vice versa) for that site.
* In Anycast mode, all the Layer 3 services are advertised in BGP via EVPN Type-5 routes with their physical IP as the next hop.
* vPC mode can support only two border gateways.
* vPC mode can support both Layer 2 hosts and Layer 3 services on local interfaces.
* In vPC mode, BUM is replicated to either of the border-gateway’s for traffic coming from external site and hence both the border gateways are forwarders for site external to site internal (DCI to Fabric) direction.
* In vPC mode, BUM is replicated to either of the border gateways for traffic coming from the local site leaf for a VLAN using Ingress Replication (IR) underlay. Both border gateways are forwarders for site internal to site external (Fabric to DCI) direction for VLANs using the IR underlay.
* In vPC mode, BUM is replicated to both border gateways for traffic coming from the local site leaf for a VLAN using the multicast underlay. Therefore, a decapper/forwarder election happens and the decapsulation winner/forwarder only forwards the site-local traffic to external site border-gateways for VLANs using the multicast underlay.
* In vPC mode, all the Layer 3 services/attachments are advertised in BGP via EVPN Type-5 routes with their virtual IP as next hop. If the VIP/PIP feature is configured, they are advertised with PIP as the next hop.
* If different Anycast Gateway MAC addresses are configured across sites, ARP suppression must be enabled for all VLANs that have been extended.
* Bind NVE to a loopback address that is separate from loopback addresses that are required by Layer 3 protocols. A best practice is to use a dedicated loopback address for the NVE source interface (PIP VTEP) and Multi-Site source interface (anycast and virtual IP VTEP).

#### Enabling VXLAN EVPN Multi-Site

First need to enable VXLAN and the EVPN on each network element with VTEP functionality (Leaves and BGWs).

Configuration 27– Enablement VXLAN and the EVPN

**! Enable VLAN-based VXLAN**

feature vn-segment-vlan-based

**! Enable VXLAN**

feature nv overlay

**! Enable the EVPN control plane for VXLAN**

nv overlay evpn

Configuration snippet below enables the VXLAN EVPN Multi-Site feature. Multi-Site is enabled on the border gateways only. The site-id must be the same on all border gateways in the fabric/site.

Configuration 28– Enabling VXLAN EVPN Multi-Site

evpn multisite border-gateway <site-id>

**! Delay restore for advertisement of Anycast BGW IP**

delay-restore time 180

!

interface Loopback <y>

description VTEP Source

no ip address <a.b.c.d>/<mm> secondary

!

interface Loopback <y+1>

description Multisite VTEP

ip address <a.b.c.d>/<mm>

ip pim sparse-mode

!

interface nve1

source-interface loopback 0

host-reachability protocol bgp

multisite border-gateway interface loopback <y+1>

member vni <vni-id>

suppress-arp

multisite ingress-replication

mcast-group <multicast-group-address>

!

!

router bgp <ASN-number>

neighbor <address-BGW/RS>

remote-as <ASN-number-remote>

description Site-External Peer (Other Site BGW, PGW or RS)

update-source loopback0

ebgp-multihop 2

password 0 <password>

**! Fabric-external configuration, only for site-external peers**

peer-type fabric-external

address-family l2vpn evpn

send-community

send-community extended

rewrite-evpn-rt-asn

!

!

neighbor <address-Spine>

remote-as <ASN-number-internal>

description Site-Internal Peer (Site Internal Spine)

update-source loopback0

ebgp-multihop 2

password 0 <password>

address-family l2vpn evpn

send-community

send-community extended

rewrite-evpn-rt-asn

!

!

!

Command *multisite border-gateway interface loopback <y+1>* Defines the loopback interface used for the border gateway virtual IP address (VIP). The border-gateway interface must be a loopback interface that is configured on the switch with a valid /32 IP address. This /32 IP address must be known by the transient devices in the transport network and the remote VTEPs. This is accomplished by advertising it through a dynamic routing protocol in the transport network. This loopback must be different than the source interface loopback. The range of vi-num is from 0 to 1023.

Command *source-interface loopback 0* 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 be known by the transient devices in the transport network and the remote VTEPs. This is accomplished by advertising it through a dynamic routing protocol in the transport network.

Configuration snippet below describes the configuration to track all DCI facing interfaces and site internal/fabric facing interfaces. Tracking is mandatory and is used to disable re-origination of EVPN routes either from or to a site if all the DCI/fabric links go down.

Configuration 29– Configuring Fabric/DCI Link Tracking

!

interface Ethernet<x>/<y>

description Site-Internal

evpn multisite fabric-tracking

!

interface Ethernet<x+1>/<y+1>

description Site-Internal

evpn multisite dci-tracking

!

#### BGW BUM Traffic Enforcement

Storm-control feature should be considered as part of overlay layer as it controls user traffic.

A special feature, called EVPN Multi-Site storm-control, is designed to control how much BUM traffic is allowed to propagate to other legacy sites. Layer 2 Broadcast, Unknown Unicast, and Multicast can be individually fine-tuned at the vPC BGW level to limit the propagation of those traffic types in aggregate toward the remote sites.

This configuration is to be applied to all the Pseudo BGWs in the VXLAN EVPN Multi-Site Large fabric template interconnecting the fabric with Legacy Network for migration purposes

Configuration 30– EVPN Storm Control for BGWs

evpn storm-control broadcast level 0-100

evpn storm-control multicast level 0-100

evpn storm-control unicast level 0-100

#### Inter-site communication

To perform inter site communication Control Plane BGP Route Server or full-mesh configuration is required and for data-plane communication over Underlay DCI is required.

##### Inter-site Data Plane communication

For BGW communications between site it is required to provide underlay DCI communication over Super Spines. Super Spines could provide resilient IP transport based on IS-IS as IGP with modified number of paths. To provision Super Spines, it is necessary to perform next steps as discussed above in Underlay chapter:

* Adjust MTU (if required)
* Create IP address plan and provision IP addresses on appropriate interfaces (L3 interfaces, Port-Channels and loopbacks)
* Provision IGP – IS-IS
* Enable BFD (optional)

##### Inter-site Control Plane communication

EVPN Multi-Site architecture requires every BGW from a local site to peer with every BGW at remote sites. There are two options as discussed in HLD, full-mesh and Route Servers spread across sites within DCI Underlay. To provision full-mesh for peering procedure in chapters above could be used, but for Route Server a special configuration is required.

Route Server functionality could be placed on any Super Spine or on separate device. To help ensure that the route-server deployment provides resiliency for the EVPN Multi-Site control-plane exchange in any failure scenario, connectivity or device redundancy is required.

The route server must be able to support the EVPN address family, reflect VPN routes, and manipulate the next-hop behavior (next-hop unchanged). In addition, the route server should support route-target rewrite to simplify the deployment.

The configuration snippet for a site-external route server is shown below.

Configuration 31– Route Server

feature bgp

!

**! No need to enable VXLAN overlay or encapsulation**

**! The route map enforces the policy to leave the overlay next hop unchanged when the route server is used.**

**! The route server is not a VTEP or BGW and hence should not have the next hop pointing to itself.**

!

route-map UNCHANGED permit 10

set ip next-hop unchanged

!

**! Define the BGP routing instance with a site-independent autonomous system, ASN has to be from private range.**

**! You must ensure that all the received EVPN advertisements are reflected even if all the tenant VRF instances are not created on the route server. The route targets must be preserved while that function is performed (retain route-target all).**router bgp <ASN-number>

!

address-family l2vpn evpn

retain route-target all

!

!

**! The per-neighbor configuration for the overlay control-plane function in a route server can be simplified. The configuration of the BGP reachability function across multiple hops (ebgp-multihop) and preservation of the next hop between the BGWs are common settings. These configuration knobs, including the source interface, can be combined in a BGP peer template.**

**! BGP peer templates are part of the BGP instance configuration.**

!

template peer OVERLAY-PEERING

update-source loopback0

ebgp-multihop <hop-number>

address-family l2vpn evpn

send-community both

route-map UNCHANGED out

!

!

neighbor <BGW-address-site-x> remote-as <ASN-number-site-x>

inherit peer OVERLAY-PEERING

address-family l2vpn evpn

rewrite-evpn-rt-asn

!

!

neighbor <BGW-address-site-x+1> remote-as <ASN-number-site-x+1>

inherit peer OVERLAY-PEERING

address-family l2vpn evpn

rewrite-evpn-rt-asn

!

!

### ARP Suppression

It is recommended to enable ARP suppression in Overlay as per snippet example below.

* Configuring the *hardware access-list tcam region arp-ether <size> double-wide* is **not** required on Cisco Nexus 9200, 9300-EX, 9300-FX, 9300-GX, and 9300-FX2 platform switches.

Configuration 32 Suppressing ARP

interface nve 1

**!enable arp suppression globally on all L2VNIs**

global suppress-arp

member vni <vni-id>

!

!

### NGOAM

It is recommended to enable VXLAN operations, administration, and maintenance (OAM) on all Switches in the fabric to facilitate installing, monitoring, and troubleshooting Ethernet networks to enhance management in VXLAN based overlay networks.

Similar to ping and traceroute utilities that allow quick determination of problems in the IP networks, VXLAN OAM provides very similar ping, pathtrace, and traceroute utilities to diagnose the reachability to the hosts and the VTEPs in a VXLAN network. The OAM channel is used to identify the type of the VXLAN payload that is present in these OAM packets.

There are two types of payloads supported:

* Conventional ICMP packet to the destination to be tracked
* Special NVO3 draft Tissa OAM header which is used in pathtrace and carries useful information about ingress and egress interface and interface statistics on a per-hop basis.

This configuration must be applied to all the VTEPs in all fabric templates.

Configuration 33– NGOAM Feature

feature ngoam

ngoam install acl

ngoam profile 1

oam-channel 2

ngoam authentication-key <key-string>

Below is an example of how to use NGOAM to verify reachability to a remote host in a specific VRF with simple ICMP checks.

Configuration 34– Example of NGOAM Ping

ping nve ip <a.b.c.d> vrf <vrf-name> source <a.b.c.d> verbose

Below is an example of how to use NGOAM pathtrace to trace connectivity between the VTEPs hop-by-hop in case NGOAM ping fails. Beginning with Cisco NX-OS 7.0(3)I6(1), NGOAM authenticates the pathtrace requests to provide the statistics by using the HMAC MD5 authentication mechanism. If NGOAM MD5 key doesn't match between the devices in the path, then only interface names without statistics are sent in the pathtrace reply.

Configuration 35 - Example of NGOAM Pathtrace

pathtrace nve ip <a.b.c.d> profile <number> vrf <vrf-name> vni <nvid> req-stats verbose

## VPC VTEPs

Virtual Port-Channel (vPC) VTEPs combines two technologies, VPC and VXLAN, to provide device-level redundancy for VTEPs. A pair of vPC switches share the same VTEP address, often referred to as the anycast VTEP address, and function as a logical VTEP. The other VTEPs in the network see the two switches as a single VTEP with the anycast VTEP address. When both the vPC VTEP switches are up and running, they load share in an active-active configuration. If one vPC switch goes down, the other switch tales over the entire traffic load so that the failure event doesn’t cause loss of connectivity for the devices connected to the vPC pair.

As per BNPP requirements, Classical vPC peering with MCT link technology should be employed to provide redundancy and resiliency for end-host systems connected to pairs of Leafs (Access, Service, Border L2/L3).

The vPC Peer Link is the binding entity that pairs individual Switches into a vPC domain. This link is used to synchronize the two individual Switches and assists Layer 2 control-plane protocols, like BPDUs or LACP, as it would come from one single Node. In the cases where end-points are dual-homed to both vPC member switches, the Peer Links sole purpose is to synchronize the state information as described before, but in cases of single-connected End-Points, so called orphans, the vPC Peer Link can still potentially carry traffic.

### VPC Best Practices

* **vPC Delay Restore** – After vPC peer reload, traffic might be black-holed, before L3 connectivity is reestablished. vPC links bring up can be delayed allowing Underlay and Overlay Convergence. Default timer is 30 seconds. Relaxing this timer to 150 seconds does provide better convergence time by allowing more time for the Underlay and Overlay convergence before bringing up the VPCs.
* **SVI Delay Restore** – this feature delays SVI bring-up to allow Underlay and Overlay converge after vPC peer reload. After reloading, traffic might be black-holed. Default timer is 10 seconds.
* **NVE Hold-Down timer** – this timer suspends advertisements of NVE loopback interface until overlay has converged after vPC peer reload. After vPC peer reload, traffic going to Anycast VTEP hashed to the peer will be black-holed. NVE loopback interface bring up can be delayed using this hold-down timer. For proper overlay convergence, hold-down time needs to be longer than delay restore time. Default timer is 180 seconds. A timer of 220 seconds provides more time for the vPC port members to come up.
* **vPC Peer-Switch** – this feature makes the vPC peer devices to appear as single STP root and BPDUs processed by the logical STP root formed by the 2 vPC peer devices.
* **Unique Domain ID’s** – use unique Domain ID’s for vPC-to-vPC for all vPC pairs defined in a contiguous layer 2 domain. The vPC peer devices use the vPC domain ID to automatically assign a unique vPC system MAC address which must be unique. This system MAC is used in STP BPDU, LACP BPDU, and IGMP advertisements.
* **vPC ARP/ND synchronization** – this feature suspends SVIs when peer device goes down or peer link goes down. After restoring of the peer device, or peer link, ARP table is empty and traffic black-holed. Before bringing up SVI, peer devices synchronize ARP table over CFS service.
* **The vPC peer link should be resilient**. A recommended practice is for the peer-link port-channel to contain links at least of two separate ports if possible.
* **vPC Infrastructure VLANs** – these VLANs are used for backup routing path and present on Peer Link and used in case of failure of uplinks on vPC peer or failure of vPC port members. Also, it is used for BUM traffic transfer.
* **Bridge Assurance** should be configured on the vPC peer link ends. Entering the vPC peer link command does this automatically. ISSU requires having no Bridge Assurance on vPC member links.
* **Peer-gateway** allows a vPC peer device to act as the active gateway for packets addressed to the other peer device router MAC. It keeps the forwarding of traffic local to the vPC peer device and avoids use of the peer-link.
* **Auto-recovery** restores vPC services when its peer fails to come online. You must save this setting in the startup configuration. On reload, if the peer link is down and three consecutive peer-keepalive messages are lost, the secondary device assumes the primary STP role and the primary LACP role. The software reinitializes the vPCs, bringing up its local ports. Because there are no peers, the consistency check is bypassed for the local vPC ports. The device elects itself to be the STP primary regardless of its role priority and also acts as the master for LACP port roles (default value 240 sec). It is recommended to use an extended timer of 360 sec which is more secure in case of both VPC peers reloading at the same time.
* **LACP vpc-convergence:** when configured the switch waits until all the VLANs are initialized and programmed and then send LACP sync PDU, which will start sending traffic to the VPC domain without drops. You may configure the lacp vpc-convergence command in a VXLAN and non-VXLAN environments that have vPC port-channels to hosts that support LACP. This command must be enabled on both vPC peer switches. This command must be configured only on PortFast/Edge ports (vPC port channels on which the spanning-tree port type edge [trunk] is enabled).
* **vPC orphan-port suspend**: to suspend a nonvirtual port channel (vPC) port when the peer link of a vPC secondary goes down, use the vpc orphan-port suspend command. This command will reduce convergence time on Border Leaf with non vPC L3 routed ports connection to Edge Router. This command can also be used on all non vPC ports (except Fabric Uplinks to the Spines), especially in the scenario where servers are using active-standby type of teaming, forcing a physical shutdown on the link for the teaming software to change the traffic to flow on the other link connected to the primary.

