Cisco Dynamic Fabric Automation Migration Guide

First Published: January 31, 2014

Last Modified: June 16, 2015

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

Text Part Number: OL-31503-02

CONTENTS

Preface

The Preface contains the following sections:

Audience

This publication is for experienced network administrators who configure and maintain Cisco Dynamic Fabric Automation.

Document Organization

This document is organized into the following chapters:

Chapter	Description
"Information About Cisco DFA"	Provides an overview of Cisco Dynamic Fabric Automation (DFA) and descriptions of the Cisco DFA building blocks.
"Migration to Cisco DFA"	Provides information about how to prepare for migration to Cisco DFA, including migration steps and migration configuration.
"Troubleshooting Migration"	Provides the verification steps on how to troubleshoot Cisco DFA migration.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

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

This document uses the following conventions:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Documentation for Cisco Dynamic Fabric Automation

The Cisco Dynamic Fabric Automation documentation is at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/dynamic-fabric-automation/tsd-products-support-series-home.html.

The Cisco Nexus 6000 Series documentation is at the following URL: http://www.cisco.com/c/en/us/support/switches/nexus-6000-series-switches/tsd-products-support-series-home.html.

The Cisco Nexus 7000 Series documentation is at the following URL: http://www.cisco.com/c/en/us/support/switches/nexus-7000-series-switches/tsd-products-support-series-home.html.

The Cisco Nexus 5500 and 5600 Series documentation is at the following URL: http://www.cisco.com/c/en/us/support/switches/nexus-5000-series-switches/tsd-products-support-series-home.html.

The Cisco Nexus 1000V switch for VMware vSphere documentation is at the following URL: http://www.cisco.com/en/US/products/ps9902/tsd_products_support_series_home.html. The documentation therein includes the following guides for Cisco DFA. Additional information pertaining to troubleshooting can be located in the Cisco Nexus 1000V documentation for Cisco NX-OS Release 4.2(1)SV2(2.2).

- *Cisco Nexus 1000V DFA Configuration Guide, Release 4.2(1)SV2(2.2)*
- *Cisco Nexus 1000V VDP Configuration Guide, Release 4.2(1)SV2(2.2)*

The Cisco Prime Data Center Network Manager (DCNM) documentation is at the following URL: http://www.cisco.com/en/US/products/ps9369/tsd_products_support_series_home.html. The Cisco Prime DCNM documentation for Cisco DFA includes but is not limited to the following guides:

- *Cisco DCNM 7.0 OVA Installation Guide.*
- *Cisco DCNM 7.0 Fundamentals Guide*
- *Cisco DCNM DFA REST 7.0 API Guide*

The Cisco Prime Network Services Controller (NSC) documentation is at the following URL: http://www.cisco.com/en/US/products/ps13213/tsd_products_support_series_home.html.

The OpenStack for Cisco DFA install documentation includes the following guide and documents:

- *Open Source Used In OpenStack for Cisco DFA 1.0* at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/dfa/openstack/opensource/OpenStack_for_Cisco_DFA_1.0_Open_Source_Documentation.pdf
- *OpenStack for Cisco DFA Install Guide Using Cisco OpenStack Installer* at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/dfa/openstack/install/guide/os-dfa-coi.pdf

- *OpenStack for Cisco DFA Install Guide for Using Pre-built OpenStack for Cisco DFA Images* at the following URL: http://www.cisco.com/c/dam/en/us/td/docs/switches/datacenter/dfa/openstack/install/guide/preblt-image.pdf
- *Quick Guide to Clonezilla* at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/dfa/openstack/install/guide/clonezilla-image-restore.pdf

Documentation Feedback

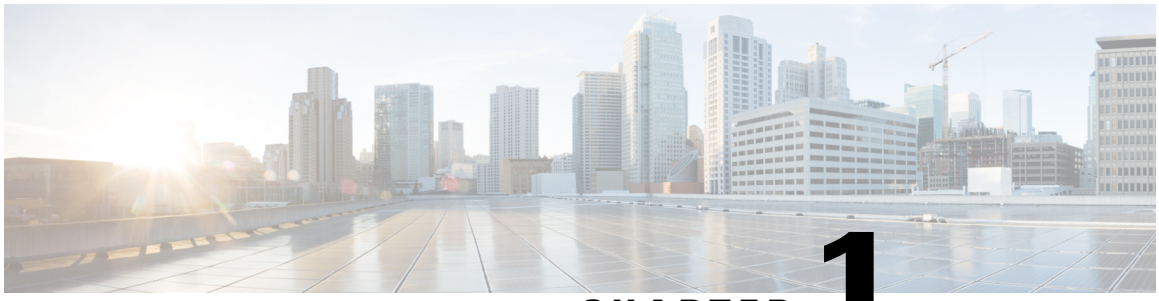
To provide technical feedback on this document, or to report an error or omission, please send your comments to: ciscodfa-docfeedback@cisco.com.

We appreciate your feedback.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation*, at: http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.



CHAPTER 1

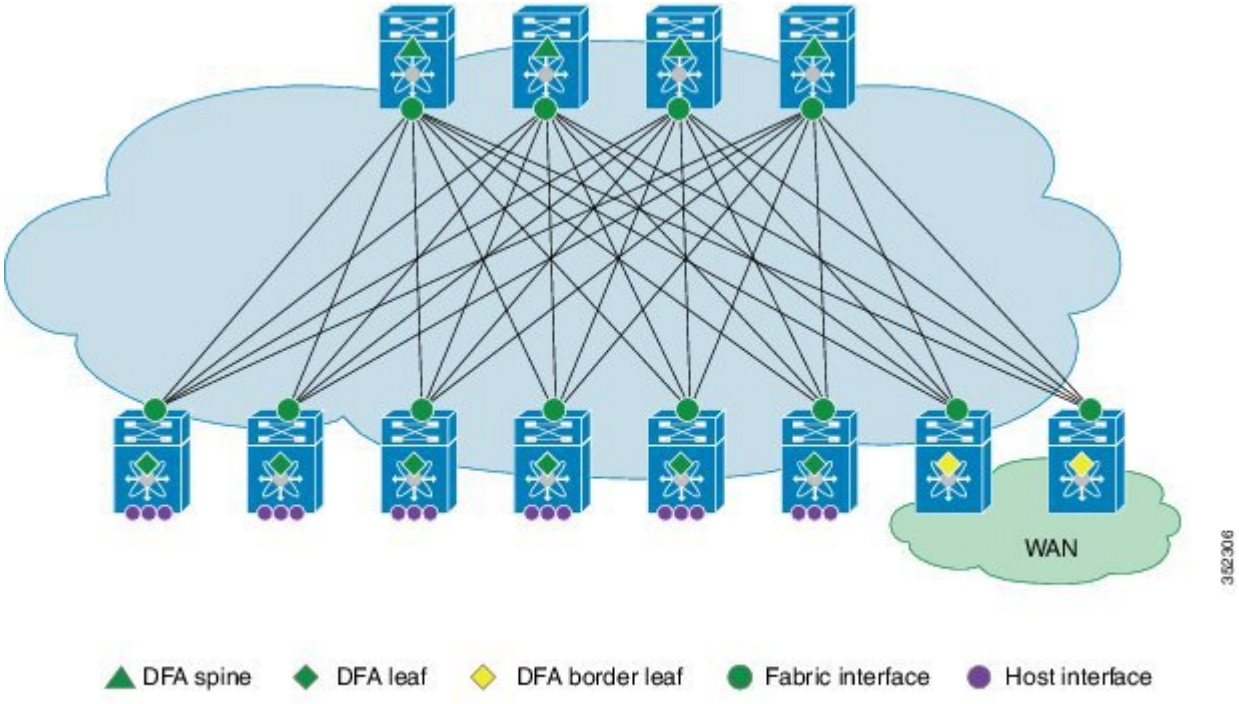
Information About Cisco DFA

This chapter includes the following sections:

Terminology

The following figure shows the terms that are used for a Cisco Dynamic Fabric Automation (DFA) deployment. You should understand these terms and definitions before you deploy Cisco DFA.

Figure 1: Terms Used in a Cisco Dynamic Fabric Automation Deployment



- Cisco DFA fabric—A multistage, switching network in which every connected device is reachable through the same number of hops. The Cisco DFA fabric enables the use of a scale-out model for optimized growth.
- Cisco DFA switch—A leaf, border leaf, or spine device.
- Leaf—Switches with ports that are connected to ethernet devices, such as servers (host interfaces) and ports (fabric interfaces), that are connected to the Cisco DFA fabric. Leaf switches forward traffic based on the enhanced control plane functionality of Cisco DFA optimized networking, which requires segment ID-based forwarding.
- Border leaf—Switches that connect external network devices or services, such as firewalls and router ports, to a Cisco DFA fabric. Border leaf switches are similar to leaf switches and can perform segment ID-based forwarding.
- Spine—Switches through which all leaf and border leaf switches are connected to each other and to which no end nodes are connected. Spine switches forward traffic based on Cisco DFA optimized networking with enhanced or traditional forwarding.

- Host interface—Leaf to server interfaces that receive traffic for connected VLANs to be extended across the Cisco DFA fabric.
- Fabric interface—Ports through which Cisco DFA switches are connected to one another.

Cisco Dynamic Fabric Automation Overview

Cisco Dynamic Fabric Automation optimizes data centers through integration. This architecture eliminates the need for overlay networks that can hinder traffic visibility and optimization and reduce scalability when physical server and virtual machine environments are integrated. The architecture enables zero-touch provisioning and greater orchestration, while delivering more predictable performance and latency for large cloud networks. The following building blocks are the foundation of Cisco DFA:

- Fabric Management—Simplifies workload visibility, optimizes troubleshooting, and automates fabric component configuration.
- Workload Automation—Integrates with automation and orchestration tools through northbound application programming interfaces (APIs) and also provides control for provisioning fabric components by automatically applying templates that leverage southbound APIs and standard-based protocols. These automation mechanisms are also extensible to network services.
- Optimized Networking—Uses a simple distributed gateway mechanism to support any subnet, anywhere, concurrently. Existing redundancy models are also used to provide N+ redundancy across the entire fabric.
- Virtual Fabrics—Extends the boundaries of segmented environments to different routing and switching instances by using logical fabric isolation and segmentation within the fabric. All of these technologies can be combined to support hosting, cloud, and multi-tenancy environments.
- DCI Automation—Automate the configuration of connecting tenants within the unified fabric to the external world, be it the Internet or other unified fabric networks. These features works in tandem with DCNM (7.1.1 onwards) to enable auto configuration of such requirement.

Note Global VLAN mutually exclude segment ID, (at least for Layer-2 Traffic). A segment ID is a global identifier, there cannot be two global identifier = VLAN + segment ID, you have to decide one or the other. Global VLANs and segment ID can co-exist in the same fabric, if the outer header is not overlapping.

Fabric Management

The fabric management network in Cisco Dynamic Fabric Automation represents a dedicated out-of-band network that is responsible for bootstrapping and managing the individual networking devices, such as spines, leafs, and border leaf switches that are controlled by fabric management. The fabric management network is responsible for transporting the protocols that are required for the different fabric management functions. The following table lists the functions and protocols across the fabric management network.

Table 1: Functions and Protocols Across the Fabric Management Network

Function	Protocol
Power On Auto provisioning (POAP) for automatically configuring network devices	• Dynamic Host Configuration Protocol (DHCP) • Trivial File Transfer Protocol (TFTP) • Secure Copy Protocol (SCP)
Fabric discovery	Simple Network Management Protocol (SNMP)
User-to-machine and machine-to-machine communication	Extensible Messaging and Presence Protocol (XMPP)
Automated network provisioning	Lightweight Directory Access Protocol (LDAP)
DCI Automation	Auto Provisioning of Data Center Interconnect on a border leaf.

The management network, also known as the management access, is the network administrator-facing interface for accessing fabric management. The management network represents the portion of your network from which you, as the network administrator, can connect to an element manager or a network management station (NMS) and to switches and routers.

The Cisco Prime Data Center Network Manager (DCNM) is a turn-key management system for fabric management, visibility, and an extensible set of functions to more efficiently control the data center fabric. Cisco Prime DCNM uses standards-based control protocol components to provide you with an extensive level of customization and integration with an operations support system (OSS) network.

Cisco Prime Data Center Network Manager

An Open Virtual Appliance (OVA) is a prebuilt software solution that comprises one or more virtual machines (VMs) that are packaged, maintained, updated, and managed as a single unit. The Cisco DCNM OVA includes an application functionality that is necessary for Cisco DFA. Cisco Prime DCNM as an OVA can be deployed on a VMware vSphere infrastructure.

Cisco Prime DCNM provides the following functionalities:

- Device auto configuration is the process of bringing up the Cisco DFA fabric by applying preset configuration templates to any device that joins the fabric. Auto configuration installs an image or applies the basic configuration.
- Cable-plan consistency checks the physical connectivity of the fabric against a documented cable-plan for compliance. The lack of compliance prevents specific links from being active and protects the fabric from unwanted errors.
- Common point of fabric access allows you, as a network administrator, to interact with the fabric as a single entity (system) to simplify queries and to eliminate switch by switch troubleshooting efforts.

- Automated network provisioning provides a new layer of automation integration in which the data center fabric-switching infrastructure is automatically provisioned for the physical or virtual workload that is being instantiated.
- Automated profile refresh allows keeping the fabric and the network information in sync in a non-disruptive manner.
- DCI Automation provides a touchless provisioning of datacenter interconnections for the tenants.
- Network, virtual fabric, and host visibility is provided by the management GUI and displays a single set of active network elements that belong to an organization in the fabric.

The Cisco DFA DCNM access network is the network administrator facing interface for accessing fabric management and for connecting northbound application program interfaces (APIs) to orchestrators.

Automated Network Provisioning

Cisco DFA fabric automatically provisions tenant networks using a database of network information. Network information database can be looked up using either tenant's traffic information or by VSI Discovery Protocol (VDP) running on the connected Vswitches. The network information database can be stored and managed using Cisco Prime DCNM. This makes it possible for a complete tenant VM orchestration with automated network provisioning to be absolutely touchless from the fabric perspective. For more information on tenant provisioning, see http://www.cisco.com/c/en/us/td/docs/switches/datacenter/dfa/configuration/b-dfa-configuration.html.

Mobility Domain

In a fabric, when auto-configuration is done using tenant's traffic, the dot1q from the traffic is used to locate the network information. Dot1Q is always used with a notion of mobility domain. A mobility domain represents a set of network ports in the fabric where dot1q is treated symmetrically.

From 7.1.x release, each network interface of a leaf can be configured with a mobility domain in addition to global leaf mobility domain configuration. By translating tenant's dot1Q values to internal leaf dynamic VLANs, true multi-tenancy is achieved with touchless orchestration. A tenant can orchestrate its own range of server VLANs without the need for coordinating the VLAN usage in the fabric. However, with Cisco Nexus 55XX Series Switches as a leaf, mobility domain can only be specified global to the leaf and no translation is possible. For more information on configuration details, see http://www.cisco.com/c/en/us/td/docs/switches/datacenter/dfa/configuration/b-dfa-configuration.html.

VDP-Based Configuration

When a Vswitch connected to the network port is VDP (Virtual station interface Discovery Protocol) capable, VDP can be used to learn segment information of the connected virtual machines in a reliable out of band manner. The segment information being global to the fabric is alone to look up to the network information. In this method, the leaf communicates a dynamically allocated VLAN to the Vswitch through the VDP messages. VDP protocol implementation is based on IEEE standard 802.1QBG. Nexus 1000V and an open source LLDPAD application (for OpenStack) have this VDP implementation.

From release 7.1 onwards, VDP can be used for virtual machines that are provisioned in a VLAN network without using the segment. VDP can also be enabled on Cisco Nexus 55XX Series Switches.

Simplified Profile Management

Network information is stored as a set of parameters in the database; these parameters are then applied to the desired profile to achieve a configuration set for a particular tenant network. Each network can be mapped to its own profile; for example, a network may need only IPv4 parameters and hence it can use a default NetworkIpv4EfProfile and a certain network may use both, where it will use its own profile. Since the 7.1 release, fabric supports universal profiles, where certain parameters can be left empty. If a particular network does not need IPv6 parameters, they can be left unfilled while the profile still contains configuration related to IPv6. This hugely simplifies profile management as only a few profiles will accomplish multiple needs. Also, profile refresh with universal profiles fabric and the network information will be in synchronization in a non-disruptive manner.

CLI-Based Auto-Configuration

Cisco DFA supports a command-line interface (CLI) based auto-configuration for pre-provisioning network devices. The auto-configuration is the same as any configuration that is based on network triggers such as data packet and Virtual Discovery Protocol (VDP). After an auto-configuration is created on a switch, you can use existing Cisco DFA commands, such as the **clear fabric database host** command, to manage the switch configuration.

Automation of Border Leaf L3 External Connectivity

This feature works in conjunction with DCNM (7.1.1 release) to enable auto-configuration of fabric external connectivity on a per-tenant basis. Enhancements have been made to UCSD 5.2 , OpenStack, border leaf POAP template, LDAP Schema, DCNM GUI, and on the switch-side software. These enhancements are done to automate the extension of the tenant towards the DC Edge router and optionally beyond to connect to other fabrics using a BGP MPLS VPN. The DFA 2.0 release completely automates the border leaf auto-configuration for the most common topologies that customers use to connect to the DC Edge box. The creation of the topology is enabled by enhancement to POAP templates for border leaf and a new POAP template is created for a Cisco Nexus 7000-based DC Edge box running a Cisco NX-OS 6.2(10) image. After these devices are booted up, they are imported into Cisco Prime DCNM. At the Cisco Prime DCNM, the imported devices are paired as per network design and assigned attributes such as maximum number of tenants to be deployed on them, the configuration profile associated with the extension. After the topology is complete at Cisco Prime DCNM, the auto-configuration can be globally enabled at Cisco Prime DCNM. At this point, the border leaf auto-configuration is ready for deployment of tenants. This extension can be initiated from the orchestrator (UCSD 5.2 or OpenStack 2). It can also be initiated from Cisco Prime DCNM itself. In Cisco NX-OS 6.2(10) release for Cisco Nexus 7000 platform, the configuration can be generated on Cisco Prime DCNM and copied and pasted manually on the N7000 DC edge device. Similar support is available for ASR9K. The N7000 border leaf (the HUB PE model) will also be supported with auto-configuration in the future releases of Cisco Prime DCNM and N7000. This feature is driven by Cisco Prime DCNM. You can refer to the *Cisco DCNM Fundamentals Guide, Release 7.x*.

After the network is ready for orchestration, the extension can be done by either UCSD or OpenStack. Similarly, the L3 extension can be removed from the orchestrator. For more details, refer to the *Cisco UCS Director Dynamic Fabric Automation Management Guide* and the *Openstack 2.0 User Guide*.

Optimized Networking

Optimized networking in Cisco DFA uses a simple distributed gateway mechanism to support any subnet, anywhere, concurrently.

Frame Encapsulation

Optimized networking in a Cisco DFA deployment uses Cisco FabricPath Frame Encapsulation (FE) for efficient forwarding based on a Shortest Path First (SPF) algorithm for unicast and multicast IP traffic. Host route distribution across the fabric is accomplished using a scalable multi-protocol Border Gateway Protocol (MP-BGP) control plane.

The Cisco DFA enhanced forwarding improves Cisco FabricPath FE by optimizing the conversational learning from Layer-2 to Layer-3. In addition to the enhanced control and data plane for unicast and multicast forwarding, Cisco DFA reduces the Layer-2 failure domain by having the Layer-2/Layer-3 demarcation on the host-connected leaf switch, which terminates the host-originated discovery protocols at this layer.

A distributed anycast gateway on all of the Cisco DFA leaf switches for a VLAN improves resilience and enables the fabric to scale to more hosts by keeping a shorter path for intra and inter-VLAN forwarding. Cisco DFA leaf switches that operate as border leaf switches interconnect the Cisco DFA fabric to external networks. Cisco DFA border leaf switches peer with external standard unicast and multicast routing protocols.

Dynamic VLAN Management

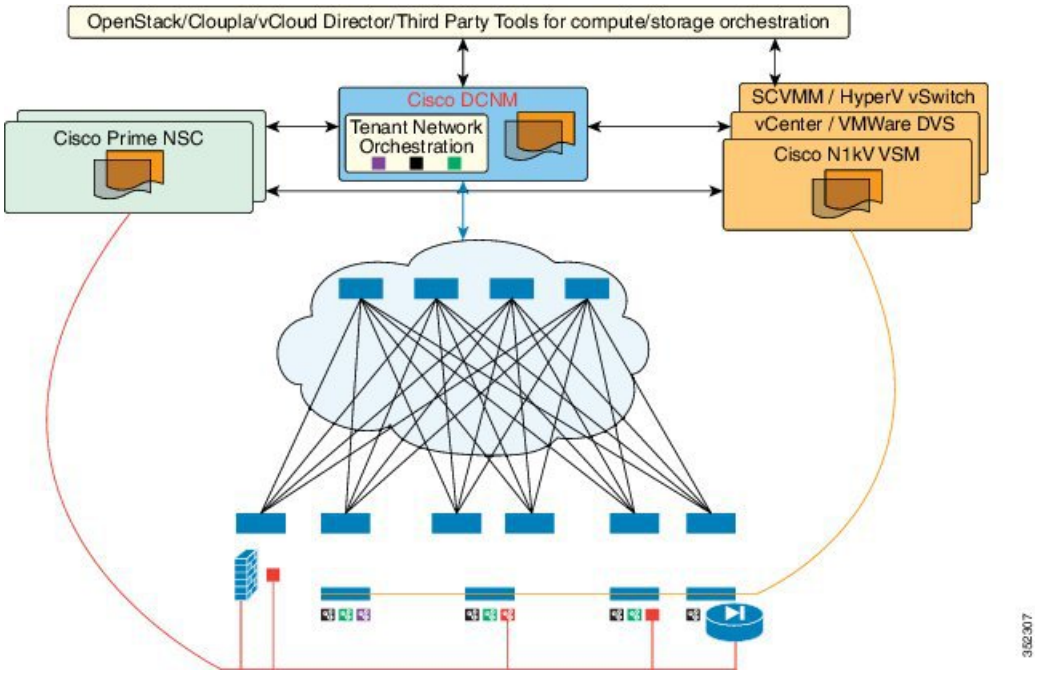
Managing VLANs that are used to interact with the servers is always complicated due to the need for more than 4K tenants. Fabric dynamic VLAN allocations can solve this problem. With a VDP-capable Vswitch, leafs can communicate with Vswitch using VDP and discover the presence of VMs. VDP can communicate segment information of a network to the leaf. The leaf then maps the segment to the next available VLAN. These allocated VLANs are communicated back to the Vswitch for use with the traffic that the VM sends out. A tenant or VM Orchestrator is completely unaware of the VLAN space that needs to be managed across all of the fabric. For a Vswitch that cannot communicate using VDP, a mobility domain can be specified for each network interface where a Vswitch is connected. Each mobility domain in a leaf can be mapped to a VLAN pool. When a tenant network is orchestrated for a particular dot1Q, the dot1Q is normalized to the next available VLAN in the leaf's VLAN pool for forwarding. The VLAN that is mapped can also be configured to carry tenant's traffic over the fabric using a segment. The number of tenant VMs that can be orchestrated under a leaf is drastically increased by enabling tenant VLANs only on the ports where the tenant is detected. When an auto-configuration of a tenant network is done for a network using either VDP or tenant's traffic, the leaf provisions the VLAN that is required for the tenant. The provisioned VLAN is brought up only on the port where the network was provisioned. Refer to the DFA Configuration guide for more details as described in the sections *Multiple Mobility Domain* and *Dynamic Virtual Port*.

Cisco Dynamic Fabric Automation Services Support

Services such as a firewall, load balancer, and virtual private networks (VPNs) are deployed at the aggregation layer in the traditional data center. In a Cisco DFA deployment, services nodes are deployed at regular leaf switches for both east-west and north-south traffic. Services can be physical or virtual services nodes.

The following figure shows the interaction between the Cisco Prime Network Services Controller (NSC) and the Cisco DFA deployment through Cisco Prime Data Center Network Manager (DCNM).

Figure 2: Cisco DFA with Services



The Cisco Prime NSC is the services orchestrator for Cisco DFA. The NSC Adapter in the Cisco Prime DCNM Open Virtual Appliance (OVA) performs the following functions:

- Provides connectivity between Cisco Prime DCNM and the Cisco Prime NSC services orchestrator
- Automatically populates the Cisco Prime NSC with the organizations, partitions, and networks that are created in Cisco Prime DCNM
- Populates Cisco Prime DCNM with the services that are stitched through Cisco Prime NSC
- Allows the use of multiple Cisco Prime NSC instances to match the Cisco Prime DCNM scale

Fabric can be provisioned for services using Cisco UCSD as well without using PNSC for certain scenarios. Containers can be used to orchestrate policies for tenant edge firewall using Physical ASA or ASAv. Containers are integrated with Cisco Prime DCNM to use DFA VLANs to create networks for a firewall's inside and outside interfaces. VSG service networks can also be orchestrated using UCSD; however, in this scenario, PNSC is required for provisioning the VSG. UCSD deploys all the virtual form factor service nodes (ASAv, VSG) using the port groups with DFA VLANs. These networks are also pushed to Cisco Prime DCNM through the Rest APIs. Note that interaction between PNSC and Cisco Prime DCNM is not needed for this approach; UCSD implements this functionality for services.

In Cisco DFA, configuration profile templates and instantiating the profiles on a leaf switch provide network automation. The templates are extended to support services in Cisco DFA. The profile templates are packaged in Cisco Prime DCNM for the services orchestrator. The table below includes a list of profile templates that are available for Cisco DFA services. It is important that you select the correct profile to orchestrate and automate services in the Cisco DFA fabric.

Table 2: Cisco Templates for Services Support

Service	Network	Routing	Service Profile
Edge Firewall	Host Network	N/A	defaultUniversalTfProfile
	Edge Firewall	Static	serviceNetworkUniversalTfStaticRoutingProfile
		Dynamic	serviceNetworkUniversalDynamicRoutingESProfile
	Tenant External Service Network	Static	externalNetworkUniversalTfStaticRoutingESProfile
		Dynamic	externalNetworkUniversalDynamicRoutingESProfile
Service Node as Router/Default Gateway	Host Network	N/A	defaultNetworkL2Profile

For NSC Adapter installation information, see the *Cisco DCNM 7.1 OVA Installation Guide*.

OpenStack for Cisco DFA

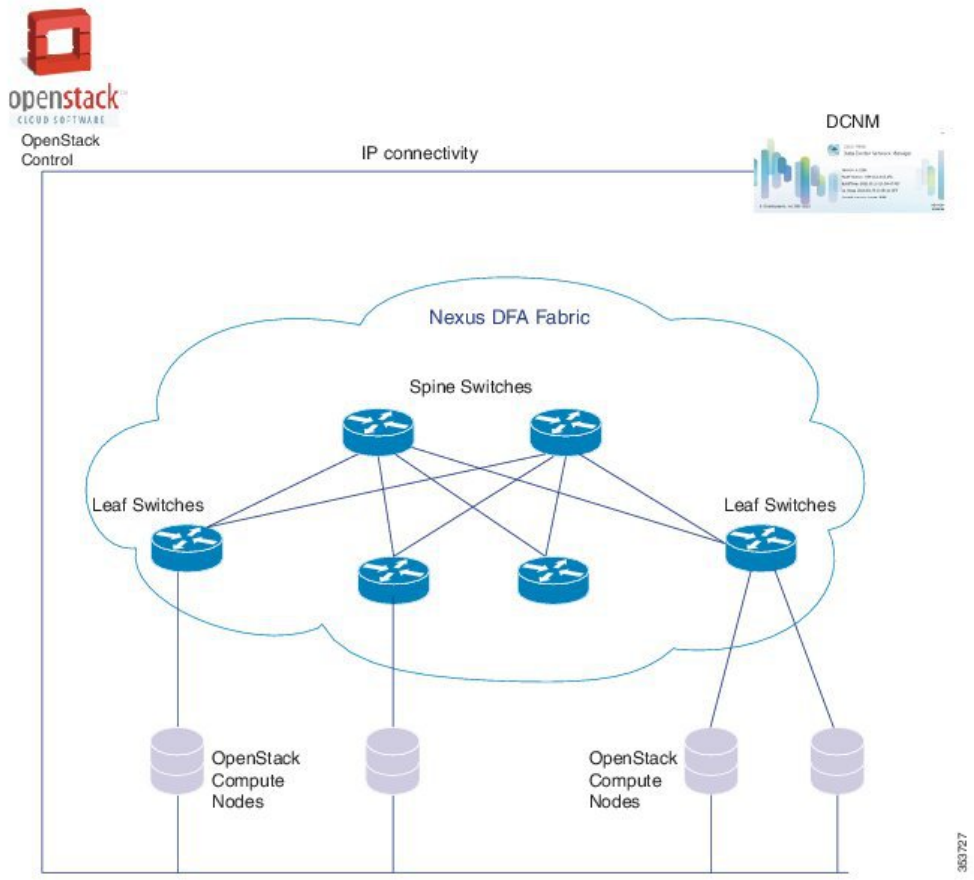
OpenStack creates a human and machine-accessible service for managing the entire life cycle of the infrastructure and applications within OpenStack clouds. The technology consists of a series of inter-related projects that control pools of processing, storage, and networking resources throughout a data center that can be managed or provisioned through a web-based dashboard, command line tools, a RESTful application programming interface (API), or Python scripts based on OpenStack Python SDK.

The OpenStack for Cisco DFA software is an application-level enabler that works with the latest Juno release. OpenStack for Cisco DFA provides orchestration of the cloud that is enabled by Cisco DFA. OpenStack for Cisco DFA provides orchestration of the cloud that is enabled by Cisco DFA.

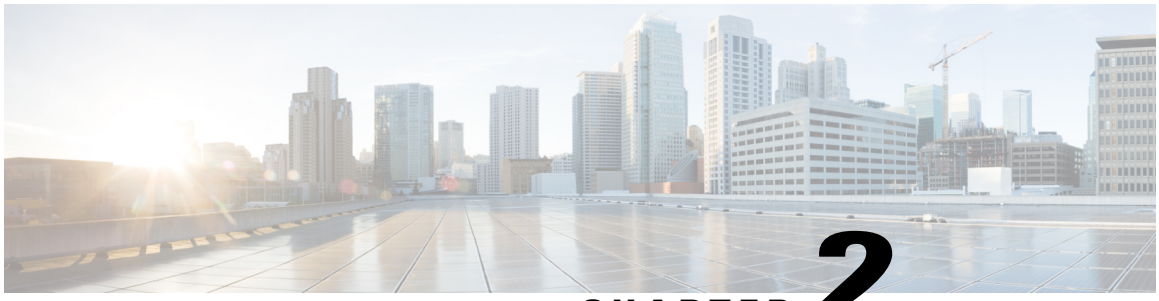
Users can choose to install OpenStack using their preferred mechanism on their chosen target servers. After the OpenStack installation, the lightweight DFA enabler installation will make the OpenStack DFA ready. The enabler will work with the Juno OpenStack release and will be qualified for prior releases (such as Icehouse) as well.

In the diagram below, OpenStack control and compute nodes are connected together after the generic OpenStack installation is finished. The compute nodes (DC servers of user choice) are connected to the leaf switches. DCNM and OpenStack control node needs to be connected using an IP network.

Figure 3: Sample Topology



For information about Open Source used in OpenStack for Cisco DFA 2.0, see the Open Source used in *OpenStack for Cisco DFA 2.0* document.



CHAPTER 2

Migration Overview

This chapter contains the following sections:

Prerequisites

To prepare for migration to the Cisco DFA solution, you must meet the following prerequisites.

- Deploy and configure Cisco Prime Data Center Network Manager 7.0
  - Perform tasks specified in the *Cisco Prime DCNM 7.0 OVA Installation Guide*
  - Perform tasks specified in the *Cisco Prime DCNM 7.0 Fundamentals Guide*
- FabricPath on Spine-Leaf Topology
  - Cisco Nexus 7000 Series spine switches with Cisco NX-OS 6.2(2)
  - Cisco Nexus 6000 Series border leaf switches with Cisco NX-OS 6.02.N2
  - Cisco Nexus 6000 Series leaf switches with Cisco NX-OS 6.02.N2 images

Note It is recommended to replace all the non-Cisco DFA device with Cisco Nexus 6000 Series switches with Cisco NX-OS version 7.0(0)N1(1) or later.

  - Cisco Nexus 1000V Series virtual switches at the virtual machine access layer

Limitations

Following are some of the limitations:

- Only legacy multicast is supported with Cisco Nexus 5500 Platform in DFA topology
- Enhanced fabric multicast is supported if DFA topology consists only of Cisco Nexus 6000 Series Switches
- The border leaf pair configuration is manual and no border leaf auto DCI configuration is supported until the topology has been fully migrated to DFA-capable nodes
- Any hosts behind border leaf are not supported
- Anchor leaf is always a border leaf and must be manually configured
- We can exit from migration mode only after all Cisco Nexus 5000 Series Switches are upgraded to Cisco Nexus 6000 Series Switches or Cisco Nexus 5600 Platform then write erase POAP via DCNM on border leaf can be done to support BL-DCI auto-configuration

Existing FabricPath Topology

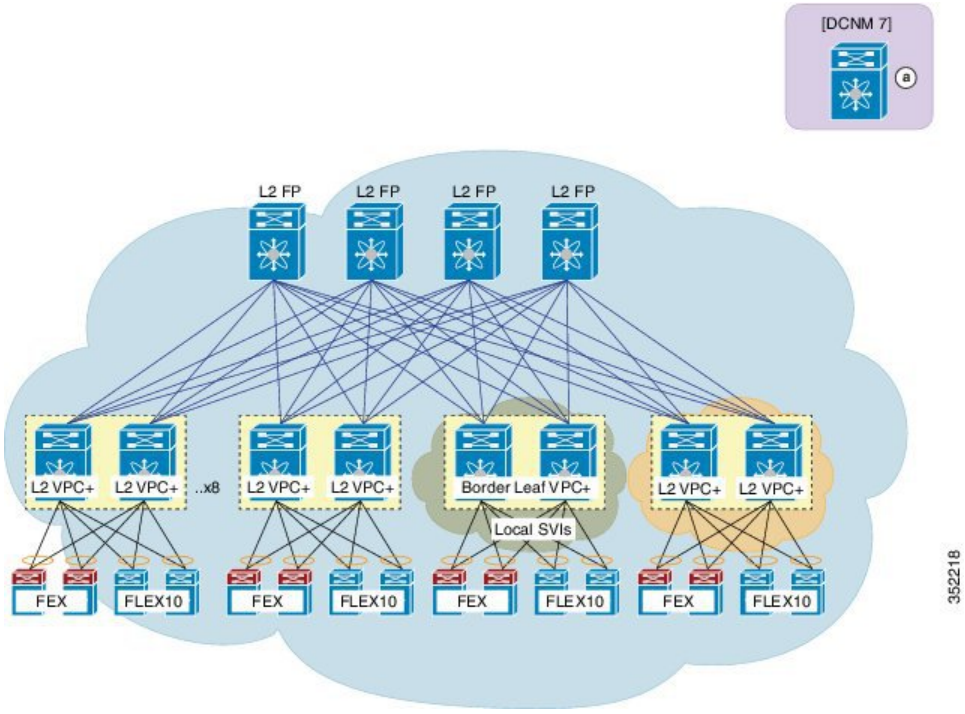
You must have this existing FabricPath topology before you can migrate to Cisco DFA.

- An access layer with FabricPath-enabled virtual port channel (vPC) peers (vPC+)
- Layer-3 aggregation layer-only connection to Spine layers
- At least one VPC+ pair, border leaf, for anchoring VLAN SVIs
- Switched Virtual Interfaces (SVIs) on only one set of vPC+ peers
- The Hot Standby Router Protocol (HSRP) running in local Layer-3 VLANs

Note Anchor leaf is supported only on border leaf.

The following figure shows the pre-migration fabric topology.