### VPC Considerations for VXLAN EVPN:

* For VPC, the loopback interface has 2 IP addresses: the primary IP address and the secondary IP address. The primary IP address is unique and is used by Layer 3 protocols. The secondary IP address on the loopback is necessary because the interface NVE uses it for the VTEP IP address, particularly for all VxLAN traffic that includes multicast and unicasts encapsulated traffic. The secondary IP address must be the same on both vPC peers.
* Each VPC peer needs to have separate BGP sessions to the spine.
* VPC peers must have identical configurations.
  + Consistent VLAN to VN-segment mapping
  + Consistent NVE1 binding to the same loopback interface
  + Using the same Secondary IP address
  + Using different primary IP addresses
  + Consistent VNI to group mapping
  + The VRF overlay VLAN should be a member of the peer-link port-channel
* VPC “peer-gateway” feature must be enabled on both peers.
* When Nexus 9000 switches are configured as VXLAN Leaf Switches also known as VXLAN Tunnel End Points (VTEP) in virtual Port Channel (vPC) domain, you must have a backup Layer 3 Routing adjacency in between them over the vPC peer-link with the use of an interface vlan. This VLAN must be local to the switches, not stretched across the VXLAN fabric and belong to the Default VRF (Global Routing Table).
* It is worth noting that once a pair of VXLAN switches is configured as part of a vPC domain, the anycast VTEP is always used as next-hop for all the EVPN advertisements relative to directly connected endpoints. This is valid also for local end points connected in Active/Standby fashion. The consequence is that roughly half of the flows destined to those devices may be delivered from the spines to the VTEP device connected to the standby end points (the spines have two equal cost paths to reach the Any cast VTEP IP address); the traffic would hence have to take an extra hop across the peer-link in order to be delivered to the active interface of the endpoint. This suboptimal behavior can be avoided by grouping endpoints based on the types of connectivity (Active/Standby vs LACP) and connecting them to separate sets of leaf switches.

### Configuring VPC

We recommend associating a peer-keepalive link to a separate virtual routing and forwarding (VRF) instance that is mapped to a Layer 3 interface in each vPC peer device. If you do not configure a separate VRF, the system uses the management VRF by default.

It is recommended to run an SVI link between the peer switches in addition to the VPC Peer link and VPC Peer-keepalive. This link is used by VXLAN to pass traffic between the peers if all uplinks are down on one of the VPC peers. If all links to the core go down, the SVI between one pair of LEAF will lead the routing continuity. iBGP peering to the Route-Reflector will stay up and running.

**Configuration 36– VPC and NVE Hold-Down timer**

feature lacp

feature vpc

!

interface mgmt0

vrf member management

ip address <a.a.a.a>/<netmask>

!

vpc domain 100

peer-switch

**! Priority has to be provisioned different on pair of Leafs**

role priority {4096 | 8192}

peer-keepalive destination <b.b.b.b> source <a.a.a.a> vrf management eq 3200

auto-recovery

delay restore 150

delay restore interface-vlan 10

auto-recovery reload-delay 360

peer-gateway

**! Configure layer3 peer-router under VPC domain, this command is needed as dynamic routing is configured on SVI 3967 over VPC peerlink**

layer3 peer-router

ip arp synchronize

!

interface nve1

source-interface hold-down-time 220

!

**! Minimum two interfaces should be part of vPC PL**

interface ethernet <number>/<x - x+1>

description vPC PL

switchport

channel-group <id> mode active

no shutdown

!

interface port-channel1

description vPC PL

switchport

switchport mode trunk

switchport trunk native vlan 2

switchport trunk allowed vlan <range>

spanning-tree port type network

vpc peer-link

interface port-channel <number>

switchport

switchport mode trunk

switchport native vlan 2

switchport trunk allowed vlan <range>

! **If not connected to another switch**

spanning-tree port type edge trunk

lacp vpc-convergence

vpc <number>

* LACP vpc-convergence reminder: this command must be enabled on both vPC peer switches. This command must be configured only on PortFast/Edge ports (vPC port channels on which the spanning-tree port type edge [trunk] is enabled).

The VXLAN VPC configuration requires a secondary loopback address added to the VXLAN source loopback interface. This IP address must be the same on both VPC peer devices.

**Configuration 37– VPC VIP**

!

interface loopback1

ip address x.x.x.x/32 secondary

!

### Timeline for vPC recovery

Interactions and timeline of the following timers:

VPC level

delay restore interface-vlan 10: this timer starts when vPC peer-link comes up

delay restore 150: this timer starts when delay restore interface-vlan timer does expire

NVE level

source-interface hold-down-time 220: this timer starts when vPC peer-link comes up

Figure 11 - Timeline for vPC recovery

T0: peer link comes up

T1: T0 + delay restore interface vlan

T2: T1 + delay restore

T3: T0 + source-interface hold-down-time 220

It takes 160 (10+150) seconds before the system brings up the vPCs, by configuring the source-interface hold-down-time to 220 this leave 60 seconds to the switch for bringing up all vPC before the NVE loopback interface is bring up.

### VXLAN Infra Vlan

When Nexus 9000 switches are configured as VXLAN Leaf Switches also known as VXLAN Tunnel End Points (VTEP) in virtual Port Channel (vPC) domain, you must have a backup Layer 3 Routing adjacency in between them over the vPC peer-link with the use of an interface vlan. This VLAN must be local to the switches, not stretched across the VXLAN fabric and belong to the Default VRF (Global Routing Table). It is possible to use any VLANs except 3968-4095 that are allocated for internal device use.

Ensure the system nve infra-vlans command is in place on Nexus 9000 platforms with CloudScale ASIC (Tahoe) like the Nexus 9300 Switches which end in EX, FX and FX2 to specify the VLAN can act as an uplink and properly forward the frames with VXLAN encapsulation over the vPC peer-link.

Vlan 3967 is an example to participate in the underlay Routing Protocol which in this case is IS-IS, and for multicast PIM (if PIM is used for BUM traffic).

Note: IP subnets used as a backup Layer 3 Routing adjacency should be unique for each vPC domain.

**Configuration 38– On Leaf1**

!  
vlan 3967

name BACKUP\_VLAN\_ROUTING\_NVE\_INFRA

!

interface Vlan3967

no shutdown

no ip redirects

ip address 10.1.2.1/24

no ipv6 redirects

mtu 9100

**ip router isis UNDERLAY**

**ip pim sparse-mode**

!

**system nve infra-vlans** 3967

!

**Configuration 39****– On Leaf2**

vlan 3967

name BACKUP\_VLAN\_ROUTING\_NVE\_INFRA

!

interface Vlan3967

no shutdown

no ip redirects

ip address 10.1.2.2/24

no ipv6 redirects

mtu 9100

**ip router isis UNDERLAY**

**ip pim sparse-mode**

!

**system nve infra-vlans** 3967

!

## Classical Ethernet & MST

VXLAN does not natively integrate with STP, and this results in a VXLAN-enabled bridge domain (VLAN/L2VNI) that is always in forwarding state. Because BPDUs are not transported over the point-to-multipoint VXLAN network, the southbound classic Ethernet connectivity is required to be loop-free.

In any case it is recommended to use STP protocols in case of loop occurrence in fabrics, apart from this there are build-in loop prevention mechanisms in VXLAN EVPN technology based on duplicate detection for IP and MAC Addresses.

As Spanning Tree Protocol, the best option is MSTP and this is the preference of BNPP engineering teams. For example, RPVST virtual ports limit is 12000 for EX/FX/FX-2 platforms and 190000 is the limit for MST PV count with single instances 0. In a Layer 2 looped topology design, spanning tree processing instances are created on each interface for each active VLAN. These logical instances are used by the spanning tree process in processing the spanning tree-related packets for each VLAN. These instances are referred to as active virtual ports. Active virtual ports are important values to consider in spanning tree designs because they affect STP convergence time and stability. Virtual ports are a per-line card value that reflects the total number of spanning tree processing instances used on a particular line card.

Apart from this MSTP consumes less processor cycles than any other STP protocol due to less instances then in RPVST (one or two instances per VLAN) for example. All these considerations make MST more advantageous then other protocols.

In addition to the specific connectivity requirements for VXLAN, protecting the network boundary is also important and a recommended practice. Traditional Layer 2 tools such as BPDU guard, root guard, or storm control provide protection at any classic Ethernet edge port. These tools allow for a clean and protected demarcation, which prevents undesirable failure propagation. As a result, desired network stability is ensured.

Configuration 40– MST for Leaf Switches

spanning-tree mst configuration

name FABRIC-STP

revision 10

instance 1 vlan 1-4094

spanning-tree mode mst

## DHCP Relay in VXLAN

With DHCP Relay, DHCP messages require to be sent through the same Switch in both directions. GiAddr (Gateway IP Address) for DHCP Relay is commonly used for Scope Selection and DHCP response messages. In any VXLAN fabric with Distributed IP Anycast Gateway, DHCP messages can be returned to ANY Switch hosting the respective Gateway IP Address (GiAddr).

Solution requires a different way of Scope Selection and Unique IP Address for each Switch. Unique Loopback Interface per Switch and per vrf will become GiAddr for responding to correct Switch. Option 82 (dhcp option vpn) will be used for Scope Selection based on L2VNI.

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 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.
* Sub-option 5(0x5) - Link Selection (Defined in RFC 3527); 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.
* It is recommended to use a DHCP server which supports the options listed above (Infoblox should support those). Windows Server 2012 with DHCP Service does not support the sub-options so it uses instead the “Option 82 suboption: (1) Agent Circuit ID” which does convey the VNI information, this solution is less flexible. For reference : <https://www.cisco.com/c/en/us/support/docs/switches/nexus-9000-series-switches/200248-Configuring-Microsoft-Windows-Server-201.html>

The following are common deployment scenarios are available:

* Client on tenant VRF and server on Layer 3 default VRF
* Client on tenant VRF (SVI A) and server on the same tenant VRF (SVI B)
* Client on tenant VRF (VRF A) and server on different tenant VRF (VRF B)
* Client on tenant VRF and server on non-default non-VXLAN VRF

BNPP will implement the following scenario: Client on tenant VRF (SVI A) and server on the same tenant VRF (SVI B), the other three solutions will lead to VRF leaking unlike this one. All flows must pass via the firewalls, VRF leaking would potentially break the security.

**Configuration 41–****Client on tenant VRF (SVI A) and server on the same tenant VRF (SVI B)**

feature dhcp

service dhcp

ip dhcp relay

ip dhcp relay information option

ip dhcp relay information option vpn

ipv6 dhcp relay

**! dedicated Loopback interface;**

interface loopback1001

vrf member vxlan-900001

ip address 11.11.11.11/32

interface Vlan10

no shutdown

vrf member vxlan-900001

ip address 192.1.42.1/24

interface Vlanl00l

description Client\_Tenant\_vrf\_vxlan-900001

**! DHCP Server address is 192.1.42.3 in VRF vxlan-900001**

ip dhcp relay address 192.1.42.3

ip dhcp relay source—interface loopback1001

ip prefix-list PL-loopback-dhcp seq 5 permit 11.11.11.0/24 eq 32

route-map RM-direct-to-bgp permit 100

match PL-loopback-dhcp

router bgp xxxxx

vrf vxlan-900001

address-family ipv4 unicast

redistribute direct route-map RM-direct-to-bgp

## External Connectivity

External Routers and Fabric are to be attached to Classical Ethernet of Service Leaves with non vPC L3 routed ports. Multitenant connectivity will be provided with VRF-lite using multiple L3 interfaces, one per VRF. These L3 interfaces will be Ethernet Sub-interfaces.

BNPP can have Static Routing or eBGP configuration options for external connectivity for all fabric templates.

The configuration options are based on Full mesh P2P model as discussed in the HLD with edge routers connected in different sites (different VTEPs).

### External Connectivity with Static Routing

Static routes will be configured per VRF. The VRF Name is purely localized and can be different to the VRF Name on the External Router, the L3VNI must be consistent across the VXLAN BGP EVPN fabric.

The below configuration example shows how to configure static route towards external router, assuming BGP EVPN and L3VNI are already configured in the VXLAN fabric.

**Configuration 42– Static Routing to Edge Routers**

interface Ethernet1/x

no switchport

vpc orphan-port suspend

no shutdown

interface Ethernet1/x.2

encapsulation dot1q 2

vrf member WEB

ip address 10.1.1.2/29

no shutdown

interface Ethernet1/x.3

encapsulation dot1q 3

vrf member APP

ip address 20.1.1.2/29

no shutdown

VRF context WEB

**! static route on VTEP pointing to Edge Router as next hop**

IP route x.x.x.x/y 10.1.1.1

VRF context APP

**! static route on VTEP pointing to Edge Router as next hop**

IP route x.x.x.x/y 20.1.1.1

**! The static route is redistributed into BGP**

router bgp 65000

vrf WEB

address-family ipv4 unicast

advertise l2vpn evpn

redistribute static route-map Static-to-BGP

vrf APP

address-family ipv4 unicast

advertise l2vpn evpn

redistribute static route-map Static-to-BGP

### External Connectivity with eBGP Routing

The VXLAN BGP EVPN Border Node acts as neighbor device to the External Router. The VRF Name is purely localized and can be different to the VRF Name on the External Router, only significance is the L3VNI must be consistent across the VXLAN BGP EVPN fabric.