Figure 4: Pre-migration Fabric Topology



Cisco Dynamic Fabric Automation Topology

You can structure your Cisco DFA topology with two distinct fabrics:

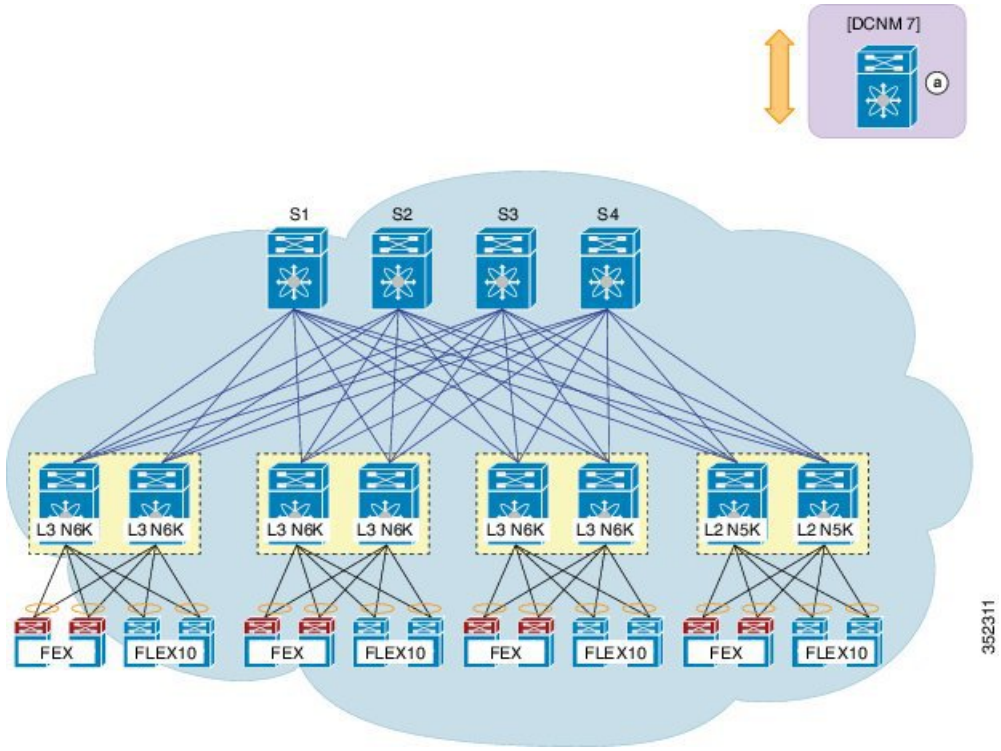
- Fabric with a mix of Cisco Nexus 5000 and 6000 Series leaf nodes
- Fabric with only Cisco Nexus 6000 Series leaf nodes

The Cisco DFA fabric with both Cisco Nexus 5000 and 6000 Series leaf nodes includes the following:

- Cisco Nexus 5000 Series remains as Layer-2
- Spine switches that can forward both 1q and 2q traffic, encapsulated in a FabricPath header
- VLAN/SVI differences are as follows:
  - On a Cisco Nexus 5000 Series Switches involved VLAN/SVI: segment IDs are not enabled on all leaf nodes for VLANs configured on Cisco Nexus 5000 Series leaf nodes. Border leaf runs HSRP/Virtual Router Redundancy Protocol as well as anycast gateway mode.
  - VLAN/SVIs with full DFA-leaf nodes only can be segment ID-enabled. The forwarding mode can be either proxy or anycast gateway mode.
  - Multicast will continue to run in the legacy multicast mode. Cisco DFA multicast should not be enabled.

The following figure shows the DFA fabric with Cisco Nexus 5000 and 6000 Series leaf nodes.

Figure 5: DFA Fabric with a Mix of Cisco Nexus 5000 and 6000 Series leaf nodes

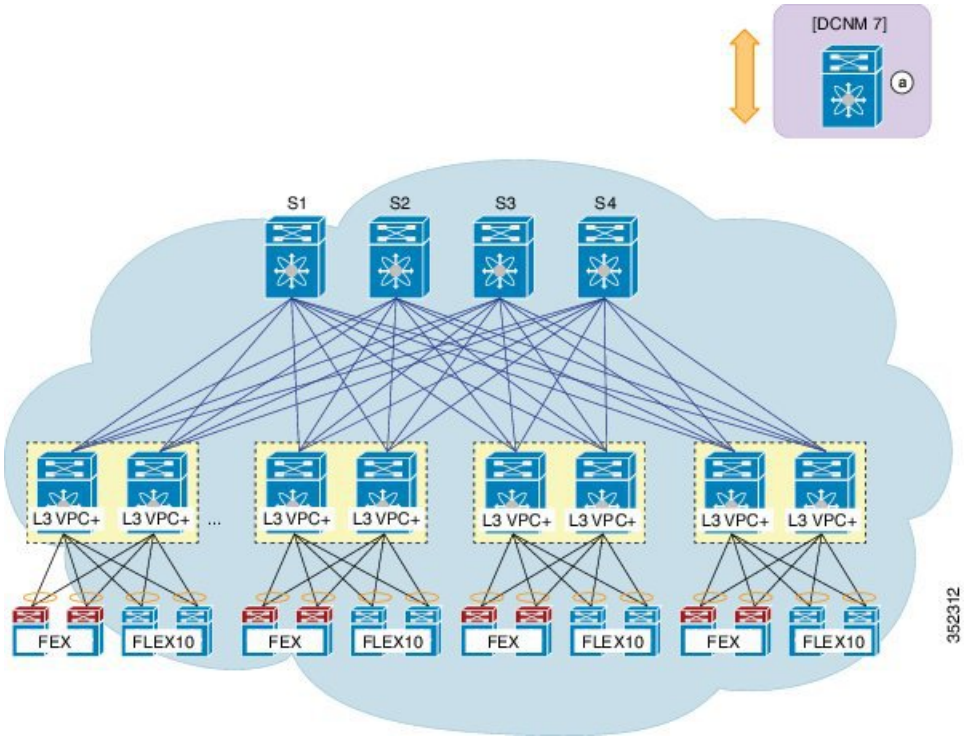


The DFA fabric with only Cisco Nexus 6000 Series leaf nodes includes the following:

- Cisco Nexus 6000 Series leaf nodes that run either anycast gateway mode or proxy gateway mode
- Spine switches that can forward both 1q and 2q traffic, encapsulated in a FabricPath header
- All VLANs can be segment ID enabled

The following figures show the DFA fabric with only Cisco Nexus 6000 Series leaf nodes.

Figure 6: DFA Fabric with Only Cisco Nexus 6000 Series leaf nodes

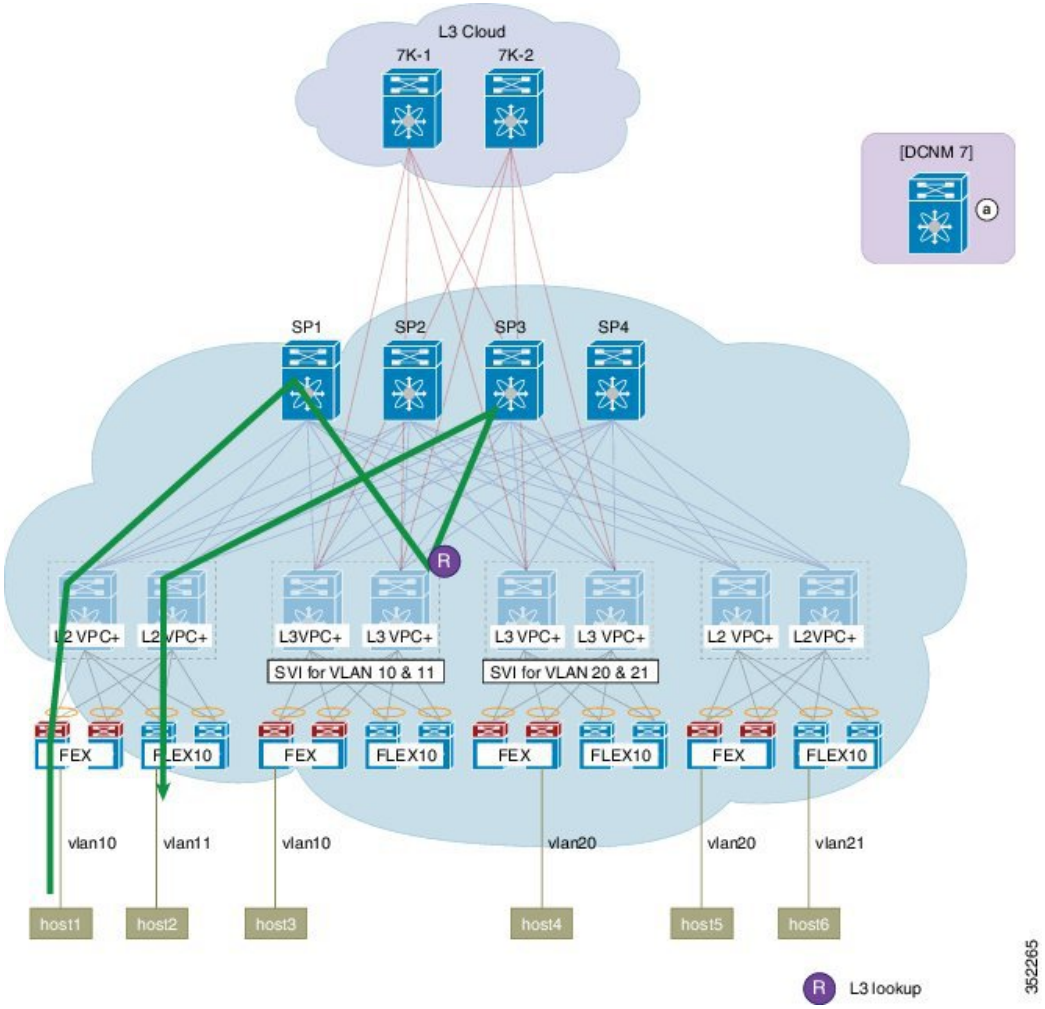


Traffic Flow Before and After Migration

As a result of changes to the topology and configuration of switches, the traffic flow is optimized after the migration. Traffic flow differences are shown in the following set of figures.

Prior to migration, the inter-VLAN traffic from Host 1 on VLAN10 goes through Layer-3 hops up through the spine to get to Host 2 on VLAN11.

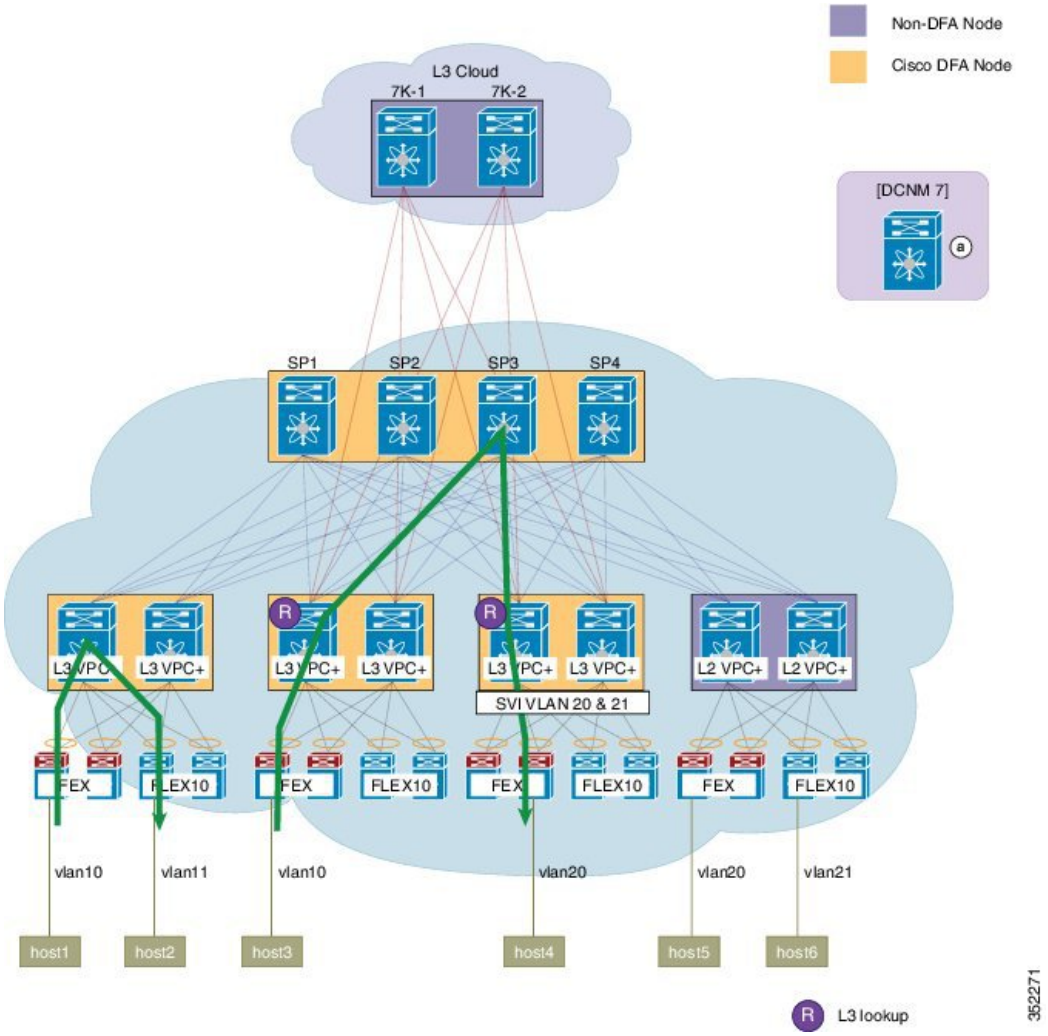
Figure 7: Pre-migration Inter-VLAN Unicast Traffic Flow to DFA node



During migration to the Cisco DFA fabric, the inter-VLAN traffic from Host 1 on VLAN 10 takes a single hop through a single leaf node where a Layer-3 lookup is performed and traffic is routed to Host 2 on VLAN

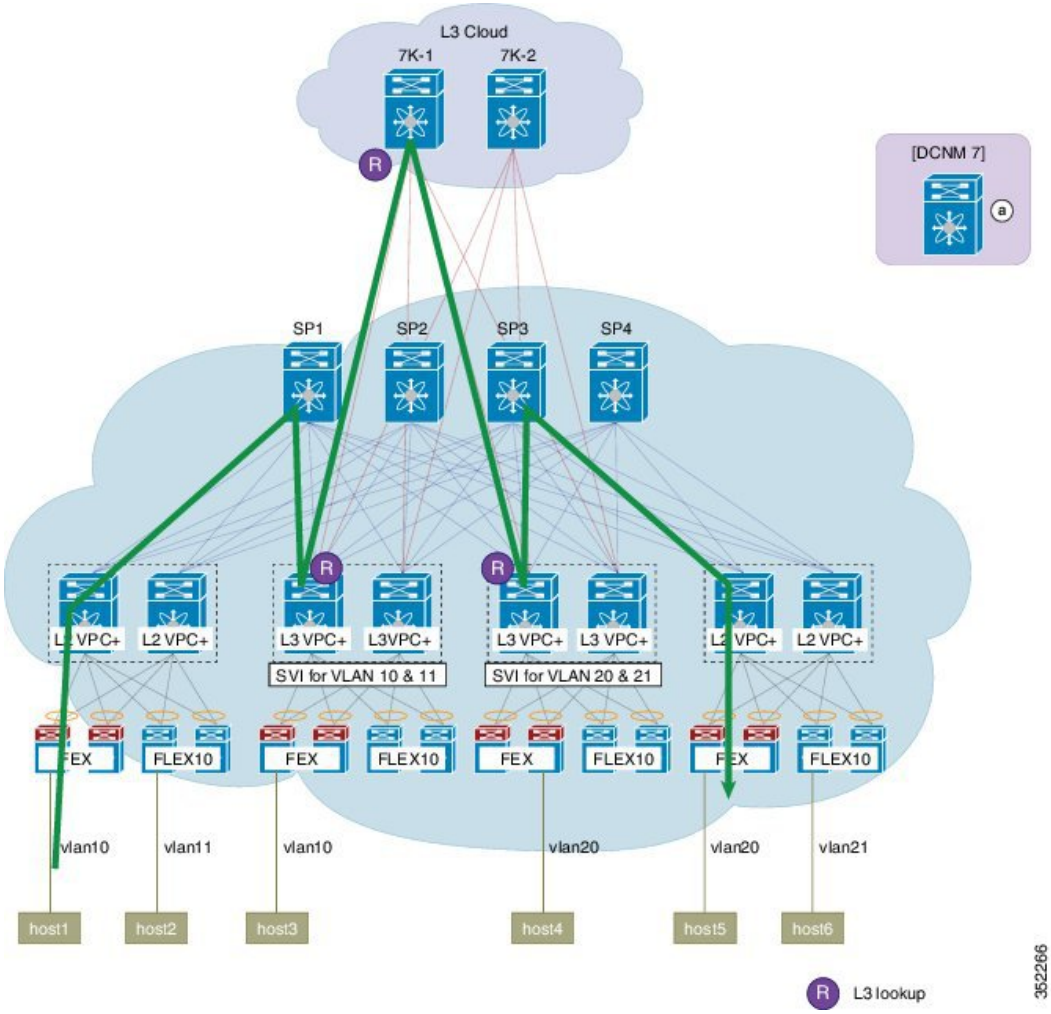
11. Border leaf nodes start to respond to the Address Resolution Protocol (ARP) with an anycast gateway MAC address.

Figure 8: Post-migration Inter-VLAN Unicast Traffic Flow to DFA node



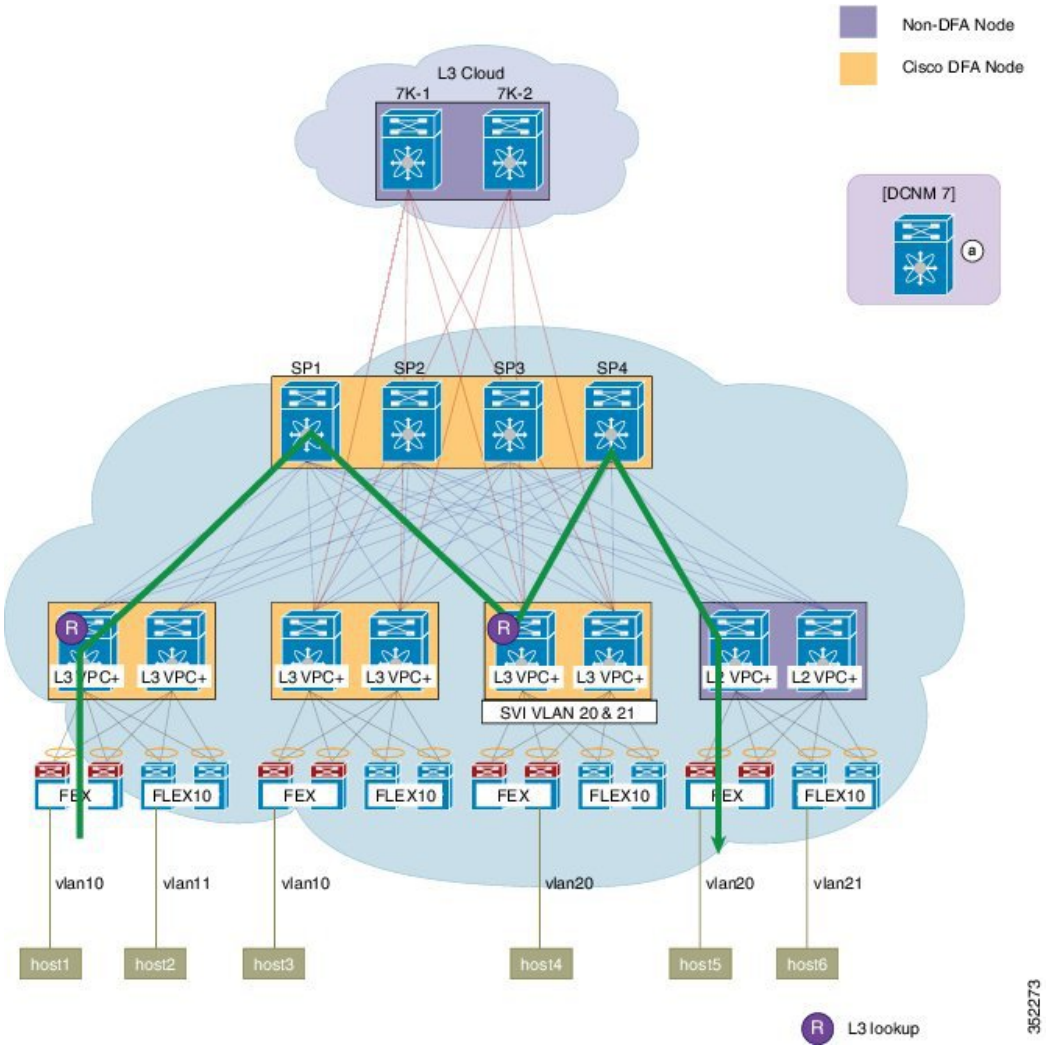
Prior to migration, the traffic going from Host1 on VLAN 10 to Host 5 on VLAN 20 takes multiple Layer-3 hops up to the Cisco Nexus 7000 Series Layer-3 and a series of Layer-3 lookups.

Figure 9: Pre-migration Inter-VLAN Unicast Traffic Flow to non-DFA node



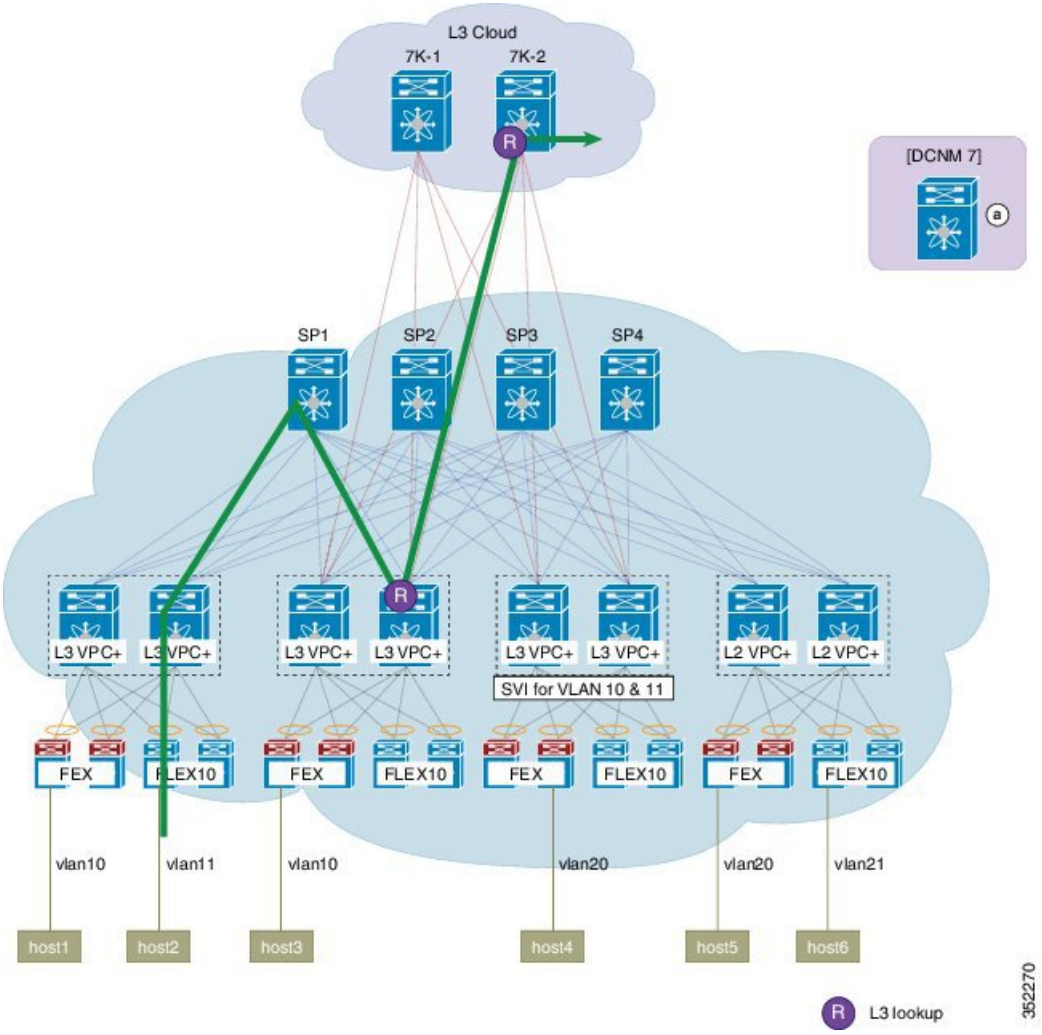
After migration, the unicast traffic that goes from Host 1 on VLAN 10 to Host 5 on VLAN 20 takes fewer Layer-3 lookups at the leaf-level, and direct forwarding occurs between border leaf pairs through the spine without going to the Cisco Nexus 7000 Series Switch.

Figure 10: Post-migration Inter-VLAN Unicast Traffic Flow to non-DFA node



The following figure shows that the north-south traffic remains unchanged after the migration and requires two Layer-3 lookups before reaching the Layer-3 cloud.

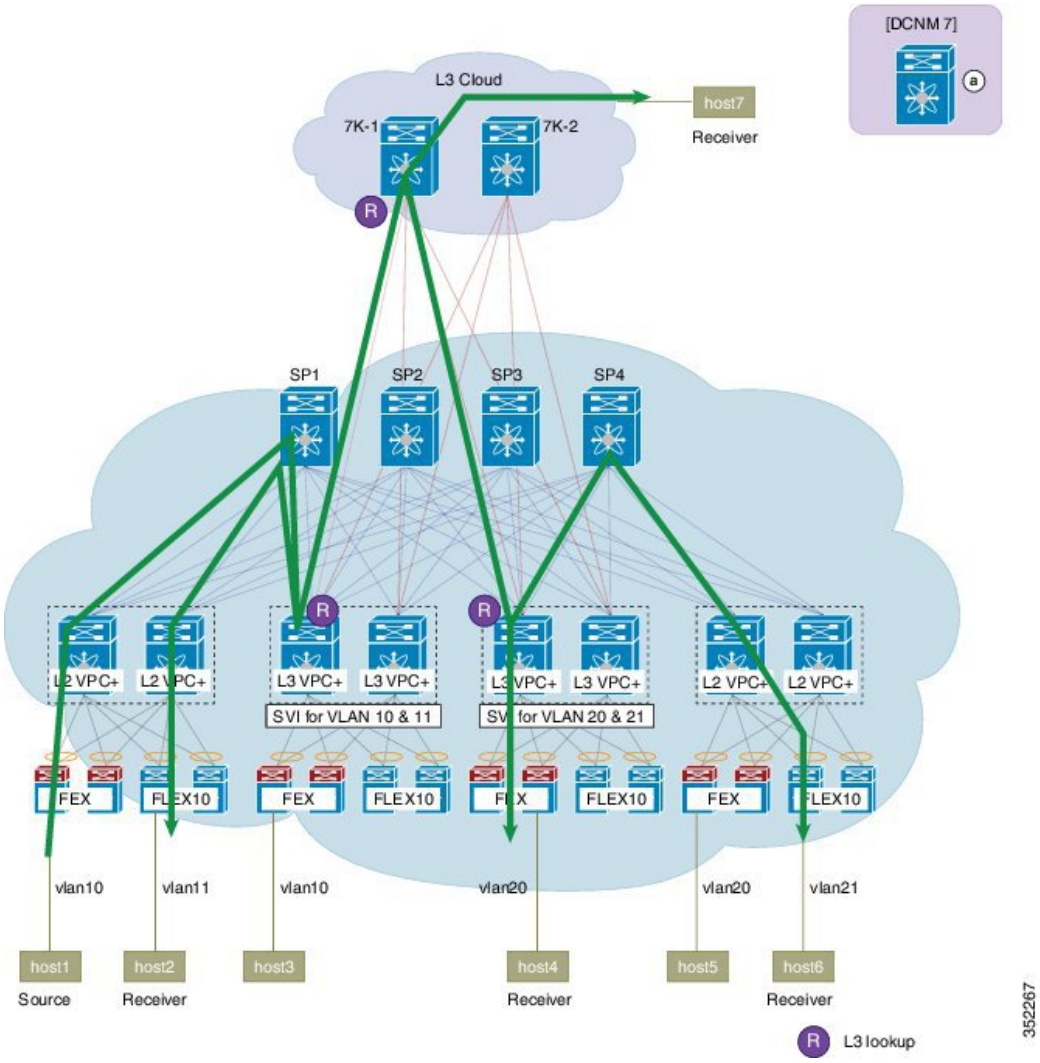
Figure 11: North South Traffic Flow



Protocol Independent Multicast (PIM)-Sparse Mode (SM) and multicast replication behavior is the same as a non-FabricPath topology. Layer-2 multicast forwarding follows a pruned FabricPath tree. The Internet Group

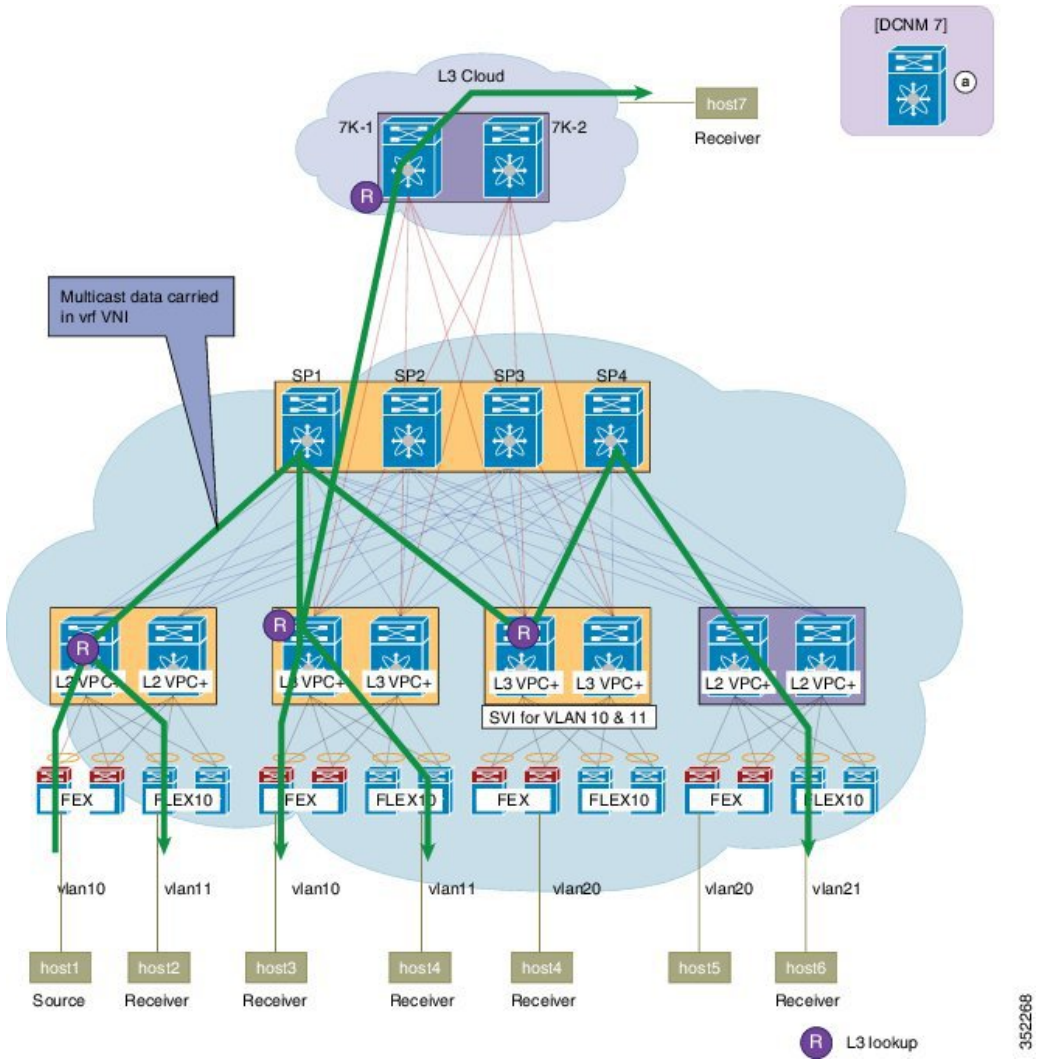
Management Protocol (IGMP) is propagated to all FabricPath nodes through Intermediate-system to intermediate-system (ISIS).

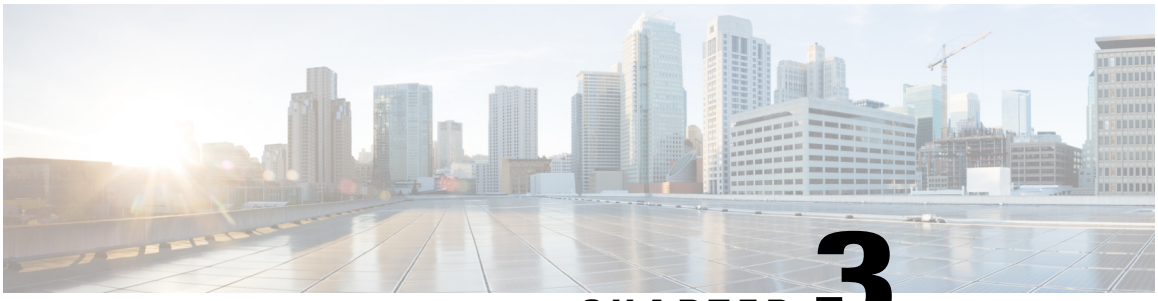
Figure 12: Pre-migration Multicast Traffic Flow



The following figures shows the traffic flow with DFA multicast and is only supported when all of the leaf nodes are Cisco Nexus 6000 Series nodes. Multicast traffic disruption will occur during the move to Cisco DFA multicast.

Figure 13: Post-Migration Multicast Traffic Flow





CHAPTER 3

Migration Steps

Note It is recommended not to use VLAN 1 as control segment.

This chapter contains the following sections:

1) Upgrading and Configuring the Spine Switch Software

You must first upgrade all the spine switch software.

Before You Begin

The following prerequisites must be met before you upgrade the Cisco Nexus 6000 Series spine switch software:

- The Cisco Nexus 6000 Series switch must be running Cisco NX-OS Release 6.0(x) or later

Note If you have anything other than a Cisco Nexus 6000 Series switch, you must physically replace the switch with Release 7.0(0)N1(1); the configuration remains the same as the previous image.

Step 1 On the Cisco Nexus 6000 Series spine switches, perform a nondisruptive in-service software upgrade (ISSU) upgrade to Cisco NX-OS Release 7.0(0)N1(1).
See the Cisco Nexus 6000 Series NX-OS Software Upgrade and Downgrade, Release 7.0 for instructions on performing the ISSU upgrade.

No impact to traffic should occur as a result of the ISSU upgrade.

Step 2 Add Cisco Dynamic Fabric Automation-specific configuration on the spine.
For more information, see Migration Configuration, on page 29.

2) Upgrading the Border Leaf Software

You can perform a disruptive in-service software upgrade (ISSU) for the first border leaf pair.

Before You Begin

Before you perform an ISSU upgrade, move any existing switch virtual interface (SVI) configurations from other switches in the spine-leaf topology to the border-leaf pair nodes.

Note You must perform configuration on the border-leaf nodes manually, not through power-on auto-provisioning (POAP).

Step 1 Upgrade the first border leaf node from Cisco NX-OS Release 6.0(2)N2 to Cisco NX-OS Release 7.0(0)N1(1) using an ISSU disruptive upgrade procedure. See the Cisco Nexus 6000 Series NX-OS Software Upgrade and Downgrade, Release 7.0 for information on performing an ISSU upgrade.

Step 2 Verify that the first border leaf comes up fully and becomes operational again.

Step 3 Verify that the traffic streams are already running intra-VLAN, inter-VLAN, across pods, and that north-bound traffic remains unaffected.

Step 4 Repeat Steps 1 to 3 for the second border leaf node in the pair and also for additional border leaf pairs.
Although the disruptive upgrade has some effect on traffic, traffic flow is not changed

3) Configuring the Border Leaf Pair

Note For specific configuration commands and examples, see Migration Configuration, on page 29.