The below configuration example shows how to peer with external router, assuming BGP EVPN and L3VNI are already configured in the VXLAN fabric.

**Configuration 43– BGP Routing to External Routers**

!

router bgp 65002

vrf myvrf\_50001

neighbor 10.31.95.95

remote-as 65099

address-family ipv4 unicast

!

interface Ethernet1/3

no switchport

vpc orphan-port suspend

no shutdown

interface Ethernet1/3.2

encapsulation dot1q 2

vrf member myvrf\_50001

ip address 10.31.95.31/24

no shutdown

The VXLAN BGP EVPN Border Node has the ability to advertise IPv4 and IPv6 default-route within the fabric. In cases where it is not beneficial to advertise the Host Routes from the VXLAN BGP EVPN fabric to the External Router, these IPv4 /32 can be filtered at the External Connectivity peering configuration.

**Configuration 44– eBGP Filtering Host Routes**

ip prefix-list default-route seq 5 permit 0.0.0.0/0 le 1

!

ip prefix-list host-route seq 5 permit 0.0.0.0/0 eq 32

!

route-map extcon-rmap-filter deny 10

match ip address prefix-list default-route

route-map extcon-rmap-filter deny 20

match ip address prefix-list host-route

route-map extcon-rmap-filter permit 1000

!

vrf context myvrf\_50001

ip route 0.0.0.0/0 10.31.95.95

!

router bgp 65002

vrf myvrf\_50001

address-family ipv4 unicast

network 0.0.0.0/0

neighbor 10.31.95.95

remote-as 65099

address-family ipv4 unicast

route-map extcon-rmap-filter out

**Active/Standby External Routers**

If external devices (Routers) connect to different VTEPs and advertise the same prefixes into the VXLAN fabric with equal routing metrics, then we will have two active exit points for the whole VXLAN fabric.

Each access leaf will prefer the local exit point as it is closer from routing perspective.

Figure 12 - Active/Standby Exit Points



In case we want to have the exit point only in DC1 for instance, then we have to modify the routing attributes to prefer BL1/BL2 when a VTEP does the routing look up for prefix A. For example, if external peering is done using eBGP and Edge Router 1 advertises the BGP community attribute with the advertised prefix, we can configure a route-map on BL1/BL2 to increase the local preference value for that specific community. In this case, the local Access Leaf 1 and the remote Access Leaf 2 will choose BL1/BL2 in DC1 as an exit point, and BL3/BL4 will be used only in case prefix A is not being advertised into the VXLAN Fabric through BL1/BL2.

In the below example Router Edge 1 is sending the prefixes with BGP community 11:11, and the configuration on BL1/BL2 modifies the local preference value for that community:

**Configuration 45– Configure a Preferred Exit Point**

**! Create Community List**

ip community-list standard FROM\_ER1 seq 1 permit 11:11

!

**! Create Route-map to match the community and set local preference value**

route-map DC1-LP permit 10

match community FROM\_ER1

set local-preference 200

!

**! Apply route-map to neighbour edge router 1**

router bgp 65002

vrf myvrf\_50001

neighbor 10.31.95.95

address-family ipv4 unicast

send-community

route-map DC1-LP in

!

* If in any case routes need to be advertised back to VXLAN fabric (with VXLAN BGP ASN in the as-path), then “**allowas-in 1**” needs to be applied on border leaves in order to accept the routes which has local ASN in AS path. If external routers are nexus devices, you need to configure “**disable-peer-as-check**” on the neighbor facing border leaves to be able to advertise these routes.

## Service Models

There are two main service models that have been defined by BNP for host connectivity provided by VXLAN EVPN Fabric:

* End Host with Anycast Gateway (AG)
* Default Gateway on FW

### End Host with Anycast Gateway (AG)

BNPP will mainly consider Use Case with Distributed Anycast Gateway (DAG) for end-hosts and Firewall in the middle. This use case is applicable to all templates and will operate in the same way except Large fabric where BGWs will perform VXLAN packet re-stitching but with L2/L3VNI consistence across sites.

Fabric will provide per VLAN separate broadcast domain with dedicated identifier L2VNI and at the same time fibric will provide Distributed Anycast Gateway (DAG) for each service. Distributed Anycast Gateway (DAG) is one of the recommended options as it provides resilient gateways for mobile virtual end-host which will use their ability to move within fabric. Centralized Gateway (CG) may have some restrictions in case of BNPP and it is not recommended.

Configuration 46– Configuring Anycast Gateway for VXLAN Routing

**! Configure distributed gateway virtual MAC address**

**! One virtual MAC per VTEP**

**! All VTEPs should have the same virtual MAC address**

fabric forwarding anycast-gateway-mac address

!

interface vlan <vlan-number>

fabric forwarding mode anycast-gateway

!

* For more details about DAG design and traffic flows, please refer to HLD document.

### Service Model with Default Gateway on FW

Another use case defined with default gateway placed on FW shown in Figure 50. This service model is simpler in comparison to previous models. The main difference is the default gateway is provided by the FW which is responsible for North-South and East-West traffic.

No VXLAN specific configuration is for this model.

* For more details about placing the default GW on Firewall design and traffic flows, please refer to HLD document.

## L4-L7 Integration

BNPP requirement is to have the Active and Standby Firewalls connected on separate pair of VPC Leaves.

This chapter will discuss the connection to Firewalls using eBGP, and Static routing.

### Connection with eBGP

Active/Standby FW pair are connected to separate leaf node pairs. Below are the design considerations:

* eBGP peering using the anycast IP is not supported. The recommended design is to use dedicated loopback IPs per VRF on each VTEP for the eBGP peering.
* Reachability to the loopback from the firewall can be configured using a static route on the firewall, pointing to the Anycast Gateway IP on the VTEPs.
* BGP peering must be enabled between the vPC VTEPs in the same VPC domain (site).
* To minimize the traffic outage after a FW failover event, the active FW should peer with local and remote leaf nodes (eBGP multi-hop is needed). To achieve this, the loopback used for eBGP peering must be redistributed into the BGP EVPN.
* Enable BGP to advertise the gateway IP in the EVPN Type-5 routes, use the export-gateway-ip command.
* After failover, there is no need to re-establish eBGP sessions between FWs and the fabric. This will reduce the convergence time in case of a FW failover event.

Figure 13 - Firewall interconnection with BGP

A close up of a map

Description automatically generated

Figure 14 - Firewall interconnection logical diagram

A picture containing umbrella

Description automatically generated

* Below configuration example is shown only for SL1 and SL2. Similar config needs to be on SL3 and SL4.

**Configuration 47– eBGP Routing to Firewalls**

**! Configuration on SL1**

Vlan 10

Name inside

Vn-segment 10010

Vlan 20

Name outside

Vn-segment 10020

Vlan 1000

Name l3vniinside

Vn-segment 1001000

Vlan 2000

Name l3vnioutside

Vn-segment 1002000

interface vlan1000

Description inside\_vrf

VRF member INSIDE

IP forward

interface vlan2000

Description outside\_vrf

VRF member OUTSIDE

IP forward

system nve infra-vlans 3967

Interface VLAN 10

Description inside\_vlan

VRF member INSIDE

IP address 10.1.1.254/24

fabric forwarding mode anycast-gateway

Interface VLAN 20

Description outside\_vlan

VRF member OUTSIDE

IP address 20.1.1.254/24

fabric forwarding mode anycast-gateway

Interface loopback100

vrf member INSIDE

ip address 172.16.1.253/32

Interface loopback101

vrf member OUTSIDE

ip address 172.18.1.253/32

router bgp 65000

vrf INSIDE

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.16.1.253/32

**!enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall Inside**

neighbor 10.1.1.0/24

remote-as 65002

update-source loopback100

ebgp-multihop 5

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

vrf OUTSIDE

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.18.1.253/32

**!enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall Outside**

neighbor 20.1.1.0/24

remote-as 65002

update-source loopback101

ebgp-multihop 5

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

**! Configuration on SL2**

Vlan 10

Name inside

Vn-segment 10010

Vlan 20

Name outside

Vn-segment 10020

Vlan 1000

Name l3vniinside

Vn-segment 1001000

Vlan 2000

Name l3vnioutside

Vn-segment 1002000

interface vlan1000

Description inside\_vrf

VRF member INSIDE

IP forward

interface vlan2000

Description outside\_vrf

VRF member OUTSIDE

IP forward

system nve infra-vlans 3967

Interface VLAN 10

Description inside\_vlan

VRF member INSIDE

IP address 10.1.1.254/24

fabric forwarding mode anycast-gateway

Interface VLAN 20

Description outside\_vlan

VRF member OUTSIDE

IP address 20.1.1.254/24

fabric forwarding mode anycast-gateway

Interface loopback100

Vrf member INSIDE

Ip address 172.16.1.254/32

Interface loopback101

Vrf member OUTSIDE

Ip address 172.18.1.254/32

router bgp 65000

vrf INSIDE

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.16.1.254/32

**!enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall Inside**

neighbor 10.1.1.0/24

remote-as 65002

update-source loopback100

ebgp-multihop 5

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

vrf OUTSIDE

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.18.1.254/32

**!enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall Outside**

neighbor 20.1.1.0/24

remote-as 65002

update-source loopback101

ebgp-multihop 5

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

### Connection with Static Routing

If the firewall does not support running a routing protocol, there is a need to have static routes on each VTEP pointing to the firewall as the next hop. The firewall also has static routes pointing to the Anycast Gateway IP as the next hop.

Active/Standby Firewalls in BNPP’s network will be connected to separate VPC VTEPs.

Figure 15 - Static Routing with Firewalls on different VPC VTEPs



The challenge with a static route is that the VTEP with an active firewall must be the one advertising the routes to the fabric. A way to accomplish this is to track the active firewall reachability via HMM and use this tracking to advertise routes into the fabric. When the active firewall is connected to VTEP 2, VTEP 2 has a static route with track where the route is advertised only if the firewall IP is learned as the HMM route. When the firewall fails and the standby takes over, VTEP 2 now learns the firewall IP using BGP and VTEP 3 learns the firewall IP using HMM. VTEP 2 withdraws the route and VTEP 3 advertises the route into the fabric.

* Without HMM tracking, the static route must be configured on all VTEPs in the respectful VRF.

Figure 16 - Static Routing with Firewalls on different VPC VTEPs (after FW failover)



**Configuration 48– Static Routing to Firewalls With HMM Tracking**

Vlan 12

Name inside

Vn-segment 10010

Vlan 11

Name outside

Vn-segment 10020

Vlan 1000

Name l3vniinside

Vn-segment 1001000

Vlan 2000

Name l3vnioutside

Vn-segment 1002000

Interface VLAN 12

Description inside\_vlan

VRF member INSIDE

IP address 10.1.1.254/24

fabric forwarding mode anycast-gateway

Interface VLAN 11

Description outside\_vlan

VRF member OUTSIDE

IP address 20.1.1.254/24

fabric forwarding mode anycast-gateway

interface vlan1000

Description inside\_vrf

VRF member INSIDE

IP forward

interface vlan2000

Description outside\_vrf

VRF member OUTSIDE

IP forward

interface nve1

no shutdown

host-reachability protocol bgp

source-interface loopback1

member vni 10010

mcastgroup 239.1.1.1

member vni 10020

mcastgroup 239.1.1.1

member vni 1001000 associate-vrf

member vni 1002000 associate-vrf

track 10 ip route 10.1.1.1/32 reachability hmm

vrf member INSIDE

track 20 ip route 20.1.1.1/32 reachability hmm

vrf member OUTSIDE

VRF context INSIDE

**! static route on VTEP pointing to Firewall next hop**

**! firewall VIP 10.1.1.1**

IP route 20.1.1.0/24 10.1.1.1 track 10

VRF context OUTSIDE

**! static route on VTEP pointing to Firewall next hop**

**! firewall VIP 20.1.1.1**

IP route 10.1.1.0/24 20.1.1.1 track 20

**! The static route is redistributed into BGP**

router bgp 65000

vrf INSIDE

address-family ipv4 unicast

redistribute static route-map INSIDE-to-BGP

vrf OUTSIDE

address-family ipv4 unicast

redistribute static route-map OUTSIDE-to-BGP

* If the route-map is matching a default route, add ‘default-information originate’ to advertise the default route. The redistribute command configured under the BGP process injects all the routes that exist in the Routing Table (permitted by a route-map) in the BGP RIB except for the default route. To additional allow the default route to be installed from the source routing protocol in the BGP RIB, the command **default-information originate** is required.

## Fabric Management

### Out-of-Band Network

Currently BNP use existing network infrastructure to provide out of band network management. New switches Cisco Nexus will be connected to existing OOB fabric for network management and for telemetry delivery to DCNM servers with NIR module.

* The OOB mgmt0 interface of Nexus switches, will also be used a vPC peer keep alive interface. So, keeping this interface UP, with a consistent configuration, is very important.

Servers and storages use OOB switches as network management switches, this out of band management network or separate heartbeat fabric will be used for heartbeat links between firewalls and load balancers in different fabrics.

### SSH/Telnet

Cisco NX-OS allows the use of SSH server to enable an SSH client to make a secure, encrypted connection to a Cisco NX-OS device. SSH uses strong encryption for authentication. The SSH server in the Cisco NX-OS software can interoperate with publicly and commercially available SSH clients.

* It is recommended to not enable telnet service as unsecure connection service for network management.

Configuration 49– SSH access configuration sample

ssh key rsa 2048 force

no ip domain-lookup

!

ip access-list 13

10 remark Version 1.4\_07.08.2018

20 remark X.X.X.X access

30 permit tcp X.X.X.X/Y any

!

line vty

exec-timeout 240

access-class 13 in

### NX-OS Programmability & Telemetry

#### YANG Data Modeling

The model-driven programmability of the NX-OS device allows to automate the configuration and control of the device.

Data modeling provides a programmatic and standards-based method of writing configurations to the network device, replacing the process of manual configuration. Data models are written in a standard, industry-defined language. Although configuration using a CLI may be more human-friendly, automating the configuration using data models results in better scalability.

The Cisco NX-OS on Cisco Nexus 9000 devices support the YANG data modeling language. YANG is a data modeling language used to describe configuration and operational data, remote procedure calls, and notifications for network devices.

Three standards-based programmable interfaces are supported by NX-OS for operations on the data model: NETCONF, RESTConf, and gRPC.

Model-Driven Programmability required by BNPP to achieve automation of the configuration and control of a devices by third-party orchestration and automation systems (Cisco DCNM will be used only for NIR telemetry).

* For more examples please refer to [Cisco Nexus OpenConfig YANG Reference](https://developer.cisco.com/docs/openconfig-yang-release-9-3x/#!about-openconfig-yang).

#### Model Driven Telemetry

Cisco NXOS streaming telemetry allows us to push data off the device to a defined endpoint as JSON or using Google Protocol Buffers (GPB) at a much higher frequency and more efficiently than SNMP/Syslog/CLI.

Telemetry data is collected from the Data Management Engine (DME) database in branches of the object model specified using distinguished name (DN) paths. The data can be retrieved periodically (frequency-based) or only when a change occurs in any object on a specified path (event-based). You can use the NX-API to collect frequency-based data.

3 simple steps are all that is required to configure telemetry.

* The is first step is to set the format and destination to which the data is to be sent.
* The second step is to configure the data that is to be collected as part of the sensor group.
* The third and final step is set the subscription between the sensor-group and the destination, along with the cadence at which to send the data (in msecs)

Configuration 50– Model Driven Telemetry Configuration Sample

!

feature nxapi

feature telemetry

!

nxapi use-vrf management

!

telemetry

destination-group 1

ip address 172.27.247.72 port 60001 protocol gRPC encoding GPB

**! create a subsciption that streams data for DME path/s**

sensor-group 1

! Default data source is DME

path sys/bgp depth unbounded

path sys/epId-1 depth unbounded

path sys/bd depth unbounded

path sys/ospf depth unbounded

path sys/intf depth unbounded

path sys/acl depth unbounded

path sys/ipqos depth unbounded

subscription 1

dst-grp 1

**! streaming data every 100 seconds**

snsr-grp 1 sample-interval 100000

**! Create stream of show commands collection (NXAPI)**

sensor-group 2

data-source NX-API

path "show system resources" depth 0

path "show version" depth 0

path "show environment power" depth 0

path "show environment fan" depth 0

path "show environment temperature" depth 0

path "show process cpu" depth 0

path "show nve peers" depth 0

path "show nve vni" depth 0

path "show nve vni 4002 counters" depth 0

path "show int nve 1 counters" depth 0

path "show policy-map vlan" depth 0

path "show ip access-list test" depth 0

path "show system internal access-list resource utilization" depth 0

subscription 1

dst-grp 1

**! streaming data every 100 seconds**

snsr-grp 2 sample-interval 100000

The following example enables streaming telemetry and configuring the source and destination of the data stream. These steps also include optional steps to enable and configure SSL/TLS certificates and GPB encoding.

* To configure SSL certificate-based authentication and the encryption of streamed data, you can provide a self-signed SSL certificate with certificate *SSL cert path* hostname "CN" command.

Configuration 51– Model Driven Telemetry Configuration Sample

!

**! To generate a private RSA key**

switch# openssl genrsa -des3 -out server.key 2048

**! To write the RSA key**

switch# openssl rsa -in server.key -out server.key

**! To create a certificate that contains the public or private key**

switch# openssl req -sha256 -new -key server.key -out server.csr -subj '/CN=localhost'

**! To create a public key**

switch# openssl x509 -req -sha256 -days 365 -in server.csr -signkey server.key -out server.crt

**!This example creates a subscription that streams data for sys/bgp every 5 seconds to destination IP 1.2.3.4 port 50003, and encrypts the stream using GPB encoding**

switch(config)# telemetry

switch(config-telemetry)# certificate /bootflash/server.key localhost

switch(conf-tm-telemetry)# destination-group 100

switch(conf-tm-dest)# ip address 1.2.3.4 port 50003 protocol gRPC encoding GPB

switch(config-dest)# sensor-group 100

switch(conf-tm-sensor)# path sys/bgp depth 0

switch(conf-tm-sensor)# subscription 100

switch(conf-tm-sub)# snsr-grp 100 sample-interval 5000

switch(conf-tm-sub)# dst-grp 100

* For more examples please refer to [NXOS Programmability Guide](https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/93x/progammability/guide/b-cisco-nexus-9000-series-nx-os-programmability-guide-93x/b-cisco-nexus-9000-series-nx-os-programmability-guide-93x_chapter_0101001.html#id_112825), [NXOS Telemetry](https://developer.cisco.com/docs/nx-os/#!telemetry/streaming-telemetry), and [VXLAN Telemetry Deployment](https://pubhub.devnetcloud.com/media/nx-os/docs/telemetryvxlan/Telemetry-Deployment-VXLAN-EVPN.pdf) documents.

### Authentication, Authorization and accounting (AAA)

In BNPP network, devices will use Authentication, Authorization and accounting (AAA) pre-defined group with TACACS+ servers in it.

Configuration 52– AAA configuration sample

!

feature tacacs

!

tacacs-server timeout 1

tacacs-server host <ACSserver1> key <key>

tacacs-server host <ACSserver2> key <key>

tacacs-server directed-request

aaa group server tacacs+ DFI\_TACACS

server <ACSserver1>

server <ACSserver2>

use-vrf management

source-interface mgmt0

!

aaa authentication login default group DFI\_TACACS

**! console login authentication is configured in the VDC default only**

aaa authentication login console local

aaa authorization commands default group DFI\_TACACS local

aaa accounting default group DFI\_TACACS

!

### Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

It is recommended to use SNMPv2/v2c and/or SNMPv3 to provide information to BNPP network management systems. Networking devices in each fabric will communicates with BNPP network management systems out-of-band via management Ethernet ports on switches.

There are access lists, associated with network management systems to provide the required SNMP access restrictions. Management plane protection configuration allows in-band SNMP access only via management Ethernet ports on switches.

Configuration 53– SNMP V3 Configuration Example

!

ip access-list SNMP-ACCESS\_IN

permit ip <snmp host1> any

permit ip <snmp host2> any

deny ip any any

!  
!  
snmp-server globalEnforcePriv  
!  
snmp-server user <username> <role> auth [md5|sha] <password> priv as- 128 <priv-pwd>  
!  
snmp-server location <location>  
snmp-server contact <contact>  
!  
snmp-server host <SNMP-Server> version 3 priv <username>  
snmp-server host <SNMP-Server> use-vrf management  
!  
snmp-server source-interface trap mgmt0  
snmp-server enable traps  
!  
snmp-server community <community> group network-admin  
snmp-server community <community> use-ipv4acl SNMP-ACCESS\_IN

### Logging

All devices in BNPP network send logging massages to logging servers to record changes of network states. Logging level 5 (Normal but significant condition) will used to log network event. Management IP addresses will used as source interface for logging messages.

Configuration 54– Configuration of the Logging Server

logging server x.x.x.x 5 use-vrf management

### Network Time Protocol (NTP)

BNPP infrastructure contains perceive network time protocol (NTP) servers as preferred for time synchronization. All network devices will use one out of these servers as preferred. Management IP addresses will used as source interface for NTP quires. Access-control list will be applied to protect NTP sessions (queries and responses).

Configuration 55– Configuration of NTP

ntp server x.x.x.x use-vrf management

ntp source-interface mgmt0

!

ip access-list NTP-ACCESS\_IN

permit ip <snmp host1> any

permit ip <snmp host2> any

deny ip any any

!

ntp access-group peer NTP-ACCESS\_IN

## Security features

Protecting and securing the network block devices themselves is a very important element of the current design, and the base for the overall BNPP secure infrastructure. The following subchapters present the special features activated for this in detail.

### Preventing VLAN Hopping

In a double tagging attack, an attacker connected to an 802.1Q-enabled port prepends two VLAN tags to a frame that it transmits. The frame (externally tagged with VLAN ID that the attacker's port is really a member of) is forwarded without the first tag because it is the native VLAN of a trunk interface. The second tag is then visible to the second switch that the frame encounters. This second VLAN tag indicates that the frame is destined for a target host on a second switch. The frame is then sent to the target host as though it originated on the target VLAN, effectively bypassing the network mechanisms that logically isolate VLANs from one another.

To prevent such attack, the following two configurations can be applied:

Configuration 56– Configuring trunk ports to use an unused VLAN.

vlan 2

shutdown

name NATIVE!

interface port-channel100

switchport trunk native vlan 2

Configuration 57– Explicit tagging of the native VLAN on all trunk ports

vlan dot1Q tag native

### Control plane security and protection

It is recommended to protect the network control protocols using authentication and encryption features like:

* IS-IS md5 authentication, between all N9K
* NTP md5 authentication, between all N9K and NTP servers
* BFD md5 authentication, between all N9K, and between external devices
* BGP md5 authentication, between all N9K and between external devices

#### IS-IS authentication

Provisioning authentication to control adjacencies and the exchange of LSPs is optional. Routers that want to become neighbors must exchange the same password for their configured level of authentication. IS-IS blocks a router that does not have the correct password. Provision IS-IS authentication globally for Level 1 routing areas only.

IS-IS supports the authentication method MD5 digest, all packets exchanged carry a message digest that is based on a 128-bit key.

To provide protection against passive attacks, IS-IS never sends the MD5 secret key as cleartext through the network. In addition, IS-IS includes a sequence number in each packet to protect against replay attacks. Below configuration sample for IS-IS authentication.

Configuration 58– IS-IS Authentication

router isis UNDERLAY

authentication-type md5 level-1

authentication key-chain <key> level-1

authentication-check level-1

#### NTP authentication

Configuring NTP authentication provides assurance that NTP messages are exchanged between trusted NTP peers. Enable authentication for NTP if at all possible.

Configuration 59– NTP authentication

ntp authentication-key <number> md5 <string>

ntp server x.x.x.x key <number>

ntp trusted-key <number>

ntp authenticate

NTP access can be controlled by using access groups. Specifically, the types of requests can be specified that the device allows and the servers from which it accepts responses. Without any access groups, NTP access is granted to all devices. With access groups, NTP access is granted only to the remote device whose source IP address passes the access list criteria.

Configuration 60– NTP access restriction

!

ip access-list NTP-ACCESS\_IN

permit ip <snmp host1> any

permit ip <snmp host2> any

permit ip <ntp server> any

deny ip any any

!

ntp access-group peer NTP-ACCESS\_IN

!

NTP Authentication and restriction is not needed on BNP cisco devices.

#### BFD authentication

Configures SHA-1 authentication for all BFD sessions on the interface. The “ascii\_key” string is a secret key shared among BFD peers. The “id” value, a number between 0 and 255, is assigned to this particular “ascii\_key”. BFD packets specify the key by “id”, allowing the use of multiple active keys. Below configuration sample for BFD authentication.

Configuration 61– BFD Authentication

interface ethernet x/x

bfd authentication keyed-sha1 <keyid id> key <ascii\_key>

#### BGP authentication

Consider configuration of TCP built-in MD5 authentication to ensure that each segment sent on the TCP connection between the peers is verified and is sent by a valid BGP L2VPN EVPN Peer. This configuration is to be applied to all the switches with BGP L2VPN EVPN sessions configured – Service Leaves, Server Leaves, Border Leaves, BGWs, Pseudo BGWs and Spines in all fabric templates.

Configuration 62– MD5 Authentication for L2VPN EVPN Peers

router bgp <asn>

neighbor <a.b.c.d>

inherit peer L2VPN\_Evpn

password 0 <text in clear>

### Management security

Special security restriction will be implemented for accessing the devices management. This will include ACL on VTY or mgmt0, user full authentication authorization and accounting (AAA) using minimum 2 TACACS+ servers (protected also by key). Access Control Lists (ACL) allows controlling traffic flows by defining which sources are allowed to “talk” to which destinations by which protocols.

Please note, the example below does not show a complete working access-list, it is just an extract from the current implementation.

Configuration 63– Management security

ip access-list MGMT0\_ACL

remark #### TACACS

permit tcp <mgmt-scope> eq tacacs any

remark #### RADIUS

permit udp <mgmt-scope> eq 1812 any

permit udp <mgmt-scope> eq 1813 any

remark #### NTP

permit udp <mgmt-scope> eq ntp any

remark #### FTP

permit tcp <mgmt-scope> any eq 20

permit tcp <mgmt-scope> eq 20 any

permit tcp <mgmt-scope> any eq 21

permit tcp <mgmt-scope> eq 21 any

remark #### TFTP

permit udp <mgmt-scope> any eq tftp

permit udp <mgmt-scope> eq tftp any

remark #### SSH

permit tcp <mgmt-scope> any eq 22

permit tcp <mgmt-scope> eq 22 any

remark #### SNMP

permit udp <mgmt-scope> any eq snmp

remark #### ICMP

permit icmp any any echo

permit icmp any any echo-reply

statistics per-entry

!

interface mgmt0

ip access-group MGMT0\_ACL in

!

line vty

access-class MGMT0\_ACL in

### Control plane policy

Nexus 9300 series have a control plane policy (CoPP) activated by default. This policy has the role of protecting the switch control plane from unwanted traffic, or security attacks by limiting the amount of traffic which will be sent from the data plane to the NX-OS/CPU. The settings of this policy are pretty granular and detailed per protocol/purpose and optimized by default by Cisco, based on the type of environment the switch is used for.

In the case of BNP, the CoPP strict default will be used on all Nexus 9300 Switches.

### Data plane security

The Nexus 9000 switches have special features which can protect the network itself from possible access server malfunctions, which can cause network instabilities.

Thee below features will be implemented in BNPPs network:

* Storm control policies on access ports
* Spanning Tree Interface Settings (BPDU Guard & BPDU Filter globally on all L2 interface to Servers)
* Err-disable Recovery timer Disabled
* Mac-move notifications

#### Storm Control

Traffic storm control (also called traffic suppression) allows you to monitor the levels of the incoming broadcast, multicast, and unicast traffic over a 3.9-millisecond interval. During this interval, the traffic level, which is a percentage of the total available bandwidth of the port, is compared with the traffic storm control level that you configured. When the ingress traffic reaches the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends.

When the traffic exceeds the configured level, you can configure traffic storm control to perform the following optional corrective actions:

* Shut down—When ingress traffic exceeds the traffic storm control level that is configured on a port, traffic storm control puts the port into the error-disabled state.
* Trap—You can configure traffic storm control to generate an SNMP trap when ingress traffic exceeds the configured traffic storm control level. The SNMP trap action is enabled by default. However, storm control traps are not rate-limited by default. You can control the number of traps generated per minute by using the snmp-server enable traps storm-control trap-rate command. Using trap as less disruptive for the active links.

By default, Cisco NX-OS takes no corrective action when traffic exceeds the configured level.

BNPP will implement Storm-Control feature on all access ports for Broadcast, Multicast, and Unknown Unicast traffic types.