Step 1 On the first border leaf switch, do the following:

a) Configure the Hot Standby Router Protocol (HSRP) per VLAN with the anycast gateway MAC address and an unused IP address.
b) (Optional) route reflector (RR) normally runs on the spine but they can be configured to run on the leaf or border leaf.
c) Configure the anycast gateway MAC address.
d) Add a vrf-tenant-profile and configure the virtual network identifier (VNI) under the virtual routing and forwarding (VRF) instance.
e) Create switch virtual interfaces (SVIs), if they are not present.
f) Enable anycast-gateway on the SVIs.
g) Enable traditional forwarding on the switch virtual interfaces (SVIs).
h) Configure the BGP so that host routes are advertised to the BGP route reflector.

Step 2 Repeat Step 1 for the second border leaf switch in the pair and also for additional border leaf pairs.

4) Upgrading the FabricPath Leaf Pair

You can perform an in-service software upgrade (ISSU) for the FabricPath leaf pair.

Step 1 Upgrade the leaf node from Cisco NX-OS Release 6.0(2)N2 to Cisco NX-OS Release 7.0(0)N1(1) using an ISSU upgrade procedure. See the Cisco Nexus 6000 Series NX-OS Software Upgrade and Downgrade, Release 7.0. for information on performing an ISSU upgrade.

Step 2 Verify that the first leaf comes up fully and becomes operational again.

Step 3 Make sure traffic streams are already running intra-VLAN, inter-VLAN, across pods, and that north-bound traffic remains unaffected.

Step 4 Repeat Steps 1 to 3 for the second border leaf node in the pair.

There is no change in the traffic flow.

5) Adding DFA Configuration to FabricPath Leaf Pair

You can configure the FabricPath leaf nodes in the network.

Note For specific configuration commands and examples, see Migration Configuration, on page 29.

Before You Begin

Before you configure the FabricPath leaf, you should upgrade the software.

Step 1 On the first switch in the pair, do the following:

a) Install a Layer-3 license.
b) Enable Cisco DFA.
c) Configure the iBGP route reflector client.
d) Add the segment ID and virtual routing and forwarding (VRF) instance.
e) Add a vrf-tenant-profile and configure the virtual network identifier (VNI) under the virtual routing and forwarding (VRF) instance.
f) Create switch virtual interfaces (SVIs) for all VLANs.
g) Enable anycast-gateway mode for all VLANs.

Step 2 Repeat Step 1 for the second border leaf switch in the pair.

If you are migrating a fabric that includes both Cisco Nexus 5000 Series and Cisco Nexus 6000 Series Switches, do the following:

- Migration is completed if you have upgraded all Cisco Nexus 6000 Series software and enabled Cisco DFA forwarding.
- HSRP/VRRP remains if there are Cisco Nexus 5000 Series leaf nodes in the network.
- In Cisco Nexus 5000 Series-involved VLANs and SVIs, the VLANs are global, non-segment-ID-enabled, and the forwarding mode is anycast gateway mode.
- In upgraded and only Cisco Nexus 6000 Series-involved VLANs and SVIs, the VLANs and SVIs can be segment ID enabled, and the forwarding mode can be either proxy or anycast gateway mode.
- Multicast continues to run in the legacy multicast mode. Cisco DFA multicast should not be enabled.

What to Do Next

If you are migrating a fabric that includes both Cisco Nexus 5000 Series and Cisco Nexus 6000 Series Switches, the migration is completed if you have upgraded all Cisco Nexus 6000 Series software and enabled Cisco DFA forwarding.

6) Upgrading and Configuring All Remaining Leaf Switches

You should perform this procedure on all of the remaining leaf switches in the network.

Note For specific configuration commands and examples, see Migration Configuration, on page 29.

Step 1 Upgrade the software on all of the remaining leaf switches.

Step 2 Add Cisco DFA-related configuration on all of the remaining leaf switches.

Step 3 Enable anycast-gateway mode on leaf switches for all VLANs.

7) Removing the HSRP Configurations on Border Leaf Pairs

Note This step is performed only when there are no Cisco Nexus 5000 Series leaf nodes in the Cisco Dynamic Fabric (DFA) topology.

During the migration, some hosts learn the anycast gateway MAC address as its MAC address for the default gateway. Some hosts learn the HSRP Virtual Mac Address (VMAC) as the MAC address for the default gateway. We recommend that you wait a couple of hours to make sure that the HSRP VMAC address is aged out on all hosts.

Note For specific configuration commands and examples, see Migration Configuration, on page 29.

Before You Begin

You should have completed migration on all leaf switches.

Step 1 Remove the HSRP configuration on each border leaf switch.

Step 2 Change the SVI IP address to the Virtual IP (VIP) address.

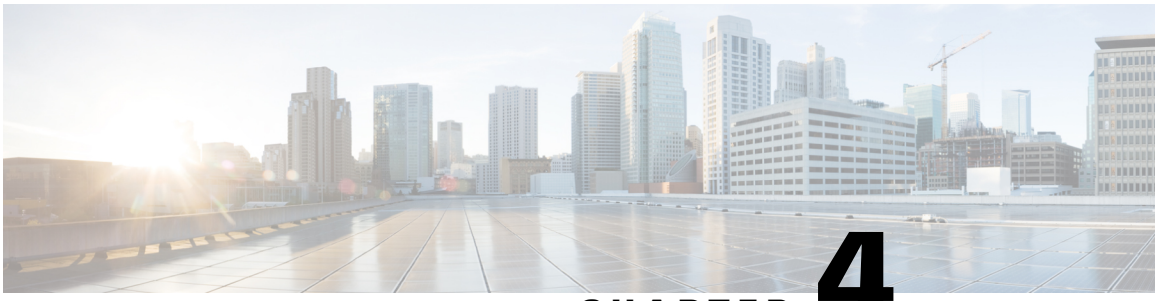
After you remove the HSRP configurations, migration is complete.

Note

- If there are two or more HSRP groups, then there will be more than one HSRP VIP configured on the attached host. But only one VIP can be used as the DFA anycast gateway IP. This requires changes on the attached hosts to use the selected VIP as default gateway.
- Only one v4 or v6 HSRP dummy group is needed per SVI.

- You can move to Cisco DFA multicast, if preferred. There is no vPC on the border leaf connecting to external multicast routers.

- SVIs can be switched to proxy forwarding mode, if preferred.
- New VLANs can be segment ID enabled.
- In an all Cisco Nexus 6000 Series topology, you can move to Cisco DFA multicast, if preferred.



CHAPTER 4

Migration Configuration

This chapter contains the following sections:

Configuring the BGP Route Reflector on a Spine

You can add the Cisco Dynamic Fabric Automation (DFA)-specific Border Gateway Protocol (BGP) configuration on the spine and identify the BGP route reflector.

Note the following requirements on spine switches:

- If you use a Cisco Nexus 6000 or 7000 Series switch as a spine switch, do the following. FabricPath non-tranit mode is supported on Cisco Nexus 7000 Series Switches.
  - For a transit mode switch, you must enter the **fabricpath mode transit** command.

Note For the **fabricpath mode transit** command to take effect, the system must be reloaded.

Note Spines in the Cisco DFA fabric might or might not be BGP route-reflector nodes. If you must configure the spine switch as a BGP route-reflector, use the configuration in the following procedure.

Before You Begin

You must upgrade the spine switch software.

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch #**configure terminal**	Enters global configuration mode.
Step 2	switch (config) # **feature bgp**	Enables the Border Gateway Protocol (BGP). You must enable the BGP feature before you can configure BGP.
Step 3	switch (config) # **router bgp** *bgp-as*	Configures a BGP process for an interface. The *as-number* is the number of an autonomous system that identifies the router to other BGP routers and tags that the routing information passed along. The AS number can be a 16-bit integer or a 32-bit integer in the form of a higher 16-bit decimal number and a lower 16-bit decimal number in xx.xx format.
Step 4	switch (config-router) # **address-family ipv4 unicast**	Enters address family mode and configures submode commands for the BGP.
Step 5	switch (config-router) # **maximum-paths ibgp** [**maximum parallel paths**]	Controls the maximum number of parallel routes that the BGP can support.
Step 6	switch (config-router-af) # **additional-paths send**	Sends additional paths to and from the BGP peers.
Step 7	switch (config-router-af) # **additional-paths selection route-map** *All-paths*	Specifies the route map for the additional paths selection.
Step 8	switch (config-router) # **address-family ipv6 unicast**	Enter address family mode and configures submode commands for the BGP.
Step 9	switch (config-router) # **maximum-paths** [**ibgp**]	Controls the maximum number of parallel routes that the BGP can support.
Step 10	switch (config-router-af) # **additional-paths send**	Sends additional paths to and from the BGP peers.
Step 11	switch (config-router-af) # **additional-paths selection route-map**	Specifies the route map for the additional paths selection.
Step 12	switch (config-router) # **address-family vpnv4 unicast**	Enters address family mode and configures submode commands for the BGP.
Step 13	switch (config-router-af) # **additional-paths send**	Sends additional paths to and from the BGP peers.
Step 14	switch (config-router-af) # **additional-paths receive**	Receives additional paths to and from the BGP peers.
Step 15	switch (config-router-af) # **additional-paths selection route-map**	Specifies the route map for the additional paths selection.
Step 16	switch (config-router) # **address-family vpnv6 unicast**	Enters address family mode and configures submode commands for the BGP.
Step 17	switch (config-router-af) # **additional-paths send**	Sends additional paths to and from the BGP peers.
Step 18	switch (config-router-af) # **additional-paths receive**	Receives additional paths to and from the BGP peers
Step 19	switch (config-router-af) # **additional-paths selection route-map**	Specifies the route map for the additional paths selection.

	Command or Action	Purpose
Step 20	switch (config-router) # **neighbor** {*bgp-client-subnet/mask*} [**remote-as** {*as-num* [,*as-num*]}	Configures a BGP neighbor (router, vrf) and enters neighbor configuration mode.
Step 21	switch (config-router-neighbor) # **address-family ipv4 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode to configure submode commands for the BGP.
Step 22	switch (config-router-neighbor-af) # **send-community**	Sends a BGP community attribute to a peer.
Step 23	switch (config-router-neighbor-af) # **send-community** [**extended**]	Sends extended BGP community attribute to a peer.
Step 24	switch (config-router-neighbor-af) #**route-reflector-client**	Configures the router as a BGP route reflector and configures the specified neighbor as its client.
Step 25	switch (config-router-neighbor) # **address-family ipv6 unicast**	Enters address family mode configure submode commands for the BGP.
Step 26	switch (config-router-neighbor-af) # **send-community** [**extended**]	Sends a BGP community attribute to a peer.
Step 27	switch (config-router-neighbor-af) #**route-reflector-client**	Configures the router as a BGP route reflector and configures the specified neighbor as its client.
Step 28	switch (config-router-neighbor) # **address-family vpnv4 unicast**	Enters address family mode configure submode commands for the BGP.
Step 29	switch (config-router-neighbor-af) # **send-community** [**extended**]	Sends a BGP community attribute to a peer.
Step 30	switch (config-router-neighbor-af) # **route-reflector-client**	Configures the router as a BGP route reflector and configures the specified neighbor as its client.
Step 31	switch (config-router-neighbor-af) # **capability additional-paths receive**	Configures BGP to advertise the capability of sending and receiving additional paths to and from the BGP peers.
Step 32	switch (config-router-neighbor) #**address-family-vpnv6 unicast**	Enters address family mode configure submode commands for the BGP.
Step 33	switch (config-router-neighbor-af) # **send-community** [**extended**]	Sends a BGP community attribute to a peer.
Step 34	switch (config-router-neighbor-af) # **route-reflector-client**	Configures the router as a BGP route reflector and configures the specified neighbor as its client.

This example shows how to configure the BGP route reflector on the spine switch.

```
switch # configure terminal
switch (config) # feature bgp
switch (config) # router bgp 100
switch (config-router) # router-id 1.1.1.4
switch (config-router) # address-family ipv4 unicast
switch (config-router-af) # redistribute hmm route-map AM <---AM is the route-map name that
```

```
 permits all IPv4 routes excluding VLAN-x backbone prefix
switch (config-router-af) # maximum-paths ibgp 2
switch (config-router-af) # additional-paths send
switch (config-router-af) # additional-paths selection route-map ALL-PATHS
switch (config-router) # address-family ipv6 unicast
switch (config-router-af) # redistribute hmm route-map host-v6 <---host-v6 is the route-map
 name that
permits all IPv6 routes
switch (config-router-af) # maximum-paths ibgp 2
switch (config-router-af) # additional-paths send
switch (config-router-af) # additional-paths selection route-map ALL-PATHS
switch (config-router) # address-family vpnv4 unicast
switch (config-router-af) # additional-paths send
switch (config-router-af) # additional-paths receive
switch (config-router-af) # additional-paths selection route-map ALL-PATHS
switch (config-router) # address-family vpnv6 unicast
switch (config-router-af) # additional-paths send
switch (config-router-af) # additional-paths receive
switch (config-router-af) # additional-paths selection route-map ALL-PATHS
switch (config-router) # neighbor 1.1.1.0/24 remote-as 100 <---Route-Reflector Spine
IP=1.1.1.1
switch (config-router-neighbor) # address-family ipv4 unicast
switch (config-router-neighbor-af) # send-community
switch (config-router-neighbor-af) # send-community extended
switch (config-router-neighbor-af) # route-reflector-client
switch (config-router-neighbor) # address-family ipv6 unicast
switch (config-router-neighbor-af) # send-community extended
switch (config-router-neighbor-af) # route-reflector-client
switch (config-router-neighbor) # address-family vpnv4 unicast
switch (config-router-neighbor-af) # send-community extended
switch (config-router-neighbor-af) # route-reflector-client
switch (config-router-neighbor) # address-family vpnv6 unicast
switch (config-router-neighbor-af) # send-community extended
switch (config-router-neighbor-af) # route-reflector-client
```

Updating SVI Configuration on Border Leaf Nodes

You can enable anycast forwarding mode on switched virtual interfaces (SVIs) without a VN-segment on border leaf devices and nondefault VRF VLANs and you can enable the Hot Standby Routing Protocol (HSRP) virtual IP addresses on the border leaf.

Before You Begin

You must upgrade the border leaf software.

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch # **configure terminal**	Enters global configuration mode.
Step 2	switch (config) # **feature hsrp**	Enters HSRP configuration mode and enables HSRP.
Step 3	switch (config) # **interface vlan** *vlan-id*	Creates a VLAN interface and enters interface configuration mode. The *vlan-id* range is from 1 to 4094.
Step 4	switch (config-if) # **no shutdown**	Disables the shutdown function on an instance of the BGP.
Step 5	switch (config-if) # **no ip redirects**	Disables IP redirects.

	Command or Action	Purpose
Step 6	switch (config-if) # **ip address** *ip-address-mask*	Specifies a primary IP address for an interface.
Step 7	switch (config-if) # **ipv6 address** {*addr* | [**eui64**] [**route-preference** *preference*] [**secondary**] [**tag** *tag-id*] | **use-link-local-only** }	Configures an IPv6 address on an interface. The *addr* format is A:B::C:D/length. The length range is 1 to 128. The **eui64** command configures the Extended Unique Identifier (EUI64) for the low-order 64 bits of the address. The **route-preference** command sets the route preference for local or direct routes. The*preference* range is from 0 to 255. The **secondary** command creates a secondary IPv6 address. The tag**tag** command configures a route tag value for local or direct routes. The **use-link-local-only** command specifies IPv6 on the interface using only a single link-local.
Step 8	switch (config-if) # **ip router ospf area** *instance-tag* **area** *area-id* [**secondaries none**]	Specifies the Open Shortest Path First (OSPF) instance and area for an interface. The *instance-tag* is locally assigned and can be any word or positive integer; can be a maximum of 20 alphanumeric characters.
Step 9	switch (config-if) # **fabric forwarding anycast-gateway-mac** *mac-address*	Specifics the MAC address of the server-facing ports across all leaf nodes. The anycast gateway MAC address is used per interface; it is replicated across all the switch virtual interfaces (SVI) that are supporting proxy gateway or anycast gateway mode.
Step 10	switch (config-if) # **hrsp version 2**	Configures the Hot Standby Redundancy Protocol (HSRP) version 2.
Step 11	switch (config-if-hsrp) # **hsrp** *group-number* [**ip4** | **ipv6**]	Enters HSRP configuration mode and creates the number of HSRP groups that can be configured on a Gigabit Ethernet port, including the main interfaces and subinterfaces. The *group-number* range is from 1 to 255. The default value is 0.
Step 12	switch (config-if-hsrp) # **preempt** [**delay** {**minimum** *min-delay* | **reload** *rel-delay* | **sync** | *sync-delay*}]	Configures a preemption delay. • If you opt to specify a **delay minimum**, this specifies the minimum number of seconds that preemption is delayed to allow routing tables to be updated before a router becomes active. • If you opt to specify a **delay reload** , this specifies the time delay after the router has reloaded. This period applies only to the first interface-up event after the router had reloaded. The default value is 0. • If you opt to specify a **delay sync**, this specifies the maximum number of seconds to allow IP redundancy clients to prevent preemption. When this period expires, preemption occurs regardless of the state of the IP redundancy clients. The default value is 0.
Step 13	switch (config-if-hsrp) # **priority** *level* [**forwarding-threshold lower** *lower-value* **upper** *upper-value*]	Sets the priority level within an HSRP group. The *level* range of values is from 1 to 255. If this router is the owner of the IP addresses, then the value is automatically set to 255. The default is 100. • If you specify a **forwarding-threshold**, you set the threshold used by a vPC to determine when to fail over to the vPC trunk. The *lower-value* range

	Command or Action	Purpose
		is from 1 to 255. The default is 1. The *upper-value* range is from 1 to 255. The default is 255.
Step 14	switch (config-if-hsrp) # **ip** [**autoconfig** | *ip-address* [**secondary**]]	Assigns a virtual address to an HSRP group.
Step 15	switch (config-if-) # **hsrp** *group-number* [**ip4** | **ipv6**]	Enters HSRP configuration mode and creates the number of HSRP groups that can be configured on a Gigabit Ethernet port, including the main interfaces and subinterfaces. The *group-number* range is from 1 to 255. The default value is 0.
Step 16	switch (config-if-hsrp) # **mac-address** *mac-address*	Configures a static MAC address for a Layer 3 interface.
Step 17	switch (config-if-hsrp) # **preempt** [**delay** {**minimum** *min-delay* | **reload** *rel-delay* | **sync** | *sync-delay*}]	Configures a preemption delay. • If you opt to specify a **delay minimum**, this specifies the minimum number of seconds that preemption is delayed to allow routing tables to be updated before a router becomes active. • If you opt to specify a **delay reload** , this specifies the time delay after the router has reloaded. This period applies only to the first interface-up event after the router had reloaded. The default value is 0. • If you opt to specify a **delay sync**, this specifies the maximum number of seconds to allow IP redundancy clients to prevent preemption. When this period expires, preemption occurs regardless of the state of the IP redundancy clients. The default value is 0.
Step 18	switch (config-if-hsrp) # **priority** *level* [**forwarding-threshold lower** *lower-value* **upper** *upper-value*]	Sets the priority level within an HSRP group. The *level* range of values is from 1 to 255. If this router is the owner of the IP addresses, then the value is automatically set to 255. The default is 100. • If you specify a **forwarding-threshold**, you set the threshold used by a vPC to determine when to fail over to the vPC trunk. The *lower-value* range is from 1 to 255. The default is 1. The *upper-value* range is from 1 to 255. The default is 255.
Step 19	switch (config-if-hsrp) # **ip** [**autoconfig** | *ip-address* [**secondary**]]	Assigns a virtual address to an HSRP group. If you use the **autoconfig** command, it generates a link-local address from the link-local prefix and a modified EUI-64 format Interface Identifier, where the EUI-64 Interface Identifier is created from the relevant HSRP virtual MAC address. The *ip-address* is the Virtual IP address for the virtual router (HSRP group). The IP address must be in the same subnet as the interface IP address. You must configure the virtual IP address for at least one of the routers in the HSRP group. Other routers in the group will pick up this address. The IP address can be an IPv4 address. The **secondary** command indicates that the IP4 address is a secondary HSRP virtual address.