Configuration 64– Configuring Traffic Storm Control

!

Interface ethernet x/y | port-channel x

storm-control broadcast level <level-value %>

storm-control multicast level <level-value %>

storm-control unicast level <level-value %>

**! Configure storm control action**

storm-control action trap

#### Spanning-Tree Interface Settings

**STP Edge Ports**

You connect STP edge ports only to Layer 2 hosts. The edge port interface immediately transitions to the forwarding state, without moving through the blocking or learning states.

The port can be and Edge or Edge Trunk:

* spanning-tree port type edge—This command explicitly enables edge behavior on the access port.
* spanning-tree port type edge trunk—This command explicitly enables edge behavior on the trunk port.

Configuration 65– Configuring STP Edge Ports

**! configuring an Edge port**

Interface ethernet x/y | port-channel x

switch port mode access

switch port mode access vlan *<number>*

spanning-tree port type edge

**! Configuring an Edge Trunk port**

Interface ethernet x/y | port-channel x

switch port mode trunk

spanning-tree port type edge trunk

**BPDU Guard**

Enabling BPDU Guard shuts down that interface if a BPDU is received.

You can configure BPDU Guard at the interface level. When configured at the interface level, BPDU Guard shuts the port down as soon as the port receives a BPDU, regardless of the port type configuration.

When you configure BPDU Guard globally, it is effective only on operational spanning tree edge ports.

By using BPDU Guard loop prevention of the fabric will be on the same level as for DFI actual fabric.

Configuration 66– Configuring STP BPDU Guard

**! Enabling STP BPDU Guard globally, enables BPDU guard for all STP edge interfaces**

spanning-tree port type edge bpduguard default

**BPDU Filtering**

You can use BPDU Filtering to prevent the device from sending or even receiving BPDUs on specified ports.

When configured globally, BPDU Filtering applies to all operational spanning tree edge ports. You should connect edge ports only to hosts, which typically drop BPDUs. If an operational spanning tree edge port receives a BPDU, it immediately returns to a normal spanning tree port type and moves through the regular transitions. In that case, BPDU Filtering is disabled on this port, and spanning tree resumes sending BPDUs on this port.

In addition, you can configure BPDU Filtering by the individual interface. When you explicitly configure BPDU Filtering on a port, that port does not send any BPDUs and drops all BPDUs that it receives. You can effectively override the global BPDU Filtering setting on individual ports by configuring the specific interface. This BPDU Filtering command on the interface applies to the entire interface, whether the interface is trunking or not.

* Use care when configuring BPDU Filtering per interface. If you explicitly configure BPDU Filtering on a port that is not connected to a host, it can result in bridging loops because the port will ignore any BPDU that it receives and go to forwarding.

This table lists all the BPDU Filtering combinations. In detail the table describe the following:

1. 1st use case shows that when the BPDU Filtering Per Port Configuration is default (there is No explicit port configuration “*no spanning-tree bpdu-filter*”), BPDU Filtering Global Configuration is enable and STP Edge Port Configuration is enable then BPDU Filtering State is enabled;
2. 2nd use case shows that when the BPDU Filtering Per Port Configuration is default (there is No explicit port configuration “*no spanning-tree bpdu-filter*”), BPDU Filtering Global Configuration is enable and STP Edge Port Configuration is disable then BPDU Filtering State is disable;
3. 3rd use case shows that when the BPDU Filtering Per Port Configuration is default (there is No explicit port configuration “*no spanning-tree bpdu-filter*”), BPDU Filtering Global Configuration is disable and STP Edge Port Configuration is N/A then BPDU Filtering State is disable;
4. 4th use case shows that when the BPDU Filtering Per Port Configuration is diable, BPDU Filtering Global Configuration is N/A and STP Edge Port Configuration is N/A then BPDU Filtering State is disable;
5. 5th use case shows that when the BPDU Filtering Per Port Configuration is enable, BPDU Filtering Global Configuration is N/A and STP Edge Port Configuration is N/A then BPDU Filtering State is enable;

Table 10 – BPDU Filtering Configurations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use cases | BPDU Filtering Per Port Configuration | BPDU Filtering Global Configuration | STP Edge Port Configuration | BPDU Filtering State |
| 1 | Default 1 | Enable | Enable | Enable 2 |
| 2 | Default | Enable | Disable | Disable |
| 3 | Default | Disable | Not applicable | Disable |
| 4 | Disable | Not applicable | Not applicable | Disable |
| 5 | Enable | Not applicable | Not applicable | Enable |

1 No explicit port configuration *“no spanning-tree bpdu-filter”*.

2 The port transmits at least 10 BPDUs. If this port receives any BPDUs, the port returns to the spanning tree normal port state and BPDU filtering is disabled.

Configuration 67– Configuring STP BPDU Filtering

**! Enabling STP BPDU Filter on a specified interface**

Interface ethernet x/y | port-channel x

Spanning-tree bpdufilter enable

!

* STP BPDU Filtering will only be used during migration in a very specific scenario, this is explained in NMP document.

#### Err-Disabled Recovery Timer

A port is in the error-disabled (err-disabled) state when the port is enabled administratively (using the no shutdown command) but disabled at runtime by any process. However, because the port is administratively enabled, the port status displays as err-disable. Once a port goes into the err-disable state, you must manually reenable it or you can configure a timeout value that provides an automatic recovery. By default, the automatic recovery is not configured, and by default, the err-disable detection is enabled for all causes.

If you do not enable the error-disabled recovery for the cause, the interface stays in the error-disabled state until you enter the shutdown and no shutdown commands. If the recovery is enabled for a cause, the interface is brought out of the error-disabled state and allowed to retry operation once all the causes have timed out.

BNPP wants to keep the err-disabled port state until there is a manual intervention (no shut) to recover the port. No recovery timer will be configured. Below configuration is just for information in case needed.

Configuration 68– Configuring Err-Disabled Recovery Timer

**! Specify a condition under which the interface automatically recovers from the error-disabled state, and the device retries bringing the interface up. The device waits 300 seconds to retry.**

Switch(config)# errdisable recovery cause <>

**! Specify the interval for the interface to recover from the error-disabled state. The range is from 30 to 65535 seconds, and the default is 300 seconds.**

Switch(config)# errdisable recovery interval *interval* <>

#### Mac-Move Notifications

When ASIC learns too many mac-addresses in a short duration, it will disable/enable MAC learning in hardware and you can see below message coming up. It could be caused if there are too many mac-moves/flaps/loops or new mac learns/moves exceeds a certain threshold. By default, on Nexus9K, you may not see the logs, which specifically tells us that the switch is experiencing mac-moves.

To find out the mac address that is causing this problem and the vlans and the interface information where, this is being experienced, we need to raise the logging level of L2FM from the default value of 2 up to 5 on N9K platform.

Configuration 69– Configuring L2FM Logging Level

**! Raise L2FM logging level from defauult value 2 to 5 on N9K**

Switch(config)# logging level l2fm 5

## Quality of Service

There are no specific requirements by BNPP for QoS. The only requirement is to include this chapter to explain the default behavior of QoS in VXLAN.

**Considerations for VXLAN QoS**

* Cisco Nexus 9364C, 9300-EX, and 9300-FX/FX2 platform switches and Cisco Nexus 9500 platform switches with -EX/FX or -R/RX line cards support VXLAN QoS.
* Beginning with Cisco NX-OS Release 9.3(3), Cisco Nexus 9300-GX platform switches support VXLAN QoS in default mode.
* VXLAN QoS is supported in the EVPN fabric.
* The original IEEE 802.1Q header is not preserved in the VXLAN tunnel. The CoS value is not present in the inner header of the VXLAN-encapsulated packet.
* Statistics (counters) are present for the NVE interface.
* Entering the policy-map type qos command in the output direction for egress policing is not supported in the ingress VTEP.
* In a vPC, configure the change of the decapsulated packet priority selection on both peers.
* The service policy on an NVE interface can attach only in the input direction.
* If DSCP marking is present on the NVE interface, traffic to the BUD node preserves marking in the inner and outer headers. If a marking action is configured on the NVE interface, BUM traffic is marked with a new DSCP value on Cisco Nexus 9364C and 9300-EX platform switches.
* A classification policy applied to an NVE interface applies only on VXLAN-encapsulated traffic. For all other traffic, the classification policy must be applied on the incoming interface.
* To mark the decapsulated packet with a CoS value, a marking policy must be attached to the NVE interface to mark the CoS value to packets where the VLAN header is present.

### Classification at the Ingress VTEP

Traffic on an ingress VTEP is classified based on the priority in the original header. Classification can be performed by matching the CoS, DSCP, and IP precedence values or by matching traffic with the ACL based on the original frame data.

When traffic is encapsulated in the VXLAN, the Layer 3 packet's DSCP value is copied from the original header to the outer header of the VXLAN encapsulated packet. This behavior is illustrated in the following figure:

Figure 17 - Copy of DSCP from L3 packet to VXLAN Outer Header

A screenshot of a cell phone

Description automatically generated

For Layer 2 frames without the IP header, the DSCP value of the outer header is derived from the CoS-to-DSCP mapping present in the hardware illustrated in table ‘Default Settings for VXLAN QoS’. In this way, the original QoS attributes are preserved in the VXLAN tunnel. This behavior is illustrated in the following figure:

Figure 18 - Copy of CoS from L2 packet to VXLAN Outer Header

**A screenshot of a cell phone

Description automatically generated**

Table 11 – Default Settings for VXLAN QOS

|  |  |
| --- | --- |
| CoS of Original L2 Frame | DSCP of Outer VXLAN Header |
| **0** | **0** |
| **1** | **8** |
| **2** | **16** |
| **3** | **26** |
| **4** | **32** |
| **5** | **46** |
| **6** | **48** |
| **7** | **56** |

### Inside the VXLAN Tunnel

Inside the VXLAN tunnel, traffic classification is based on the outer header DSCP value. Classification can be done matching the DCSP value or using ACLs for classification.

If VXLAN encapsulated traffic is crossing the trust boundary, marking can be changed in the packet to match QoS behavior in the tunnel. Marking can be performed inside of the VXLAN tunnel, where a new DSCP value is applied only on the outer header. The new DSCP value can influence different QoS behaviors inside the VXLAN tunnel. The original DSCP value is preserved in the inner header.

Figure 19 - Marking Inside of the VXLAN Tunnel

**A close up of a map

Description automatically generated**

### VXLAN to IP

Classification at the egress VTEP is performed for traffic leaving the VXLAN tunnel. For classification at the egress VTEP, the inner header values are used. The inner DSCP value is used for priority-based classification. Classification can be performed using ACLs.

Classification is performed on the NVE interface for all VXLAN tunneled traffic.

Marking and policing can be performed on the NVE interface for tunneled traffic. If marking is configured, newly marked values are present in the decapsulated packet. Because the original CoS value is not preserved in the encapsulated packet, marking can be performed for decapsulated packets for any devices that expect an 802.1p field for QoS in the rest of the network.

### Decapsulated Packet Priority Selection

At the egress VTEP, the VXLAN header is removed from the packet and the decapsulated packet egresses the switch with the DSCP value. The switch assigns the DSCP value of the decapsulated packet based on two modes:

* Uniform mode – the DSCP value from the outer header of the VXLAN packet is copied to the decapsulated packet. Any change of the DSCP value in the VXLAN tunnel is preserved and present in the decapsulated packet. Uniform mode is the default mode of decapsulated packet priority selection.
* Pipe mode – the original DSCP value is preserved at the VXLAN tunnel end. At the egress VTEP, the system copies the inner DSCP value to the decapsulated packet DSCP value. In this way, the original DSCP value is preserved at the end of the VXLAN tunnel.

## Fabric Considerations and Preparations

This section summarizes in general the preparations and considerations to take into account before building the fabric. The information in this section are already mentioned in the previous chapters.

1. Preparing the physical connections and discussed in [Chapter 3](#_Physical_Design_Overview).
2. Initial configuration of the devices, naming conventions, and connection to OOB through mgmt0 interface.
3. Configuring the Underlay. Keep in mind that IP addresses need to be reserved for the below:

* Loopback 0, this loopback interface is required on every network element within fabric for IGP’s router ID. Technically it will not be used by IGP for any purposes, there is only match between router ID and IP address of this loopback.
* Loopback 1, this loopback utilizes one IP address as Virtual IP address (VIP) shared between two vPC peers and one Physical IP address (PIP) as unique VTEP identifier. This interface is required for all Leaves and BGWs (BGWs IP addressing will be considered in detail later in this document).
* Loopback <y> & <y+1>, these loopbacks are used for RP Anycast mode on set of Spines/SuperSpines if multicast is the mechanism of BUM traffic delivery.
* Interface vlan <x>, this Infra VLAN for Underlay backup routing over Peer Link between vPC Peers.

1. Once the underlay is functional, the overlay can be configured according to [Chapter 4.2](#_Overlay). BGP AS number is unique per fabric.
2. VPC VTEPS, please make sure the VPC domain IDs are unique per VPC Pair devices. VPC domains must not be duplicated.
3. Fabric Management and Security Features can be configured as per requirements.
4. External connectivity and L4-L7 Integration can be configured as per requirements.

* Please refer to the respective chapters as a main reference and details. The sequence of actions in this section is not necessarily in order.

# Configuration Templates

In this chapter we have configuration templates for Access Leaf, Border/Service Leaf, Spine, and Super-spine.

* NXOS Telemetry is not included in the below templates because there are no specific requirements for the data to be collected, for details please refer to [Chapter 4.9.3](#_NX-OS_Programmability_&)

## Access Leaf Configuration Template

Configuration 70– Configuring Template for Access Leaf

**! All ip addresses, vlan numbers, BGP ASNs, VNIs, Interface numbers are just examples and will change based on real environment.**

switchname <name>

**## Enable Features**

feature isis

feature pim **!optional, if PIM will be used for BUM replication.**

feature nv overlay

feature vn-segment-based

feature bgp

feature ngoam

feature lacp

feature nxapi

feature vpc

feature interface-vlan

feature fabric forwarding

feature ssh

feature tacacs

feature nxapi

feature telemetry

feature bfd

nxapi use-vrf management

ssh key rsa 2048 force

clock timezone UTC 2 0

clock summer-ime UTC 5 Sunday March 02:00 5 Sunday October 02:00 60

nv overlay evpn

**## Key Chain**

key chain *<name>*

key *<number>*

key-string *<string>*

system jumbomtu 9000

ip tcp mss 8960

**## Logging**

logging server x.x.x.x 5 use-vrf management

**!Raise L2FM logging**

logging level l2fm 5

**## DNS**

no ip domain-lookup

**## AAA**

username *admin* password <> role network-admin

aaa authentication login default group DFI\_TACACS

aaa authentication login console local

aaa authorization commands default group DFI\_TACACS local

aaa accounting default group DFI\_TACACS

**## TACACS**

tacacs-server timeout 1

tacacs-server host <ACSserver1> key <key>

tacacs-server host <ACSserver2> key <key>

tacacs-server directed-request

aaa group server tacacs+ DFI\_TACACS

server <ACSserver1>

server <ACSserver2>

use-vrf management

source-interface mgmt0

!