	Command or Action	Purpose
Step 20	switch (config-if) # **hsrp** *group-number* [**ip4** | **ipv6**]	Enters HSRP configuration mode and creates the number of HSRP groups that can be configured on a Gigabit Ethernet port, including the main interfaces and subinterfaces. The *group-number* range is from 1 to 255. The default value is 0.
Step 21	switch (config-if-hsrp) # **mac-address** *mac-address*	Configures a static MAC address for a Layer 3 interface.
Step 22	switch (config-if-hsrp) # **preempt** [**delay** {**minimum** *min-delay* | **reload** *rel-delay* | **sync**|*sync-delay*}]	Configures a preemption delay. • If you opt to specify a **delay minimum**, this specifies the minimum number of seconds that preemption is delayed to allow routing tables to be updated before a router becomes active. • If you opt to specify a **delay reload** , this specifies the time delay after the router has reloaded. This period applies only to the first interface-up event after the router had reloaded. The default value is 0. • If you opt to specify a **delay sync**, this specifies the maximum number of seconds to allow IP redundancy clients to prevent preemption. When this period expires, preemption occurs regardless of the state of the IP redundancy clients. The default value is 0.
Step 23	switch (config-if-hsrp) # **priority** *level* [**forwarding-threshold lower** *lower-value* **upper** *upper-value*]	Sets the priority level within an HSRP group. The *level* range of values is from 1 to 255. If this router is the owner of the IP addresses, then the value is automatically set to 255. The default is 100. • If you specify a **forwarding-threshold**, you set the threshold used by a vPC to determine when to fail over to the vPC trunk. The *lower-value* range is from 1 to 255. The default is 1. The *upper-value* range is from 1 to 255. The default is 255.
Step 24	switch (config-if-hsrp) # **ip** [**autoconfig** | *ip-address* [**secondary**]]	Assigns a virtual address to an HSRP group. If you use the **autoconfig** command, it generates a link-local address from the link-local prefix and a modified EUI-64 format Interface Identifier, where the EUI-64 Interface Identifier is created from the relevant HSRP virtual MAC address. The *ip-address* is the Virtual IP address for the virtual router (HSRP group). The IP address must be in the same subnet as the interface IP address. You must configure the virtual IP address for at least one of the routers in the HSRP group. Other routers in the group will pick up this address. The IP address can be an IPv4 address. The **secondary** command indicates that the IP4 address is a secondary HSRP virtual address.

This example shows how to configure the SVI interfaces for default/nondefault VRF instances, as well as associated HSRP and dummy HSRP groups with anycast gateway MAC addresses.

```
switch (config) # feature hsrp
switch (config) # interface vlan20
switch (config-if) # no shutdown
switch (config-if) #  no ip redirects
switch (config-if) # ip address 20.1.1.104/24
switch (config-if) # ipv6 address 20:1::104/64
```

```
switch (config-if) # ip router ospf 1 area 0.0.0.
switch (config-if) # fabric forwarding mode anycast gateway <---must be added to configure
 vlan-20 in Cisco DFA mode
switch (config-if) # hsrp version 2
switch (config-if) # hrsp 20 ip4
switch (config-if-hsrp) # preempt
switch (config-if-hsrp) # priority 110
switch (config-if-hsrp) # ip 20.1.1.100
switch (config-if) # hsrp 20 ipv6
switch (config-if-hsrp) # preempt
switch (config-if-hsrp) # priority 110
switch (config-if-hsrp) # ip 20:1::100
switch (config-if) # hsrp 50 ipv4 <---dummy HSRP group (ipv4 or ipv6)
switch (config-if-hsrp) # mac-address DEAD.0000.DEAF <---anycast gateway MAC
switch (config-if-hsrp) # preempt
switch (config-if-hsrp) # priority 110
switch (config-if-hsrp) # ip 20.1.1.200 <---functionally unused IP
```

Configuring Border Leafs for DFA

You can configure an upgraded border leaf.

Before You Begin

You must upgrade the border leaf software.

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch # **configure terminal**	Enters global configuration mode.
Step 2	switch (config) # **install feature-set fabricpath**	Installs the FabricPath feature set on the switch.
Step 3	switch (config) # **install feature-set fabric**	Installs the fabric feature on the switch.
Step 4	switch (config) # **feature-set fabricpath**	Enables a FabricPath feature set.
Step 5	switch (config) # **feature-set fabric**	Enables the fabric feature on the switch.
Step 6	switch (config) # **feature fabric forwarding**	Enables fabric network services on a device and enables the Host Mobility Manager and release-specific HMM configuration commands..
Step 7	switch (config) # **feature bgp**	Enables the Border Gateway Protocol (BGP). You must enable the BGP feature before you can configure BGP.
Step 8	switch (config) # **feature isis**	Enables the intermediate-system-to-intermediate-system (ISIS) for FabricPath core.
Step 9	switch (config) # **feature vn-segment-vlan-based**	Enables the VLAN-based virtual network (VN) segment feature on a switch. You can use this feature only if the FabricPath feature set is enabled on the switch..
Step 10	switch (config) # **system fabric dynamic-vlans** *vlan-range*	This allocation is mandatory to include for tenant VRFs core VLAN range as well as the entire server facing VLANs range. The range has to be continuous. Control segment VLAN is not a part of this dynamic range.

	Command or Action	Purpose
Step 11	switch (config) # **system fabric core-vlans** *vlan-id -or-range*	Defines a range of VLANs out of the dynamic range, to be used for tenant core SVI. The VLAN range is reserved to be in use for Tenant VRF core VLANs.
Step 12	switch (config) # **fabric forwarding identifier** *id*	Specifies a unique fabric ID. The *id* range is from 1 to 65535.
Step 13	switch (config) # **fabric forwarding anycast-gateway-mac** *mac-address*	Specifies the MAC address of the server-facing ports across all leaf nodes. The anycast gateway MAC address is used per interface, so it is replicated across all the switch virtual interfaces (SVIs) that are supporting proxy gateway or anycast gateway.
Step 14	switch (config) # **fabric forwarding switch-role** [**border**] {**leaf** | **spine**}	Defines the switch role. The Leaf adds tenant (vrf) functionality; the border leaf adds the ability to connect with routers.
Step 15	switch (config) # **fabricpath domain default**	Enters global FabricPath Layer 2 ISIS configuration mode.
Step 16	switch (config) # **vlan** *fabric-control-vlan-id*	Specifies the VLAN IDs of the allowed FabricPath VLANs in the anycast bundle. You can specify a*vland-id* in a range from 1 to 4094.
Step 17	switch (config-vlan) # **mode fabricpath**	Enables the VLAN as a FabricPath VLAN and enters FabricPath mode.
Step 18	switch (config) # **interface vlan** *vlan-id*	Creates the corresponding Layer 3 VLAN interface and enters interface configuration mode. The *vlan-id* range is from 2 to 4094.
Step 19	switch (config-if) # **no shutdown**	Disables the shutdown function on an instance of the BGP.
Step 20	switch (config-if) # **ip address** *ip-address-mask*	Configures the IP address to be used as BGP endpoints.
Step 21	switch (config-if) # **fabric forwarding control-segment**	Specifies this interface to be the DFA control segment. Only one interface can be this type.
Step 22	switch (config) # **route-map** *map-tag*	Enters route map configuration mode and specifies a route map by identifying the route map name (map-tag). The Maximum size is 63 characters. This name should be the same as the name that you use to configure the BGP additional paths.
Step 23	switch (config-route-map) # **set path-selection all advertise**	Sets the path selection criteria for BGP.
Step 24	switch (config-s) # **ip access-list** *access-list-name*	Defines an IP4 access list access control list (ACL) in order to enable filtering for packets.
Step 25	switch (config-s-acl) # **permit ip** *source destination*	Creates an access control list (ACL) rule that permits traffic that matches its conditions. The source destination identifies the source network address and the destination network address.
Step 26	switch (config) # **ipv6 access-list** *access-list-name*	Creates an IPv6 access control list (ACL) or enters IP access list configuration mode for a specific ACL.
Step 27	switch (config-acl) # *sequence-number* **permit** *protocol*	Configures a permit rule in an IPv6 ACL.

	Command or Action	Purpose
Step 28	switch (config) # **route-map** *map-tag* [**deny** | **permit**] [*sequence-number*]	Predefines a route map for redistribution of HMM host routes. The name should be the same as name that you used when you entered the BGP **redistribute-hmm route map** command. Use the**permit** command to specify that the route or packet is not distributed if the match criteria are met for the route map. Use the **permit** command to specify that the route or packet is distributed if the match criteria for this route are met. The *sequence-number* indicates the position a new route map has in the list of map routes already configured with same name. The no form of this command deletes the position of the route map. The range is 0 to 65535.
Step 29	switch (config-route-map) # **match interface** {*interface-type number* [,*interface-type number...*]}	Matches an interface in a route map. Use the **match interface** command to provide a list of interfaces to match a route against. The route next-hop addresses that are reached by one of these interfaces result in a match for the route map.
Step 30	switch (config) # **route-map** *map-tag* [**deny** | **permit**] [*sequence-number*]	Specifies a route map by identifying the route map name (map-tag). The maximum size is 63 characters. Use the**permit** command to specify that the route or packet is not distributed if the match criteria are met for the route map. Use the **permit** command to specify that the route or packet is distributed if the match criteria for this route are met. The *sequence-number* indicates the position a new route map has in the list of map routes already configured with same name. The no form of this command deletes the position of the route map. The range is 0 to 65535.
Step 31	switch (config-route-map) # **match ip address** {**prefix-list***prefix-list name* [*prefix-list name...*]}	Distributes routes that have a destination IPv6 network number address that is permitted by a standard access list, an expanded list, or a prefix list. The *prefix-list name* can be any alphanumeric string up to 63 characters. The ellipsis indicates that multiple values can be entered, up to 32 prefix lists.
Step 32	switch (config-route-map) # **match interface** {*interface-type number* [,*interface-type number...*]}	Matches an interface in a route map. Use the **match interface** command to provide a list of interfaces to match a route against. The route next-hop addresses that are reached by one of these interfaces result in a match for the route map.
Step 33	switch (config-route-map) # **match ip address** *prefix-list name* [*prefix-list name..*] *access-list-name*	Distributes routes that have a destination IPv6 network number address that is permitted by a standard access list, an expanded list, or a prefix list. The *prefix-list name* can be any alphanumeric string up to 63 characters. The ellipsis indicates that multiple values can be entered, up to 32 prefix lists.
Step 34	Device (config) # **router bgp** *as-number*	Configures a BGP process for an interface. The *as-number* is the number of an autonomous system that identifies the router to other BGP routers and tags that the routing information passed along. The AS number can be a 16-bit integer or a 32-bit integer in the form of a higher 16-bit decimal number and a lower 16-bit decimal number in xx.xx format.

	Command or Action	Purpose
Step 35	Device (config-router) # **address-family ipv4 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode and configures submode commands for the BGP.
Step 36	Device (config-router-af) # **redistribute hmm route-map** *map-name*	Enables redistribution of IPv4 and IPv6 Host Mobility Manager (HMM) routes through specific route maps.
Step 37	switch (config-router-af) # **maximum-paths** [**ibgp**] *number-paths*	Controls the maximum number of parallel routes that the BGP can support.
Step 38	switch (config-router-af) # **additional-paths receive**	Receives additional paths to and from the BGP peers.
Step 39	switch (config-router) # **address-family ipv6 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode and configure submode commands for the BGP.
Step 40	switch (config-router-af) # **redistribute hmm route-map** *map-name*	Redistributes IPv4 and IPv6 Host Mobility Manager (HMM) routes through specific route maps.
Step 41	switch (config-router-af) # **maximum-path** [**ibgp**] *number-paths*	Controls the maximum number of parallel routes that the BGP can support.
Step 42	switch (config-router-af) # **additional-paths-receive**	Receives additional paths to and from the BGP peers.
Step 43	switch (config) # **address-family vpnv4 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode and configure submode commands for the BGP.
Step 44	switch (config-router-af) # **additional-paths receive**	Receives additional paths to and from the BGP peers.
Step 45	switch (config-router) # **address-family vpnv6 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode and configure submode commands for the BGP.
Step 46	switch (config-router-af) # **additional-paths receive**	Receives additional paths to and from the BGP peers.
Step 47	switch (config-router) # **neighbor** {*ip-addr* |*ip-prefixlentgth*} [**remote-as** {*as-num* [,*as-num*] |**route-map** *map name*}	Configures a BGP neighbor (router, VRF) and enters neighbor configuration mode.
Step 48	switch (config-router-neighbor) # **address-family ipv4 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode to configure submode commands for the BGP.
Step 49	switch (config-router-neighbor-af) # **send community** *text*	Sends a message to the active user session. The text string can be up to 80 alphanumeric characters and is case-sensitive.

This example shows how to configure the core for a border leaf.

```
switch # configure terminal
switch (config)# install feature-set fabricpath
switch (config)# install feature-set fabric
switch (config)# feature-set fabricpath
switch (config)# feature fabric forwarding
```

```
switch (config)# feature bgp
switch (config)# feature isis
switch (config)# feature interface-vlan
switch (config)# feature vn-segment-vlan-based

switch (config)# system fabric dynamic-vlans 20-21, 201-202, 1001-1010
switch (config)# system fabric core-vlans 1001-1010
switch (config)# fabric forwarding identifier 100
switch (config)# fabric forwarding anycast-gateway-mac.DEAD.0000.DEAF
switch (config)# fabric forwarding switch-role border-leaf
switch (config)# fabricpath domain default

switch (config)# vlan 1001-1010
switch (config-vlan)# mode fabricpath

switch (config) # interface Vlan10
switch (config-if) # no shutdown
switch (config-if) # ip address 1.1.1.4/24
switch (config-if) # fabric forwarding control-segment

switch (config) # route-map ALL-PATHS permit 10
switch (config-route-map) # set path-selection all advertise

switch (config-s)# ip access-list HOSTS
switch (config-s-acl)# 10 permit ip any any
switch (config-s)# ipv6 access-list hosts-v6
switch (config-s-acl)# 10 permit ipv6 any any

switch (config) # route-map AM deny 10
switch (config-route-map) # match interface Vlan10
switch (config) # route-map AM permit 20
switch (config-route-map) # match ip address HOSTS
switch (config) # route-map hosts-v6 permit 20
switch (config-route-map) # match ipv6 address hosts-v6

switch (config) # router bgp 100
switch (config-router) # address-family ipv4 unicast
switch (config-router-af) # redistribute hmm route-map AM
switch (config-router-af) # maximum-paths ibgp 2
switch (config-router-af) # additional-paths receive
switch (config-router-af) # additional-paths selection route-map ALL PATHS
switch (config-router) # address-family ipv6 unicast
switch (config-router-af) # redistribute hmm route-map hosts-v6
switch (config-router-af) # maximum-paths ibgp 2
switch (config-router-af) # additional-paths receive
switch (config-router-af) # addtional-path seelction route-map ALL PATHS
switch (config-router) # address-family vpnv4 unicast
switch (config-router-af) # additional-paths receive
switch (config-router) # address-family vpnv6 unicast
switch (config-router-af) # additional-paths receive
switch (config-router) # neighbor 1.1.1.1 remote-as 100
switch (config-router-neighbor) # address-family ipv4 unicast
switch (config-router-neighbor-af) # send-community both
```

Adding a Host-Facing Tenant Interface (VLAN)

You can add a host-facing tenant interface (VLAN) to allocate a new VLAN ID and an unused VNI and tie them together, create the corresponding Layer-3 interface and put it into the VRF, and configure the appropriate fabric forwarding mode.

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch # **configure terminal**	Enters global configuration mode.
Step 2	switch (config) # **vlan** {*vlan-id* | *vlan-range*}	Specifies the VLAN IDs of the allowed FabrichPath VLANs in the anycast bundle. The *vlan-id* range is from 2 to 4094.
Step 3	switch (config-vlan) # **mode fabricpath**	Enables the VLAN as a FabricPath VLAN.
Step 4	switch (config-vlan) # **vn-segment** *vni*	Configures the virtual network (VN) segment id of the VLAN.
Step 5	switch (config-profile-vrf) # **interface vlan***vlan-id*	Specifies an interface type and number. The *vlan-id* range is from 2 to 4094.
Step 6	switch (config-if) # **vrf member** *name*	Creates a VPN routing and forwarding instance (VRF) or enters VRF configuration mode to configure submode commands for the Intermediate System-to-Intermediate System (IS-IS) Intradomain Routing Protocol.
Step 7	switch (config-if) # **ip address** *ip-address-mask*	Specifies a primary IP address for an interface.
Step 8	switch (config-if) # [**ip pim sparse-mode**]	Enables IPv4 Protocol Independent Multicast (PIM) sparse mode on an interface.
Step 9	switch (config-if) # **fabric forwarding mode anycast-gateway**	Enables the TF mode in DFA.
Step 10	switch (config-if) # **no shutdown**	Disables the shutdown function on a BGP instance.