**## Management Interface**

ip access-list MGMT0\_ACL

remark #### TACACS

permit tcp <mgmt-scope> eq tacacs any

remark #### RADIUS

permit udp <mgmt-scope> eq 1812 any

permit udp <mgmt-scope> eq 1813 any

remark #### NTP

permit udp <mgmt-scope> eq ntp any

remark #### FTP

permit tcp <mgmt-scope> any eq 20

permit tcp <mgmt-scope> eq 20 any

permit tcp <mgmt-scope> any eq 21

permit tcp <mgmt-scope> eq 21 any

remark #### TFTP

permit udp <mgmt-scope> any eq tftp

permit udp <mgmt-scope> eq tftp any

remark #### SSH

permit tcp <mgmt-scope> any eq 22

permit tcp <mgmt-scope> eq 22 any

remark #### SNMP

permit udp <mgmt-scope> any eq snmp

remark #### ICMP

permit icmp any any echo

permit icmp any any echo-reply

statistics per-entry

remark ### VPC Keepalive

permit udp any any eq 3200

!

vrf context management

ip route 0.0.0.0/0 <default-gw>

interface mgmt0

description MGMT OOB

ip access-group MGMT0\_ACL in

vrf member management

ip address <address>/<mask>

!

line vty

access-class MGMT0\_ACL in

**## NTP**

ntp server x.x.x.x use-vrf management

ntp source-interface mgmt0

!

ip access-list NTP-ACCESS\_IN

permit ip <snmp host1> any

permit ip <snmp host2> any

permit ip <ntp server> any

deny ip any any

!

ntp access-group peer NTP-ACCESS\_IN

ntp authentication-key <number> md5 <string>

ntp server x.x.x.x key <number>

ntp trusted-key <number>

ntp authenticate

**## SNMP**

ip access-list SNMP-ACCESS\_IN

permit ip <snmp host1> any

permit ip <snmp host2> any

deny ip any any

!

snmp-server globalEnforcePriv  
!  
snmp-server user <username> <role> auth [md5|sha] <password> priv as- 128 <priv-pwd>  
!  
snmp-server location <location>  
snmp-server contact <contact>  
!  
snmp-server host <SNMP-Server> version 3 priv <username>  
snmp-server host <SNMP-Server> use-vrf management  
!  
snmp-server source-interface trap mgmt0  
snmp-server enable traps

snmp-server enable traps bgp  
!  
snmp-server community <community> group network-admin  
snmp-server community <community> use-ipv4acl SNMP-ACCESS\_IN

**## STP**

spanning-tree mst configuration

name *<name>*

revision *<vlaue>*

instance 1 vlan *<range>*

spanning-tree mode mst

**! Enabling STP edge port globally, enables adge port for all access interfaces.**

spanning-tree port type edge bpduguard default

**## DAG MAC**

fabric forwarding anycast-gateway-mac <mac-address>

**## Optional, IP PIM**

**! Configure RP for mcast group on all Leafs, only if PIM is enabled**

ip pim rp-address <address> group-list <mcast group>

ip pim bfd **!only if PIM is enabled**

**## VXLAN**

system nve infra-vlans <vpc infra vlan>

**!The system vlan nve-overlay id command should be used for a VRF or a Layer-3 VNI (L3VNI) only. Do not use this command for regular VLANs or Layer-2 VNIs (L2VNI).**

system vlan nve-overlay id <vlan IDs>

**!Example**

system vlan nve-overlay id 800,900,1000,2000

**!Configuring VLAN and VXLAN VNI**

**!Preventing VLAN Hopping**

vlan dot1Q tag native

vlan 2

shutdown

name NATIVE

vlan <vlan-number>

vn-segment <vn-number>

**!Example**

Vlan 10

name inside-B1\_Vlan

vn-segment 20010

Vlan 20

name inside-B2\_Vlan

vn-segment 20020

Vlan 30

name outside-B1\_Vlan

vn-segment 20030

Vlan 40

name outside-B2\_Vlan

vn-segment 20040

Vlan 800

name inside-B1

vn-segment 30010

Vlan 900

name inside-B2

vn-segment 30020

Vlan 1000

name outside-B1

vn-segment 30030

Vlan 2000

name outside-B2

vn-segment 30040

!

**! Manually configure RD and RT for L2VNI**

evpn

vni <20000+VLANID> l2

rd *RID:L2VNI*

route-target both *ASN:L2VNI*

!

**!Create Infra VLAN**

vlan <number>

name BACKUP\_VLAN\_ROUTING\_NVE\_INFRA

**!Configuring VRF for VXLAN Routing**

vrf context <context-name>

vni <number>

rd *RID:L3VNI*

address-family ipv4 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

**!Example**

VRF context INSIDE-B1

vni 30010

rd *RID:L3VNI*

address-family ipv4 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

VRF context INSIDE-B2

vni 30020

rd *RID:L3VNI*

address-family ipv4 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

VRF context OUTSIDE-B1

vni 30030

rd *RID:L3VNI*

address-family ipv4 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

VRF context OUTSIDE-B2

vni 30040

rd *RID:L3VNI*

address-family ipv4 unicast

route-target both *ASN:L3VNI*

route-target both *ASN:L3VNI* evpn

**!Configure VRF overlay SVI for the VRF**

interface Vlan <number>

no shutdown

vrf member <context name>

mtu 9100

ip forward

**!Example**

interface vlan800

Description outside-B1-L3VNI

VRF member INSIDE-B1

mtu 9100

IP forward

interface vlan900

Description outside-B2-L3VNI

VRF member INSIDE-B2

mtu 9100

IP forward

interface vlan1000

Description outside-B1-L3VNI

VRF member OUTSIDE-B1

mtu 9100

IP forward

interface vlan2000

Description outside-B2-L3VNI

VRF member OUTSIDE-B2

mtu 9100

IP forward

**## VPC Domain**

vpc domain <number>

peer-switch

**! Priority has to be provisioned different on pair of Leafs**

role priority {4096 | 8192}

peer-keepalive destination <address> source <address> vrf management eq 3200

auto-recovery reload-delay 360

delay restore 150

delay restore interface-vlan 10

peer-gateway

layer3 peer-router

ip arp synchronize

!

**!Configuring VPC peer-link**

**!Minimum two interfaces should be part of vPC PL**

interface ethernet <number>/<x - x+1>

description vPC PL

switchport

channel-group <id> mode active

no shutdown

!

interface port-channel<id>

description vPC PL

switchport

switchport mode trunk

switchport trunk native vlan 2

switchport trunk allowed vlan <range>

spanning-tree port type network

vpc peer-link

**## Configuring an uplink interface towards spine**

Interface e1/x-y

Description \_Uplink\_To\_Spine

mtu 9100

ip address <address>/31

isis network point-to-point

isis authentication-type md5

isis authentication key-chain *<key-chain>*

ip router isis UNDERLAY

no ip redirects

bfd authentication keyed-sha1 <keyid id> key <ascii\_key>

bfd interval 300 min\_rx 300 multiplier 3

**!Below steps are optional in case PIM/PIM-BFD is enabled**

ip pim sparse-mode

ip pim bfd-instance

!

**## Create infra VLAN SVI**

interface Vlan<id>

no shutdown

no ip redirects

mtu 9100

ip address <address>/31

isis authentication-type md5 level-1

isis authentication key-chain *<key-chain>* level-1

ip router isis UNDERLAY

ip pim pim sparse-mode **!Optional if PIM is used for BUM replication**

**## Configure host/server SVI**

interface vlan <number>

description OVERLAY-interface-toward-servers

vrf member <context name>

IP address <address>

mtu <1500-9000>

fabric forwarding mode anycast-gateway

!

**!Example**

Interface VLAN 10

Description inside-B1\_vlan

VRF member INSIDE-B1

IP address 10.1.1.254/24

mtu <1500-9000>

fabric forwarding mode anycast-gateway

Interface VLAN 20

Description inside-B2\_vlan

VRF member INSIDE-B2

IP address 20.1.1.254/24

mtu <1500-9000>

fabric forwarding mode anycast-gateway

Interface VLAN 30

Description outside-B1\_vlan

VRF member OUTSIDE-B1

IP address 30.1.1.254/24

mtu <1500-9000>

fabric forwarding mode anycast-gateway

Interface VLAN 40

Description outside-B2\_vlan

VRF member OUTSIDE-B2

IP address 40.1.1.254/24

mtu <1500-9000>

fabric forwarding mode anycast-gateway

**## Router ID Looopback**

interface loopback 0

description IGP Loopback Router\_ID

ip address <address>/32

ip router isis UNDERLAY

ip pim sparse-mode **!only if PIM is used for BUM traffic**

!

**## VTEP Loopback**

interface loopback1

description VTEP\_loopback

ip address <address>/32

ip address <address>/32 secondary

ip router isis UNDERLAY

ip pim sparse-mode **!only if PIM is used for BUM traffic**

!

**## NVE Interface**

interface NVE 1

host-reachability protocol bgp

source-interface loopback1

source-interface hold-down-time 220

advertise virtual-rmac

**!enable ingress-replication globally on all L2VNIs, only if PIM not used for BUM replication**

global ingress-replication protocol bgp

**!enable arp suppression globally on all L2VNIs**

global suppress-arp

member vni <l3vni-number> associate-vrf

member vni <l2vni-number>

suppress-arp

mcast-group <mcast address> **!if PIM is used for BUM Replication**

**!Example**

member vni 30010 associate-vrf

member vni 30020 associate-vrf

member vni 30030 associate-vrf

member vni 30040 associate-vrf

member vni 20010

mcast-group <mcast address> **!if PIM is used for BUM Replication**

member vni 20020

mcast-group <mcast address> **!if PIM is used for BUM Replication**

member vni 20030

mcast-group <mcast address> **!if PIM is used for BUM Replication**

member vni 20040

mcast-group <mcast address> **!if PIM is used for BUM Replication**

!

**## Server port Configuration**

Interface ethernet x/y | port-channel x

description interface\_to\_server

switchport

mtu <1500-9000>

**! Configure port mode, either access or trunk**

**! configure STP edge port with port mode access**

switchport mode access

switchport mode access vlan <number>

**! configure STP edge trunk port with port mode trunk**

switchport mode trunk

switchport trunk native vlan 2

switchport trunk allowed vlan <range>

**! Enable edge trunk if port is configured in Trunk mode**

spanning-tree port type edge trunk

**! Enable vpc orphan-port suspend on all non vPC (Orphan) ports**

vpc orphan-port suspend

**! Configure storm control**

storm-control broadcast level <level-value %>

storm-control multicast level <level-value %>

storm-control unicast level <level-value %>

**! Configure storm control action**

storm-control action trap

!

**## InterSwitch port Configuration**

Interface ethernet x/y | port-channel x

description interface\_to\_server

switchport

mtu <1500-9000>

**! configure trunk port with port mode trunk**

switchport mode trunk

switchport trunk native vlan 2

switchport trunk allowed vlan <range>

**! Configure storm control**

storm-control broadcast level <level-value %>

storm-control multicast level <level-value %>

storm-control unicast level <level-value %>

**! Configure storm control action**

storm-control action trap

!

**## VPC port configuration**

Interface ethernet x/y

channel-group x mode active

Interface port-channel x

!!! **only if edge/edge trunk port**

lacp vpc-convergence

vpc x

!

**## ISIS**

router isis UNDERLAY

log-adjacency-changes

net <network-entity-title>

is-type level-1

set-overload-bit on-startup 60

max-lsp-lifetime 65535

spf-interval level-1 5000

lsp-gen-interval level-1 5000

address-family ipv4 unicast

maximum-paths <number>

bfd

authentication-type md5 level-1

authentication key-chain <key> level-1

authentication-check level-1

**## BGP**

**!specify the ASN notation to asdot format**

as-format asdot

router bgp *<AS-number>*

log-neighbor-changes

address-family l2vpn evpn

**!requires advertise virtual-rmac in NVE**

advertise-pip

router-id *<address of loopback0>*

**!create peer-template to peer with the RR/Spine**

template peer EVPN-RR

inherit peer-session EVPN-SESSION

address-family l2vpn evpn

inherit peer-policy EVPN-RR-POLICY 10

**!create peer-session**

template peer-session EVPN-SESSION

password 3 *<pwd>*

update-source loopback0

remote-as *<as-number>*

**!create peer-policy**

template peer-policy EVPN-RR-POLICY

send-community

send-community extended

**!configure neighborship to RR/spine**

neighbor <RR/spine-address>

inherit peer EVPN-RR

vrf <vrf-name>

address-family ipv4 unicast

maximum-paths ibgp 2

**## ngoam**

ngoam install acl

ngoam profile 1

oam-channel 2

ngoam authentication-key <key-string>

**## DHCP Relay Config**

**Client on tenant VRF (SVI A) and server on the same tenant VRF (SVI B)**

feature dhcp

service dhcp

ip dhcp relay

ip dhcp relay information option

ip dhcp relay information option vpn

ipv6 dhcp relay

**! dedicated Loopback interface;**

interface loopback2

vrf member vxlan-900001

ip address <address>/32

interface Vlan<id>

description DHCP\_Server\_vrf\_vxlan-900001

no shutdown

vrf member vxlan-900001

ip address <address>/24

interface Vlan<id>

description Client\_Tenant\_vrf\_vxlan-900001

**! DHCP Server address is 192.1.42.3 in VRF vxlan-900001**

ip dhcp relay address 192.1.42.3

ip dhcp relay source—interface loopback2

ip prefix-list PL-loopback-dhcp seq 5 permit x.x.x.0/24 eq 32

route-map RM-direct-to-bgp permit 100

match PL-loopback-dhcp

router bgp xxxxx

vrf vxlan-900001

address-family ipv4 unicast

redistribute direct route-map RM-direct-to-bgp

## Border/Service Leaf Configuration Template

Configuration 71– Configuring Template for Border/Service Leaf

**!Configuraion is similar to Access Leaf template, in addition to the below sections which represent the peering with External Routers, and the Peering with FW**

**! All ip addresses, vlan numbers, BGP ASNs, VNIs, Interface numbers are just examples and will change based on real environment.**

**! Configs on this section are based on user VLANs 10,20,30,40 and user VRFs INSIDE-B1, INSIDE-B2, OUTSIDE-B1, OUTSIDE-B2 which were already created on Access Leaf Template**

**## Section 1: Peering with FW**

**!Inside Interface connecting to FW**

Interface ethernet x/y

Switchport

mtu *<1500-9000>*

switchport mode trunk

spanning-tree port type edge trunk

switchport trunk native vlan 2

switchport trunk allowed vlan 10,20

channel-group x mode active

Interface port-channel x

switchport

switchport mode trunk

switchport trunk native vlan 2

switchport trunk allowed vlan 10,20

spanning-tree port type edge trunk

lacp vpc-convergence

vpc x

**!Outside Interface connecting to FW**

Interface ethernet x/z

Switchport

mtu *<1500-9000>*

switchport mode trunk

spanning-tree port type edge trunk

switchport trunk native vlan 2

switchport trunk allowed vlan 30,40

channel-group x mode active

Interface port-channel xz

switchport

switchport mode trunk

switchport trunk native vlan 2

switchport trunk allowed vlan 30,40

spanning-tree port type edge trunk

lacp vpc-convergence

vpc x

!