The following adds a host-facing tenant interface (VLAN).

```
switch # configure terminal
//Enter configuration commands, one per line. End with CNTL/Z.
switch (config)# vlan 20
switch (config-vlan)# mode fabricpath
switch (config-vlan)# vn-segment 20
switch (config-vlan)# interface vlan 20
switch (config-if) # vrf member VRF2
//Warning: Deleted all L3 config on interface Vlan20
switch (config-if) # ip address 1.1.1.4/24
switch (config-if) # [ip pim sparse-mode]
switch (config-if) # no shutdown
switch (config-if) # fabric forwarding mode anycast-gateway
switch (config-if) # exit
```

Adding a Tenant (VRF) Instance on a Leaf

To add a tenant instance, perform the following:

- Configure a profile named **vrf-tenant-profile**
- Allocate a VLAN

- Create a VRF instance
- Configure the route distinguisher and route targets
- Tie the VNI/segment ID to the VRF instance
- Create a Layer-3 VLAN and configure it with the same IP address/mask as the fabric control VLAN interface to map the BGP endpoint and the VRF BD VLAN

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch # **configure terminal**	Enters global configuration mode.
Step 2	switch # **configure profile** *vrf-tenant-profile*	Configures a profile and enters configuration profile mode to configure profile parameters.
Step 3	switch (config-profile) # **apply profile** *vrf-tenant-profile*	Applies the VRF tenant profile configuration profile to configure hosts.
Step 4	switch (config-profile) # **vlan** {*vlan-id* | *vlan-range*}	Specifies the VLAN IDs of the allowed FabricPath VLANs in the anycast bundle. The *vlan-id* range is from 1 to 4094.
Step 5	switch (config-profile-vlan) # **mode fabricpath**	Enables the VLAN as a FabricPath VLAN.
Step 6	switch (config-profile-vlan) # **vn-segment** *segment-id*	Configures the virtual network (VN) segment ID of the VLAN. The *segment-id* range is from 4096 to 16773119.
Step 7	switch (config-profile) # **vrf context** *name*	Creates a virtual routing and forwarding (VRF) instances and enters VRF configuration mode. The name of the VRF can be any case-sensitive, alphanumeric string up to 32 characters.
Step 8	switch (config-profile-vrf) # **rd** *route-distinguisher*	Creates routing and forwarding tables.
Step 9	switch (config-profile-vrf) # **address-family-ipv4 unicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode and configures submode commands for the BGP.
Step 10	switch (config-profile-vrf-af) # **route-target import** *route-target-ext-community*	Imports routing information from the target virtual private network (VPN) extended community.
Step 11	switch (config-profile-vrf-af) # **route-target export** *route-target-ext-community*	Exports routing information from the target virtual private network (VPN) extended community.
Step 12	switch (config-profile-vrf) # **vni** [*vni-id* | [*-vni-id*]]	Configures the virtual network identifier (VNI). **Note** You can specify a single ID or a range. For example, 4099, 5000-5005.
Step 13	switch (config-profile-vrf) # **interface vlan***vlan-id*	Specifies an interface type and number. The *vlan-id* range is from 1 to 4094.
Step 14	switch (config-profile-if-vrf) # **vrf member** *name*	Creates a VPN routing and forwarding instance (VRF) or enters VRF configuration mode to configure submode commands for the Intermediate System-to-Intermediate System (IS-IS) Intradomain

	Command or Action	Purpose
		Routing Protocol. The *name* can be any case-sensitive, alphanumeric string up to 32 characters.
Step 15	switch (config-profile-if-vrf) # **ip address** *ip-address-mask*	Specifies a primary IP address for an interface
Step 16	switch (config-profile-if-vrf) # **no shutdown**	Disables the shutdown function on a BGP instance.
Step 17	switch (config-profile-if) # **router bgp** *as-number*	Configures a BGP process for an interface. The *as-number* is the number of an autonomous system that identifies the router to other BGP routers and tags that the routing information passed along. The AS number can be a 16-bit integer or a 32-bit integer in the form of a higher 16-bit decimal number and a lower 16-bit decimal number in xx.xx format.
Step 18	switch (config-profile-if) # **vrf** *name*	Creates a VPN routing and forwarding (VRF) instance or enters VRF configuration mode to configure submode commands for the Intermediate System-to-Intermediate System (IS-IS) Intradomain Routing Protocol. The *name* can be any case-sensitive, alphanumeric string up to 32 characters.
Step 19	switch (config-profile-if-vrf) # **address-family ipv4 multicast**	Enters address family mode or a virtual routing and forwarding (VRF) address-family mode and configures submode commands for BGP.
Step 20	switch (config-profile-if-vrf-af) # **redistribute hmm route-map** *map-name*	Redistributes IPv4 Host Mobility Manager (HMM) routes through specific route maps.

This example shows how to configure the profile name and add the tenant VRF profile and associated parameters:

```
switch # configure profile vrf-tenant-profile
//Enter config profile mode, name = vrf-tenant-profile
//Enter configuration commands, one per line. End with CNTL/Z.
switch (config-profile)# configure terminal
//Exit configure profile mode.
switch (config)# apply profile vrf-tenant-profile
switch (config)# vlan 20
switch (config-profile-vlan)# mode fabricpath
switch (config-profile-vlan)# vn-segment 5000
switch (config-profile-vlan)# vrf context vrf2
switch (config-profile-vrf) # rd auto
switch (config-profile-vrf) # address-family ipv4 unicast
switch (config-profile-vrf-af-ipv4) # route-target import 7000:1
switch (config-profile-vrf-af-ipv4) # route-target export 7000:1
switch (config-profile-vrf-af-ipv4) # vni 7000
switch (config-profile-vrf-af-ipv4) # interface vlan 20
switch (config-profile-if-verify) # vrf member VRF2
switch (config-profile-if-verify) # ip address 1.1.1.4/24
switch (config-profile-if-verify) # no shutdown
switch (config-profile-if-verify) # router bgp 100
switch (config-profile-router) # vrf VRF2
switch (config-profile-router-vrf) # address-family ipv4 multicast
switch (config-profile-router-vrf-af) # address-family ipv4 unicast
switch (config-profile-router-vrf-af) # redistribute hmm route-map AM
```

Removing HSRP Configuration on all Border Leafs

During the migration, some hosts start learning the anycast gateway MAC address and will start using it. HSRP is required until the last leaf pair is upgraded to the DFA configuration.

Note HSRP/VRRP is required for VLANs where hosts are connected behind a Cisco Nexus 5000 Series switch in the topology for those VLANs.

You can remove the HSRP configuration on border leafs after you migrate all of the switches.

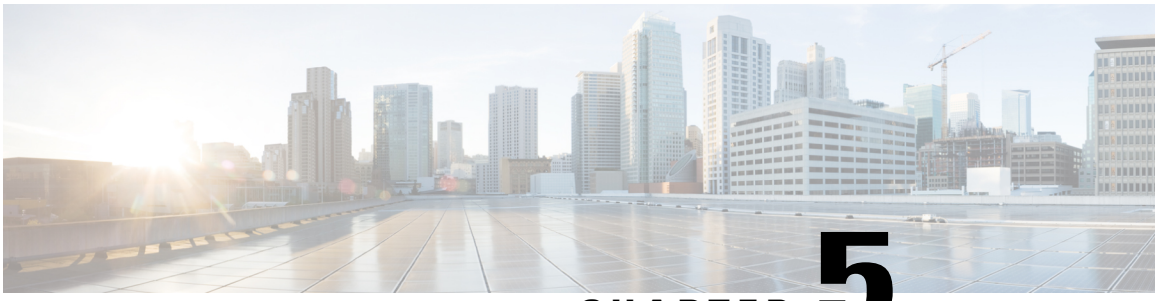
DETAILED STEPS

	Command or Action	Purpose
Step 1	switch # **show running-config interface** *type-number*	Displays the interface for the VLAN.
Step 2	switch (config) # **interface vlan** *vlan-id*	Creates a VLAN interface and enters interface configuration mode. The *vlan-id* range is from 1 to 4094.
Step 3	switch (config-if-hsrp) # **no hsrp** *group-number*	Disables HSRP.
Step 4	switch (config-if-hsrp) # **show running-config interface** *type-number*	Displays an interface type and number.

This example shows how to disable the HSRP configuration on a border leaf:

```
switch # show running-config interface vlan20

!Command: show running-config interface Vlan20
!Time: Tue Jun 9 17:56:19 2015

version 7.2(0)N1(1)

interface Vlan20
  no shutdown
  no ip redirects
  ip address 20.1.1.100
  ipv6 address 20:1::100/64
  ip router ospf 1 area 0.0.0.0
  fabric forwarding mode anycast-gateway
  hsrp version 2
  hsrp 20
    preempt
    priority 110
    ip 20.1.1.100
  hsrp 20 ipv6
    preempt
    priority 110
    ip 20:1::10
  hsrp 50
    mac-address DEAD.0000.DEAF
    preempt
    priority 110
    ip 20.1.1.200
```

```
switch # configure terminal
switch (config) # interface vlan 20
switch (config-if) # no hsrp 50
switch (config-if) # show running-config interface vlan 20

!Command: show running-config interface Vlan20
!Time: Tue Jun  9 17:58:21 2015

version 7.2(0)N1(1)

interface Vlan20
  no shutdown
  no ip redirects
  ip address 20.1.1.100
  ipv6 address 20:1::100/64
  ip router ospf 1 area 0.0.0.0
  fabric forwarding mode anycast-gateway
  hsrp version 2
  hsrp 20
    preempt
    priority 110
    ip 20.1.1.100
  hsrp 20 ipv6
    preempt
    priority 110
    ip 20:1::10

switch (config-if) # interface vlan 20
switch (config-if) # no hsrp 20 ipv4
switch (config-if) # show running-config interface vlan 20

!Command: show running-config interface Vlan20
!Time: Tue Jun  9 17:59:01 2015

version 7.2(0)N1(1)

interface Vlan20
  no shutdown
  no ip redirects
  ip address 20.1.1.100
  ipv6 address 20:1::100/64
  ip router ospf 1 area 0.0.0.0
  fabric forwarding mode anycast-gateway
  hsrp version 2
  hsrp 20 ipv6
    preempt
    priority 110
    ip 20:1::10

switch (config-if) # interface vlan 20
switch (config-if) # no hsrp 20 ipv6
switch (config-if) # show running-config interface vlan 20

!Command: show running-config interface Vlan20
!Time: Tue Jun  9 17:59:27 2015

version 7.2(0)N1(1)

interface Vlan20
  no shutdown
  no ip redirects
  ip address 20.1.1.100
  ipv6 address 20:1::100/64
  ip router ospf 1 area 0.0.0.0
  fabric forwarding mode anycast-gateway
  hsrp version 2

switch (config-if) # interface vlan 20
switch (config-if) # no hsrp version 2
```



CHAPTER 5

Troubleshooting the Migration

This chapter contains the following sections:

Troubleshooting the Cisco Dynamic Fabric Automation (DFA) Migration

After you completed the software and topology migration from FabricPath to the Cisco Dynamic Fabric Automation (DFA) solution, you can use the following checklist to troubleshoot problems with the migration. For more information on troubleshooting, see Cisco Dynamic Fabric Automation Troubleshooting Guide.

Checklist	Complete?
Verifying That Unicast Connectivity Is Established, on page 48	
Verifying That BGP Sessions Are Established, on page 49	
Verifying the VNI, on page 50	
Verifying That the Host Is Learned from ARP and That the Adjacency Table is Properly Updated, on page 51	
Verifying That the HSRP Is Up and ARP Entries Are Updated on Both vPC Peers, on page 52	
Verifying the Port and Virtual Port-Channel Status, on page 53	
Verifying the RIB Entry , on page 54	
Verifying That the BGP Configuration is Enabled on the Remote Leaf , on page 55	
Verifying the Proper IS-IS FabricPath Adjacency, on page 57	
Verifying the IS-IS FabricPath Topology and Database, on page 59	
Verifying That Leafs and Border Leafs Have RP Reachability, on page 61	
Verifying That Multicast Routes Are Properly Propagated, on page 63	

Verifying That Unicast Connectivity Is Established

You can verify that the unicast connectivity is established.

Step 1 On the leaf or border leaf that connects to the sources, send a unicast **ping** command to the receiver. In this example, vpn1 is the name of the vrf, 188.0.0.1 is the unicast address where the receiver is located, and 199.0.0.2 is the local address on this leaf under vrf vpn1.

Example:

```
leaf # ping 188.0.0.1 vrf vpn1 source 199.0.0.2
PING 188.0.0.1 (188.0.0.1) from 199.0.0.2: 56 data bytes
64 bytes from 188.0.0.1: icmp_seq=0 ttl=253 time=1.541 ms
64 bytes from 188.0.0.1: icmp_seq=1 ttl=253 time=1.237 ms
64 bytes from 188.0.0.1: icmp_seq=2 ttl=253 time=1.204 ms
64 bytes from 188.0.0.1: icmp_seq=3 ttl=253 time=1.179 ms
64 bytes from 188.0.0.1: icmp_seq=4 ttl=253 time=1.183 ms
--- 188.0.0.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 1.179/1.268/1.541 ms
```

Step 2 If the ping fails, verify if the unicast route is present on the leaf or border leaf.

Example:

```
leaf #  show ip route vrf vpn1
IP Route Table for VRF "vpn1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
0.0.0.0/0, ubest/mbest: 2/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
1.1.0.0/24, ubest/mbest: 1/0, attached
    *via 1.1.0.1, Vlan10, [0/0], 4d17h, direct
1.1.0.1/32, ubest/mbest: 1/0, attached
    *via 1.1.0.1, Vlan10, [0/0], 4d17h, local
19.19.19.19/32, ubest/mbest: 1/0
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
155.0.0.0/24, ubest/mbest: 2/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
166.0.0.0/24, ubest/mbest: 1/0
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
166.0.0.1/32, ubest/mbest: 1/0
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
177.0.0.0/24, ubest/mbest: 1/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
177.0.0.1/32, ubest/mbest: 1/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
188.0.0.0/24, ubest/mbest: 1/0
    *via 1.1.0.2%default, [200/0], 4d17h, BGP-100, internal, tag 100
188.0.0.1/32, ubest/mbest: 1/0
    *via 1.1.0.2%default, [200/0], 4d16h, BGP-100, internal, tag 100
199.0.0.0/24, ubest/mbest: 1/0, attached
    *via 199.0.0.2, Vlan110, [0/0], 4d17h, direct
199.0.0.1/32, ubest/mbest: 1/0, attached
```

```
    *via 199.0.0.1, Vlan110, [190/0], 4d16h, hmm
199.0.0.2/32, ubest/mbest: 1/0, attached
    *via 199.0.0.2, Vlan110, [0/0], 4d17h, local

leaf #  sh ip route vrf vpn1
IP Route Table for VRF "vpn1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
0.0.0.0/0, ubest/mbest: 2/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
1.1.0.0/24, ubest/mbest: 1/0, attached
    *via 1.1.0.2, Vlan10, [0/0], 4d17h, direct
1.1.0.2/32, ubest/mbest: 1/0, attached
    *via 1.1.0.2, Vlan10, [0/0], 4d17h, local
19.19.19.19/32, ubest/mbest: 1/0
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
155.0.0.0/24, ubest/mbest: 2/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
166.0.0.0/24, ubest/mbest: 1/0
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
166.0.0.1/32, ubest/mbest: 1/0
    *via 1.1.0.4%default, [200/0], 4d17h, BGP-100, internal, tag 100
177.0.0.0/24, ubest/mbest: 1/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
177.0.0.1/32, ubest/mbest: 1/0
    *via 1.1.0.3%default, [200/0], 4d17h, BGP-100, internal, tag 100
188.0.0.0/24, ubest/mbest: 1/0, attached
    *via 188.0.0.2, Vlan110, [0/0], 4d17h, direct
188.0.0.1/32, ubest/mbest: 1/0, attached
    *via 188.0.0.1, Vlan110, [190/0], 4d16h, hmm
188.0.0.2/32, ubest/mbest: 1/0, attached
    *via 188.0.0.2, Vlan110, [0/0], 4d17h, local
199.0.0.0/24, ubest/mbest: 1/0
    *via 1.1.0.1%default, [200/0], 4d17h, BGP-100, internal, tag 100
199.0.0.1/32, ubest/mbest: 1/0
    *via 1.1.0.1%default, [200/0], 4d16h, BGP-100, internal, tag 100
```

What to Do Next

Verify that the BGP sessions are established.

Verifying That BGP Sessions Are Established

You can verify that the Border Gateway Protocol (BGP) session is established.

Before You Begin

Verify that unicast connectivity is established.

Step 1 Verify that the BGP session on the leaf is attached to the source and verify that the session with the route reflector is established.

Example:

```
leaf #  show BGP session
Total peers 1, established peers 1
```

```
ASN 100
VRF default, local ASN 100
peers 1, established peers 1, local router-id 1.1.0.1
State: I-Idle, A-Active, O-Open, E-Established, C-Closing, S-Shutdown
Neighbor        ASN    Flaps LastUpDn|LastRead|LastWrit St Port(L/R)  Notif(S/R)
1.1.0.5           100 0     4d17h   |00:00:05|00:00:35 E  22421/179  0/0
```

Step 2 Verify the BGP session on the spine route reflector and verify that the session with the leafs/border leafs are established.