**! Option 1: eBGP Routing to Firewalls**

**! Configuration below doesn’t include L2VNI and L3VNI config, they should be created already.**

system nve infra-vlans 3967

Interface loopback101

vrf member INSIDE-B1

ip address 172.16.1.253/32

Interface loopback102

vrf member INSIDE-B2

ip address 172.18.1.253/32

Interface loopback103

vrf member OUTSIDE-B1

ip address 172.20.1.253/32

Interface loopback104

vrf member OUTSIDE-B2

ip address 172.22.1.253/32

**! Configure layer3 peer-router under VPC domain, this command is needed as dynamic routing is configured on SVI 3967 over VPC peerlink**

vpc domain <number>

layer3 peer-router

**!Create prefix-list to redistribute connected subnets**

ip prefix-list PL-DIRECT permit 0.0.0.0/0 eq 24

!

route-map RM-DIRECT permit 10

match ip address prefix-list PL-DIRECT

**!Create prefix-list for Filtering Host Routes advertisement to FW**

ip prefix-list PL-HOST-ROUTE seq 5 permit 0.0.0.0/0 eq 32

route-map RM-EXT-HOST-ROUTE-FILTER deny 10

match ip address prefix-list PL-HOST-ROUTE

route-map RM-EXT-HOST-ROUTE-FILTER permit 1000

router bgp 65000

vrf INSIDE-B1

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.16.1.253/32

**! Redistribute direct subnets to FW**

redistribute direct route-map RM-DIRECT

**! Enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall**

neighbor 10.1.1.1

remote-as 65002

update-source loopback101

ebgp-multihop <*number-of-hops*>

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

**! Filter out host routes**

route-map RM-EXT-HOST-ROUTE-FILTER out

vrf INSIDE-B2

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.18.1.253/32

**! Redistribute direct subnets to FW**

redistribute direct route-map RM-DIRECT

**! Enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall**

neighbor 20.1.1.1

remote-as 65002

update-source loopback102

ebgp-multihop <*number-of-hops*>

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

**! Filter out host routes**

route-map RM-EXT-HOST-ROUTE-FILTER out

vrf OUTSIDE-B1

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.20.1.253/32

**! Redistribute direct subnets to FW**

redistribute direct route-map RM-DIRECT

**! Enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall**

neighbor 30.1.1.1

remote-as 65002

update-source loopback103

ebgp-multihop <*number-of-hops*>

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

**! Filter out host routes**

route-map RM-EXT-HOST-ROUTE-FILTER out

vrf OUTSIDE-B2

**! Advertise eBGP peering loopback interface**

address-family ipv4 unicast

network 172.22.1.253/32

**! Redistribute direct subnets to FW**

redistribute direct route-map RM-DIRECT

**! Enable BGP to advertise the gateway IP in the EVPN Type-5 routes**

export-gateway-ip

**! peer with Firewall**

neighbor 40.1.1.1

remote-as 65002

update-source loopback104

ebgp-multihop <*number-of-hops*>

address-family ipv4 unicast

**! Allow learning prefixes originated from fabric through FW**

allowas-in

**! Filter out host routes**

route-map RM-EXT-HOST-ROUTE-FILTER out

**!Option 2: Static routing to FWs**

**!Static Routing to Firewalls With HMM Tracking**

**!Configuration below doesn’t include L2VNI and L3VNI config, they should be created already.**

track 10 ip route 10.1.1.1/32 reachability hmm

vrf member INSIDE-B1

track 20 ip route 20.1.1.1/32 reachability hmm

vrf member INSIDE-B2

track 30 ip route 30.1.1.1/32 reachability hmm

vrf member OUTSIDE-B1

track 40 ip route 40.1.1.1/32 reachability hmm

vrf member OUTSIDE-B2

VRF context INSIDE-B1

**! static route on VTEP pointing to Firewall next hop**

**! firewall VIP 10.1.1.1**

ip route 20.1.1.0/24 10.1.1.1 track 10

ip route 30.1.1.0/24 10.1.1.1 track 10

ip route 40.1.1.0/24 10.1.1.1 track 10

**! Example of default route instead**

ip route 0.0.0.0/0 10.1.1.1 track 10

VRF context INSIDE-B2

**! static route on VTEP pointing to Firewall next hop**

**! firewall VIP 20.1.1.1**

ip route 10.1.1.0/24 20.1.1.1 track 20

ip route 30.1.1.0/24 20.1.1.1 track 20

ip route 40.1.1.0/24 20.1.1.1 track 20

**! Example of default route instead**

ip route 0.0.0.0/0 10.1.1.1 track 10

VRF context OUTSIDE-B1

**! static route on VTEP pointing to Firewall next hop**

**! firewall VIP 30.1.1.1**

ip route 10.1.1.0/24 30.1.1.1 track 30

ip route 20.1.1.0/24 30.1.1.1 track 30

ip route 40.1.1.0/24 30.1.1.1 track 30

VRF context OUTSIDE-B2

**! static route on VTEP pointing to Firewall next hop**

**! firewall VIP 40.1.1.1**

ip route 10.1.1.0/24 40.1.1.1 track 40

ip route 20.1.1.0/24 40.1.1.1 track 40

ip route 30.1.1.0/24 40.1.1.1 track 40

**!Only where static route is used with FW, create prefix-list to match static routes towards FWs/LBs**

ip prefix-list PL-INSIDE-B1-STATIC permit x.x.x.x/y

ip prefix-list PL-INSIDE-B2-STATIC permit a.b.c.d/y

ip prefix-list PL-OUTSIDE-B1-STATIC permit x.x.x.x/y

ip prefix-list PL-OUTSIDE-B2-STATIC permit a.b.c.d/y

!

route-map RM-INSIDE-B1-STATIC-to-BGP permit

match ip address perfix-list PL-INSIDE-B1-STATIC

!

route-map RM-INSIDE-B2-STATIC-to-BGP permit

match ip address perfix-list PL-INSIDE-B2-STATIC

!

route-map RM-OUTSIDE-B1-STATIC-to-BGP permit

match ip address perfix-list PL-OUTSIDE-B1-STATIC

!

route-map RM-OUTSIDE-B2-STATIC-to-BGP permit

match ip address perfix-list PL-OUTSIDE-B2-STATIC

**! The static route is redistributed into BGP**

router bgp 65000

vrf INSIDE-B1

address-family ipv4 unicast

redistribute static route-map RM-INSIDE-B1-STATIC-to-BGP

**! If the route-map is matching a default route, add ‘default-information originate’ to advertise the default route. The redistribute command configured under the BGP process injects all the routes that exist in the Routing Table (permitted by a route-map) in the BGP RIB with the exception of the default route. To additional allow the default route to be installed from the source routing protocol in the BGP RIB, the command default-information originate is required.**

default-information originate

vrf INSIDE-B2

address-family ipv4 unicast

redistribute static route-map RM-INSIDE-B2-STATIC-to-BGP

**! If the route-map is matching a default route, add ‘default-information originate’ to advertise the default route. The redistribute command configured under the BGP process injects all the routes that exist in the Routing Table (permitted by a route-map) in the BGP RIB with the exception of the default route. To additional allow the default route to be installed from the source routing protocol in the BGP RIB, the command default-information originate is required.**

default-information originate

vrf OUTSIDE-B1

address-family ipv4 unicast

redistribute static route-map RM-OUTSIDE-B1-STATIC-to-BGP

vrf OUTSIDE-B2

address-family ipv4 unicast

redistribute static route-map RM-OUTSIDE-B2-STATIC-to-BGP

**## Section 2: External Connectivity to External Router**

**! Configs on this section are based on user VRFs OUTSIDE-B1, OUTSIDE-B2 which were alrready created on Access Leaf Template**

**!Option 1: External Connectivity with Static Routing**

**!The below configuration example shows how to configure static route towards external router, assuming BGP EVPN and L3VNI are already configured in the VXLAN fabric**

**! Configuration of Interface facing external router**

interface Ethernet1/x

no switchport

vpc orphan-port suspend

no shutdown

interface Ethernet1/x.2

encapsulation dot1q 2

vrf member OUTSIDE-B1

ip address 30.1.1.y/29

no shutdown

interface Ethernet1/x.3

encapsulation dot1q 3

vrf member OUTSIDE-B2

ip address 40.1.1.y/29

no shutdown

VRF context OUTSIDE-B1

**! static route on VTEP pointing to Edge Router as next hop**

IP route x.x.x.x/y 30.1.1.x

VRF context OUTSIDE-B2

**! static route on VTEP pointing to Edge Router as next hop**

ip route a.b.c.d/y 40.1.1.x

**!create prefix-list to match static routes towards external router**

ip prefix-list PL-OUTSIDE-B1-STATIC permit x.x.x.x/y

ip prefix-list PL-OUTSIDE-B2-STATIC permit a.b.c.d/y

route-map RM-OUTSIDE-B1-STATIC-to-BGP permit

match ip address perfix-list PL-OUTSIDE-B1-STATIC

!

route-map RM-OUTSIDE-B2-STATIC-to-BGP permit

match ip address perfix-list PL-OUTSIDE-B2-STATIC

**! The static routes are redistributed into BGP**

router bgp 65000

vrf OUTSIDE-B1

address-family ipv4 unicast

redistribute static route-map RM-OUTSIDE-B1-STATIC-to-BGP

vrf OUTSIDE-B2

address-family ipv4 unicast

redistribute static route-map RM-OUTSIDE-B2-STATIC-to-BGP

**! Option 2: External Connectivity with eBGP Routing**

**!The below configuration example shows how to peer with external router, assuming BGP EVPN and L3VNI are already configured in the VXLAN fabric.**

**! Configuration of Interface facing external router**

interface Ethernet1/3

Description connection\_to\_external\_router

no switchport

vpc orphan-port suspend

no shutdown

interface Ethernet1/3.2

encapsulation dot1q 2

vrf member OUTSIDE-B1

ip address 10.31.95.31/24

no shutdown

interface Ethernet1/3.3

encapsulation dot1q 3

vrf member OUTSIDE-B2

ip address 10.31.96.31/24

no shutdown

VRF context OUTSIDE-B1

**! static routes in this VRF pointing to FW/LB, please refer to previous config part for peering with FWs**

ip route e.f.g.h/y <next-hop-ip>

VRF context OUTSIDE-B2

**! static routes in this VRF pointing to FW/LB, please refer previous section for peering with FWs**

ip route e.f.g.h/y <next-hop-ip>

**!create prefix-list to match static routes towards FWs/LBs**

ip prefix-list PL-OUTSIDE-B1-STATIC permit x.x.x.x/y

ip prefix-list PL-OUTSIDE-B2-STATIC permit a.b.c.d/y

**!**

route-map RM-OUTSIDE-B1-Static-to-BGP permit

match ip address perfix-list PL-OUTSIDE-B1-STATIC

!

route-map RM-OUTSIDE-B2-Static-to-BGP permit

match ip address perfix-list PL-OUTSIDE-B2-STATIC

**!Create prefix-list for Filtering Host Routes advertisement to external router**

ip prefix-list PL-HOST-ROUTE seq 5 permit 0.0.0.0/0 eq 32

route-map RM-EXT-HOST-ROUTE-FILTER deny 10

match ip address prefix-list PL-HOST-ROUTE

route-map RM-EXT-HOST-ROUTE-FILTER permit 1000

**!Create prefix-list to redistribute connected subnets**

ip prefix-list PL-DIRECT permit 0.0.0.0/0 eq 24

!

route-map RM-DIRECT permit 10

match ip address prefix-list PL-DIRECT

**!eBGP Routing to External Routers, this step also includes redistributing static routes to BGP**

!

router bgp 65000

vrf OUTSIDE-B1

address-family ipv4 unicast

redistribute static route-map RM-OUTSIDE-B1-Static-to-BGP

redistribute direct route-map RM-DIRECT

neighbor 10.31.95.x

remote-as 65099

address-family ipv4 unicast

route-map RM-EXT-HOST-ROUTE-FILTER out

vrf OUTSIDE-B2

address-family ipv4 unicast

redistribute static route-map RM-OUTSIDE-B2-Static-to-BGP

redistribute direct route-map RM-DIRECT

neighbor 10.31.96.y

remote-as 65099

address-family ipv4 unicast

route-map RM-EXT-HOST-ROUTE-FILTER out

!

## Spine Configuration Template

Configuration 72– Configuring Template for Spine

**! All ip addresses, vlan numbers, BGP ASNs, VNIs, Interface numbers are just examples and will change based on real environment.**

switchname <name>

**## Enable Features**

feature isis

feature pim **!optional if PIM will be used for BUM replication**

feature fabric forwarding

feature nv overlay

feature vn-segment-vlan-based

feature bgp

feature lacp

feature nxapi

feature ngoam

feature tacacs

feature ssh

featture bfd

feature telemetry

nxapi use-vrf management

ssh key rrsa 2048 force

clock timezone UTC 2 0

clock summer-ime UTC 5 Sunday March 02:00 5 Sunday October 02:00 60

nv overlay evpn

**## Key Chain**

key chain *<name>*

key *<number>*

key-string *<string>*

no ip domain-lookup

system jumbomtu 9000

ip tcp mss 8960

**## Logging**

**!Refer to access leaf config**

**## DNS**

**!Refer to access leaf config**

**## AAA**

**!Refer to access leaf config**

**## TACACS**

**!Refer to access leaf config**

**## Management Interface**

**!Refer to access leaf config**

**## NTP**

**!Refer to access leaf config**

**## SNMP**

**!Refer to access leaf config**

**## Optional, IP PIM**

ip pim bfd **!optional if PIM is enabled**

**! Configure RP for mcast group on all spines, if PIM is enabled**

ip pim rp-address <address> group-list <mcast group>

**! Below config is only for Spines acting as PIM RPs and only if PIM is used for BUM traffic replication**

**! Loopback Interface (Anycast RP) on all RPs**

interface loopback 254

description Anycast RP

ip address <address>/32

ip pim sparse-mode

ip router isis UNDERRLAY

**! Anycast-RP Configuration on all RPs**

ip pim rp-address <rp address> group-list <mcast group>

ip pim anycast-rp < rp address > < RP-1 lo0 address >

ip pim anycast-rp < rp address > < RP-1 lo0 address >

**## ISIS**

router isis UNDERLAY

log-adjacency-changes

net <network-entity-title>

is-type level-1

set-overload-bit on-startup 60

max-lsp-lifetime 65535

spf-interval level-1 5000

lsp-gen-interval level-1 5000

address-family ipv4 unicast

maximum-paths <number>

bfd

authentication-type md5 level-1

authentication key-chain <key> level-1

authentication-check level-1

!

**## BGP**

**!specify the ASN notation to asdot format**

as-format asdot

router bgp *<AS-number>*

router-id *<address of loopback0>*

log-neighbor-changes

**!create peer template for RR client neighbors (local leafs)**

template peer EVPN-RR-CLIENT

inherit peer-session EVPN-SESSION

address-family l2vpn evpn

inherit peer-policy EVPN-RR-CLIENT-POLICY 10

**!create peer template for RRs in different sites (Medium and Small fabrics)**

template peer EVPN-RR

inherit peer-session EVPN-SESSION

address-family l2vpn evpn

inherit peer-policy EVPN-RR-POLICY 10

**!create template peer-session**

template peer-session EVPN-SESSION

password 3 *<pwd>*

update-source loopback0

remote-as *<as-number>*

**!create template peer-policy for RR clients**

template peer-policy EVPN-RR-CLIENT-POLICY

send-community

send-community extended

route-reflector-client

**!create template peer-policy for RRs in different sites (Medium and Small fabrics)**

template peer-policy EVPN-RR-POLICY

send-community

send-community extended

**!configure bgp peering with all leafs in the same site**

neighbor *<leaf address>*

description *<description>*

inherit peer EVPN-RR-CLIENT

**!configure bgp peering with all RRs in different sites for Medium and Small fabrics**

neighbor <remmote RR address>

description *<description>*

inherit peer EVPN-RR

!

**## Router ID Loopback**

interface loopback 0

description IGP Loopback Router\_ID

ip address <address>/32

ip router isis UNDERLAY

ip pim sparse-mode **!only if PIM is used for BUM traffic**

!

**## configuring an uplink interface towards super-spine in medium fabric or a spine in different site for small fabric**

Interface e1/x-y

Description \_Uplink\_To\_Spine/Superspine

mtu 9100

ip address <address>/31

isis network point-to-point

isis authentication-type md5

isis authentication key-chain *<key-chain>*

ip router isis UNDERLAY

bfd authentication keyed-sha1 <keyid id> key <ascii\_key>

bfd interval 300 min\_rx 300 multiplier 3

**!Below steps are optional in case PIM/PIM-BFD is enabled**

ip pim sparse-mode

ip pim bfd-instance

**## configuring a downlink interface towards leaf/BGW**

Interface e1/x-y

Description \_Downlink\_To\_Leaf

mtu 9100

ip address <address>/31

isis network point-to-point

isis authentication-type md5

isis authentication key-chain *<key-chain>*

ip router isis UNDERLAY

bfd authentication keyed-sha1 <keyid id> key <ascii\_key>

bfd interval 300 min\_rx 300 multiplier 3

**!Below steps are optional in case PIM/PIM-BFD is enabled**

ip pim sparse-mode

ip pim bfd-instance

**## ngoam**

ngoam install acl

ngoam profile 1

oam-channel 2

ngoam authentication-key <key-string>

!

## Super-spine Configuration Template

Configuration 73– Configuring Template for Super-spine

**! All ip addresses, vlans, BGP ASNs, VNIs, Interface numbers are only examples.**

switchname <name>

**## Enable Features**

feature isis

feature pim **!optional if PIM will be used for BUM replication.**

feature lacp

feature nxapi

feature tacacs

feature ssh

feature bfd

feature telemetry

nxapi use-vrf management

no ip domain-lookup

clock timezone UTC 2 0

clock summer-ime UTC 5 Sunday March 02:00 5 Sunday October 02:00 60

**## Key Chain**

key chain *<name>*

key *<number>*

key-string *<string>*

system jumbomtu 9000

ip tcp mss 8960

**## Logging**

**!Refer to access leaf config**

**## DNS**

**!Refer to access leaf config**

**## AAA**

**!Refer to access leaf config**

**## TACACS**

**!Refer to access leaf config**

**## Management Interface**

**!Refer to access leaf config**

**## NTP**

**!Refer to access leaf config**

**## SNMP**

**!Refer to access leaf config**

**## Optional, IP PIM**

ip pim bfd **!optional if PIM is enabled**

**! Configure RP for mcast group on all super-spines**

ip pim rp-address <address> group-list <mcast group>

!

**! Below config is only for SuperSpines acting as PIM RPs and only if PIM is used for BUM traffic replication**

**! Loopback Interface (Anycast RP) on all RPs**

interface loopback 254

description Anycast RP

ip address <address>/32

ip pim sparse-mode

ip router isis UNDERRLAY

**! Anycast-RP Configuration on all RPs**

ip pim rp-address <rp address> group-list <mcast group>

ip pim anycast-rp <rp address> <RP-1 lo0 IP>

ip pim anycast-rp <rp address> <RP-2 lo0 IP>

!

**## ISIS**

router isis UNDERLAY

log-adjacency-changes

net <network-entity-title>

is-type level-1

set-overload-bit on-startup 60

max-lsp-lifetime 65535

spf-interval level-1 5000

lsp-gen-interval level-1 5000

address-family ipv4 unicast

maximum-paths <number>

bfd

authentication-type md5 level-1

authentication key-chain <key> level-1

authentication-check level-1

!

**## Router ID Looopback**

interface loopback 0

description IGP Loopback Router\_ID

ip address <address>/32

ip router isis UNDERLAY

ip pim sparse-mode **!only if PIM is used for BUM traffic**

!

**## configuring an underlay interface towards a spine or a BGW Leaf**

Interface e1/x-y

Description \_Underlay\_

mtu 9100

ip address <address>/31

isis network point-to-point

isis authentication-type md5

isis authentication key-chain *<key-chain>*

ip router isis UNDERLAY

bfd authentication keyed-sha1 <keyid id> key <ascii\_key>

bfd interval 300 min\_rx 300 multiplier 3

**!Below steps are optional in case PIM/PIM-BFD is enabled**

ip pim sparse-mode

ip pim bfd-instance

!

# Appendix A: Acronym Listing or Full Glossary

|  |  |
| --- | --- |
| Term | Definition |
| AAA | Authentication, Authorization, and Accounting |
| AFI | Address Family Identifier |
| ARP | Address Resolution Protocol |
| AS | Autonomous System |
| ASN | Autonomous System Number |
| BFD | Bidirectional Forwarding Detection |
| BGP | Border Gateway Protocol |
| BGW | Border Gateway |
| BUM | Broadcast, Unknown Unicast and Multicast |
| BPDU | Bridge Protocol Data Unit |
| CE | Classical Ethernet |
| CFS | Cisco Fabric Services |
| CFSoE | Cisco Fabric Services over Ethernet |
| DAG | Distributed Anycast Gateway |
| DC | Data Center |
| DCI | Data Center Interconnect |
| DHCP | Dynamic Host Configuration Protocol |
| DNS | Domain Name System |
| DWDM | Dense Wavelength Division Multiplexing |
| eBGP | External BGP |
| ETHPM | Ethernet Port Manager |
| EVPN | Ethernet VPN |
| FHRP | First-Hop Redundancy Protocol |
| FEX | Fabric Extender |
| GE | Gigabit Ethernet |
| HA | High Availability |
| HIF | Host Interface |
| HMM | Host Mobility Manager |
| HSRP | Hot Standby Router Protocol |
| iBGP | Internal BGP |
| IGP | Interior Gateway Protocol |
| IP | Internet Protocol |
| IPv4 | Internet Protocol version 4 |
| IPv6 | Internet Protocol version 6 |
| IRB | Integrated Routing and Bridging |
| L2 | Layer 2 |
| L2VPN | Layer 2 Virtual Private Network |
| L3 | Layer 3 |
| LACP | Link Aggregation Control Protocol |
| LAG | Link Aggregation Group |
| LAN | Local Area Network |
| LLD | Low Level Design |
| MAC | Media Access Control |
| MACSec | MAC Security |
| MCEC | Multi-Chassis Ether Channel |
| MCT | Multi-Chassis Trunk, a vPC Peer-Link |
| MDT | Multi-Destination Tree |
| MD5 | Message-Digest 5 |
| MP-BGP | Multi-Protocol Border Gateway Protocol |
| MTU | Maximum Transmission Unit |
| NGOAM | OAM for use in VXLAN |
| NIC | Network Interface Card |
| NX-OS | Nexus-series Operating System |
| OAM | Operations, Administration and Maintenance |
| OOB | Out-Of-Band |
| OSFP | Open Shortest Path First |
| PDU | Protocol Data Unit |
| PIM | Protocol-Independent Multicast |
| PIP | Primary IP |
| PVST | Per-VLAN Spanning-Tree |
| QoS | Quality of Service |
| RPVST+ | Rapid Per-VLAN Spanning-Tree + |
| SAFI | Sub-Address Family Identifier |
| SLA | Service Layer Agreement |
| SNMP | Simple Network Management Protocol |
| STP | Spanning-Tree Protocol |
| SVI | Switched Virtual Interface |
| TCP | Transmission Control Protocol |
| UDP | User Datagram Protocol |
| VIP | Virtual IP |
| VLAN | Virtual Local Area Network |
| vPC | Virtual PortChannel |
| VPN | Virtual Private Network |
| VRF | Virtual Routing and Forwarding |
| VTEP | VXLAN Tunnel Endpoint |
| VXLAN | Virtual Extensible LAN |
| WAN | Wide Area Network |

# Appendix B : SFP Modules Matrix Compatibility

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Transceiver Product Family | Transceiver Product ID | N9K-C9336C-FX2 | N9K-C93240YC-FX2 | N9K-C93180YC-FX | N9K-C93108TC-FX | N9K-C9364C-GX | Form Factor | Reach | Cable Type | Media | Connector Type | Transceiver Type | Case Temperature | DOM Capable |
| QSFP100 | QSFP-100G-SR4-S | + | + | + | + | + | QSFP28 | 100m (OM4) | Ribbon Fiber | MMF | MPO-12 | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-SR4-S | - | + | - | - | + | QSFP28 | 70m (OM3) | Ribbon Fiber | MMF | MPO-12 | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-CWDM4-S | + | + | + | + | + | QSFP28 | 2km | Duplex Fiber | SMF | LC | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-PSM4-S | + | + | + | + | + | QSFP28 | 500m | Ribbon Fiber | SMF | MPO-12 | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-LR4-S | + | + | + | + | + | QSFP28 | 10km | Duplex Fiber | SMF | LC | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-SM-SR | + | + | + | + | + | QSFP28 | 2km | Duplex Fiber | SMF | LC | Optic | 10 to 60C | Y |
| QSFP100 | QSFP-100G-ER4L-S | + | + | + | + | - | QSFP28 | 40km | Duplex Fiber | SMF | LC | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-FR-S | - | - | - | - | + | QSFP+ | 2km | Duplex Fiber | SMF | LC | Optic | 0 to 70C | Y |
| QSFP100 | QSFP-100G-CU1M | + | + | + | + | + | QSFP28 | 1m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-CU2M | + | + | + | + | + | QSFP28 | 2m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-CU3M | + | + | + | + | + | QSFP28 | 3m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-CU5M | + | + | + | + | + | QSFP28 | 5m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC1M | + | + | + | + | + | QSFP28 | 1m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC2M | + | + | + | + | + | QSFP28 | 2m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC3M | + | + | + | + | + | QSFP28 | 3m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC5M | + | + | + | + | + | QSFP28 | 5m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC7M | + | + | + | + | - | QSFP28 | 7m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC10M | + | + | + | + | + | QSFP28 | 10m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC15M | + | + | + | + | + | QSFP28 | 15m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC20M | + | + | + | + | - | QSFP28 | 20m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC25M | + | + | + | + | + | QSFP28 | 25m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-100G-AOC30M | + | + | + | + | + | QSFP28 | 30m | N/A (Incl AOC and DAC) | AOC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-4SFP25G-CU1M | + | - | + | + | + | QSFP28 | 1m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-4SFP25G-CU2M | + | - | + | + | + | QSFP28 | 2m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-4SFP25G-CU3M | + | - | + | + | + | QSFP28 | 3m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |
| QSFP100 | QSFP-4SFP25G-CU5M | + | - | - | + | + | QSFP28 | 5m | N/A (Incl AOC and DAC) | DAC | N/A | Cable | 0 to 70C | N |

Trademarks and Disclaimers

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THIRD PARTY SUPPLIERS ARE PROVIDED “AS IS” WITH ALL FAULTS. CISCO AND THIRD PARTY SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

©2019 Cisco Systems, Inc. All rights reserved.

Document Acceptance

|  |  |  |  |
| --- | --- | --- | --- |
| Name |  | Name |  |
| Title |  | Title |  |
| Company |  | Company |  |
| Signature |  | Signature |  |
| Date |  | Date |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Name |  | Name |  |
| Title |  | Title |  |
| Company |  | Company |  |
| Signature |  | Signature |  |
| Date |  | Date |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Name |  | Name |  |
| Title |  | Title |  |
| Company |  | Company |  |
| Signature |  | Signature |  |
| Date |  | Date |  |