Example:

```
spine_route reflector #  show BGP session
Total peers 4, established peers 4
ASN 100
VRF default, local ASN 100
peers 4, established peers 4, local router-id 1.1.0.5
State: I-Idle, A-Active, O-Open, E-Established, C-Closing, S-Shutdown
Neighbor        ASN    Flaps LastUpDn|LastRead|LastWrit St Port(L/R)  Notif(S/R)
1.1.0.1           100 0     4d17h   |00:00:46|00:00:15 E  179/22421  0/0
1.1.0.2           100 0     4d17h   |00:00:20|00:00:31 E  179/34401  0/0
1.1.0.3           100 0     4d17h   |00:00:49|00:00:46 E  179/14080  0/0
1.1.0.4           100 0     4d17h   |00:00:26|00:00:15 E  179/7980   0/0
```

What to Do Next

Verify that the correct virtual network identifier (VNI) is being used.

Verifying the VNI

You can verify that the correct virtual network identifier (VNI) is being used.

Because of a limitation with a Cisco Nexus 7000 Series spine switch that is running in transit mode, the segment-ID and VNI range that you can use on the leafs/border leafs are determined by the VLAN configured on the spine switch.

VLAN is configured on spine for the following reasons:

- Control-segment SVI for BGP route reflector termination on the spine (not applicable if running route reflector on leaf)
- Flooding of BUM traffic only to leaf nodes where the VLAN is defined. If VLAN is not configured in spine, BUM traffic are flooded to all leaf nodes. For Cisco Nexus 6000 Series Switches, config VLAN in 'fabricpath mode'. For Cisco Nexus 7000 Series Switches, config VLAN in 'fabricpath mode' and underlying SVI in no-shut state. 4K VNIs are reserved for each VLAN configured on the Spine (not usable as VLAN segment IDs). Reserved VNIs are from <*vlan-id*>*4096 to (<*vlan-id*>*4096)+4095.

Before You Begin

- Verify that the unicast connectivity is established.
- Verify that the Border Gateway Protocol (BGP) sessions are established.

Step 1 Determine which VLANs are configured on the Cisco Nexus 7000 Series spines. In the following example, VLAN 2 is configured on the spine route reflector (RR).

Example:

```
spine_rr #  show run | inc vlan
feature interface-vlan
vlan 1
```

Step 2 Determine which vn-segment and VNI are configured on the leafs and border leafs. In the following example, VNI and vn-segment 9000 are configured on this leaf.

Example:

```
source_leaf #  show  run | i vn-segment
feature vn-segment-vlan-based
  vn-segment 9000
source_leaf # show  run | i vni
  vni 9000

receiver_leaf # show run | i vn-segment
feature vn-segment-vlan-based
  vn-segment 9000
source_leaf# show  run | i vni
  vni 9000
```

Step 3 Verify if the VNI/vn-segment can be used.
If the VLAN you found in Step 1 is "n", you cannot use VNI/vn-segment from n*4096 to (n*4096)+4095. In this example, you cannot use VNIs 4096 to 8191. In Step 2, you determined that you are using VNI/vn-segment 9000, which is outside the forbidden range. Therefore, the configuration is correct.

What to Do Next

Verify that the host is learned through Address Resolution Protocol (ARP) and that the adjacency table is properly updated.

Verifying That the Host Is Learned from ARP and That the Adjacency Table is Properly Updated

You can verify that the host is learned from address resolution protocol (ARP) and that the adjacency table is properly update.

Before You Begin

- Verify that the unicast connectivity is established.
- Verify that the border gateway protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used .

Step 1 Verify that the adjacency table is updated properly.

Example:

```
leaf #  show ip arp vrf default<<< Make sure to check for valid vrf
Flags: * - Adjacencies learnt on non-active FHRP router
       + - Adjacencies synced via CFSoE
```

```
       # - Adjacencies Throttled for Glean
       D - Static Adjacencies attached to down interface
IP ARP Table for context default
Total number of entries: 1
Address         Age       MAC Address     Interface
90.99.0.1       00:00:12  000c.29e2.104b  Vlan90   <<< shows 90.99.0.1 is learnt
```

Step 2 Verify that the IP adjacency table is properly populated.

Example:

```
switch # show ip adjacency vrf default<<< Make sure to check for valid vrf
Flags: # - Adjacencies Throttled for Glean
       G - Adjacencies of vPC peer with G/W bit
IP Adjacency Table for VRF default
Total number of entries: 10
Address         MAC Address     Pref Source     Interface
90.99.0.1       000c.29e2.104b  50   arp        Vlan90
```

What to Do Next

Verify that the Hot Standby Router Protocol (HSRP) is up and Address Resolution Protocol (ARP) entries are updated on both vPC peers.

Verifying That the HSRP Is Up and ARP Entries Are Updated on Both vPC Peers

You can verify that the Hot Standby Router Protocol (HSRP) is up and that the Address Resolution Protocol (ARP) entries are updated on both virtual port channels (vPCs).

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.

Step 1 Verify the Hot Standby Router Protocol (HSRP) status.

Example:

```
leaf #  show hsrp
Vlan13 - Group 13 (HSRP-V1) (IPv4)
  Local state is Active, priority 90 (Cfged 90), may preempt
    Forwarding threshold(for vPC), lower: 1 upper: 90
  Hellotime 3 sec, holdtime 10 sec
  Virtual IP address is 10.75.13.1 (Cfged)
  Active router is local
  Standby router is 10.75.13.5, priority 80 expires in 8.607000 sec(s)
  Authentication text "cisco"
  Virtual mac address is 0000.0c07.ac0d (Default MAC)
  0 state changes, last state change never
  IP redundancy name is hsrp-Vlan13-13 (default)
```

Step 2 Verify the vPC port status.

Example:

```
leaf #  show vpc
Legend:
(*) - local vPC is down, forwarding via vPC peer-link
vPC domain id                     : 1
vPC+ switch id                    : 1001
Peer status                       : peer adjacency formed ok
vPC keep-alive status             : peer is alive
vPC fabricpath status             : peer is reachable through fabricpath
Configuration consistency status  : success
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : secondary
Number of vPCs configured         : 1
Peer Gateway                      : Enabled
Peer gateway excluded VLANs     : -
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Enabled (timeout = 240 seconds)
vPC Peer-link status
---------------------------------------------------------------------
id   Port   Status Active vlans
--   ----   ------ --------------------------------------------------
1    Po100  up     1,90,100,121-122,125-126,1201-1202,1205-1206
vPC status
----------------------------------------------------------------------------
id     Port        Status Consistency Reason       Active vlans vPC+ Attrib
--     ----------  ------ ----------- ------       ------------ -----------
20     Po1         up*  success     success      -            DF: No, FP
                                                                MAC: 1001.0.0
```

What to Do Next

Verify the member port status and that the vPC leg port channel is up.

Verifying the Port and Virtual Port-Channel Status

You can verify that the both port and that the vPC port-channel are up.

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port channel (vPC) peers.

Step 1 Determine the port-channel status.

Example:

```
leaf #  show port-channel summary interface port-channel 1
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        S - Switched    R - Routed
        U - Up (port-channel)
        M - Not in use. Min-links not met
--------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------------
1     Po1(SD)     Eth      LACP      Eth1/35(D)
switch#
```

Step 2 Determine the interface status. If it is down due to a peer link, it might require an autorecovery to timeout before the interface status is up. The default autorecovery timeout default is 240 seconds.

Example:

```
leaf # show int eth 1/35 | head
Ethernet1/35 is up (vpc peerlink is up)
 Dedicated Interface
  Belongs to Po1
  Hardware: 1000/10000 Ethernet, address: 002a.6a22.c1ca (bia 002a.6a22.c1ca)
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  Port mode is trunk
  auto-duplex, 1000 Mb/s, media type is 1G
  Beacon is turned off
```

What to Do Next

Verify the route in the Routing Information Base (RIB) entry and that the RIB is properly populated.

Verifying the RIB Entry

You can verify that the route in the routing information base (RIB) entry is correct and that the RIB is properly populated .

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port channel (vPC) peers.

- Verify the port and vPC port channel status.

Step 1 Verify the RIB local route and ensure to check for HMM route in RIB.

Example:

```
leaf # show ip route vrf vrf2 2.10.1.2/32
IP Route Table for VRF "vrf2"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
2.10.1.2/32, ubest/mbest: 1/0, attached
    *via 2.10.1.2, Vlan2, [190/0], 2d17h, hmm
```

Step 2 Verify that the Border Gateway Protocol (BGP) configuration exists.

Example:

```
leaf #show run BGP 100
  router-id 1.1.100.22
neighbor 1.1.100.11
    remote-as 200
vrf vrf10
    address-family ipv4 unicast
      redistribute hmm route-map hmm-to-BGP <<< Make sure to check  that the hmm routes are being
redistibuted

leaf #  show run rpm
!Command: show running-config rpm
!Time: Fri Mar  7 23:04:31 2014
version 7.0(1)N1(1)
route-map hmm-to-BGP permit 10 <<< Make sure to check for route-map exist
```

What to Do Next

Verify that the remote hosts route appears in both the RIB and the BGP RIB.

Verifying That the BGP Configuration is Enabled on the Remote Leaf

If the remote Border Gateway Protocol (BGP) routes do not appear in the BGP routing information base (RIB), you can verify that the BGP configuration is properly enabled on the remote leaf.

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port channel (vPC) peers.

- Verify the port and vPC port-channel status.
- Verify the routing information base (RIB) entry.

Step 1 If the remote BGP routes are not seen in BRIB, verify that the BGP configuration is properly enabled on the remote leaf.

Example:

```
leaf # show run BGP
  !Command: show running-config bgp
!Time: Tue May 19 18:08:45 2015

version 7.2(0)N1(1)
feature bgp

router bgp 65000
  router-id 1.1.1.64
  address-family ipv4 unicast
    redistribute hmm route-map AM
    maximum-paths ibgp 2
    additional-paths receive
  address-family ipv6 unicast
    redistribute hmm route-map hosts-v6
    maximum-paths ibgp 2
    additional-paths receive
  address-family vpnv4 unicast
    additional-paths receive
  address-family vpnv6 unicast
    additional-paths receive
  neighbor 1.1.1.61 remote-as 65000
    address-family ipv4 unicast
      send-community both
    address-family ipv6 unicast
      send-community both
    address-family vpnv4 unicast
      send-community extended
      capability additional-paths receive
    address-family vpnv6 unicast
      send-community both
  vrf vrf1
    address-family ipv4 unicast
      redistribute hmm route-map AM
    address-family ipv6 unicast
      redistribute hmm route-map hosts-v6
  vrf vrf2
    address-family ipv4 unicast
      redistribute hmm route-map AM
    address-family ipv6 unicast
      redistribute hmm route-map hosts-v6 vrf context vrf1
  rd auto
  address-family ipv4 unicast
    route-target import 5000:1
    route-target export 5000:1
  address-family ipv6 unicast
    route-target import 5000:1
    route-target export 5000:1
vrf context vrf2
  rd auto
  address-family ipv4 unicast
    route-target import 7000:1
    route-target export 7000:1
  address-family ipv6 unicast
    route-target import 7000:1
```

```
    route-target export 7000:1
```

Step 2 Verify that the virtual route forwarding (VRF) instance has the **route-target** configured as **auto** or **export** on the other end.

Example:

```
leaf # show run vrf vrf2
!Command: show running-config vrf vrf2
!Time: Tue May 19 18:09:26 2015

version 7.2(0)N1(1)
vrf context vrf2
  rd auto
  address-family ipv4 unicast
    route-target import 7000:1
    route-target export 7000:1
  address-family ipv6 unicast
    route-target import 7000:1
    route-target export 7000:1
  vni 7000
router bgp 65000
  vrf vrf2
    address-family ipv4 unicast
      redistribute hmm route-map AM
    address-family ipv6 unicast
      redistribute hmm route-map hosts-v6
```

What to Do Next

Verify the proper Intermediate-system-to-intermediate-system (ISIS) FabricPath adjacency operation.

Verifying the Proper IS-IS FabricPath Adjacency

You can verify the proper Intermediate-system to intermediate-system (ISIS) FabricPath Adjacency operation.

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port-channel (vPC) peers.
- Verify the port and vPC port channel status.
- Verify the Routing Information Base (RIB) entry.

- Verify that the remote host appears in both the RIB and the BGP RIB.

Step 1 Verify the FabricPath IS-IS adjacency on the spine for all the attached leaf nodes.

Example:

```
spine #  show fabricpath isis adjacency
Fabricpath IS-IS domain: default Fabricpath IS-IS adjacency database:
System ID       SNPA            Level  State  Hold Time  Interface
ln1             N/A             1      UP     00:00:30   Ethernet2/1
ln2             N/A             1      UP     00:00:22   Ethernet2/2
ln3             N/A             1      UP     00:00:24   Ethernet2/3
```

Step 2 Verify the FabricPath IS-IS unique switch ID.

Example:

```
leaf #  show fabricpath isis switch-id
Fabricpath IS-IS domain: default
Fabricpath IS-IS Switch-ID Database
Legend: C - Confirmed, T - tentative, W - swap
        S - sticky, E - Emulated Switch
        A - Anycast Switch
        '*' - this system
System-ID        Primary  Secondary  Reachable  Bcast-Priority FtagRootCapable?
MT-0
000e.0100.0030*  3479[C]       0[C]  Yes               64              Y
000e.0101.0030   1599[C]       0[C]  Yes               64              Y
000f.0100.0030   374 [C]       0[C]  Yes               64              Y
000f.0101.0030   1533[C]       0[C]  Yes               64              Y
000f.0102.0030   692 [C]       0[C]  Yes               64              Y
# 3479 is the local switch-id
```

Step 3 Verify FabricPath IS-IS routes.

Example:

```
leaf # show fabricpath isis route
Fabricpath IS-IS domain: default MT-0
Topology 0, Tree 0, Swid routing table
374, L1
via Ethernet2/1, metric 400
692, L1
via Ethernet2/3, metric 400
1533, L1
via Ethernet2/2, metric 400
1599, L1
via Ethernet2/1, metric 800
via Ethernet2/2, metric 800
via Ethernet2/3, metric 800
```

Step 4 Verify the FabricPath IS-IS interface.

Example:

```
leaf # show fabricpath isis interface
Fabricpath IS-IS domain: default
Interface: Ethernet2/1
  Status: protocol-up/link-up/admin-up
  Index: 0x0001, Local Circuit ID: 0x01, Circuit Type: L1
  No authentication type/keychain configured
  Authentication check specified
  Extended Local Circuit ID: 0x1A080000, P2P Circuit ID: 0000.0000.0000.00
  Retx interval: 10, Retx throttle interval: 666 ms
  LSP interval: 33 ms, MTU: 1500
  P2P Adjs: 1, AdjsUp: 1, Priority 64
  Hello Interval: 10, Multi: 3, Next IIH: 00:00:04
  Level   Adjs   AdjsUp  Metric   CSNP  Next CSNP  Last LSP ID
```

```
  1          1        1     400     60  Inactive   ffff.ffff.ffff.ff-ff
  Topologies enabled:
    Topology Metric  MetricConfig Forwarding
    0        400     no           UP
Interface: Ethernet2/2
  Status: protocol-up/link-up/admin-up
  Index: 0x0002, Local Circuit ID: 0x01, Circuit Type: L1
  No authentication type/keychain configured
  Authentication check specified
  Extended Local Circuit ID: 0x1A081000, P2P Circuit ID: 0000.0000.0000.00
  Retx interval: 10, Retx throttle interval: 666 ms
  LSP interval: 33 ms, MTU: 1500
  P2P Adjs: 1, AdjsUp: 1, Priority 64
  Hello Interval: 10, Multi: 3, Next IIH: 00:00:06
  Level   Adjs   AdjsUp  Metric   CSNP  Next CSNP  Last LSP ID
  1          1        1     400     60  Inactive   ffff.ffff.ffff.ff-ff
  Topologies enabled:
    Topology Metric  MetricConfig Forwarding
    0        400     no           UP
Interface: Ethernet2/3
  Status: protocol-up/link-up/admin-up
  Index: 0x0003, Local Circuit ID: 0x01, Circuit Type: L1
  No authentication type/keychain configured
  Authentication check specified
  Extended Local Circuit ID: 0x1A082000, P2P Circuit ID: 0000.0000.0000.00
  Retx interval: 10, Retx throttle interval: 666 ms
  LSP interval: 33 ms, MTU: 1500
  P2P Adjs: 1, AdjsUp: 1, Priority 64
  Hello Interval: 10, Multi: 3, Next IIH: 00:00:02
  Level   Adjs   AdjsUp  Metric   CSNP  Next CSNP  Last LSP ID
  1          1        1     400     60  Inactive   ffff.ffff.ffff.ff-ff
  Topologies enabled:
    Topology Metric  MetricConfig Forwarding
0        400     no           UP


leaf # show ip adjacency
Flags: # - Adjacencies Throttled for Glean
       G - Adjacencies of vPC peer with G/W bit

IP Adjacency Table for VRF default
Total number of entries: 4
Address         MAC Address     Pref Source     Interface
1.1.1.61        002a.6a1e.acbc  1    ISIS       Vlan1
1.1.1.62        002a.6afe.33bc  1    ISIS       Vlan1
1.1.1.63        002a.6afe.32c1  1    ISIS       Vlan1
1.1.1.65        5897.1ef0.780c  1    ISIS       Vlan1
```

What to Do Next

Verify the FabricPath ISIS topology and database and verify that the Unicast Routing Information Base (uRIB) and Unicast Forwarding Information Base (uFIB) are properly populated.

Verifying the IS-IS FabricPath Topology and Database

You can verify that the IS-IS FabricPath topology and database are properly configured and verify that the Unicast Routing Information Base (uRIB) and Unicast Forwarding Information Base (uFIB) are properly populated.

Before You Begin

- Verify that unicast connectivity is established.

- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port channel (vPC) peers.
- Verify the port and vPC port-channel status.
- Verify the routing information base (RIB) entry.
- Verify that the remote host appears in both the RIB and the BGP RIB.
- Verify the proper FabricPath Intermediate-System-to-Intermediate-System (ISIS) adjacency.

Step 1 Determine the FabricPath IS-IS topology.

Example:

```
leaf # show fabricpath isis topology summary
Summary is set!.
FabricPath IS-IS Topology Summary
Fabricpath IS-IS domain: default
MT-0
  Configured interfaces:  Ethernet2/1  Ethernet2/2  Ethernet2/3
Max number of trees: 2  Number of trees supported: 2
    Tree id: 1, ftag: 1, root system: 000f.0100.0030, 2084
    Tree id: 2, ftag: 2, root system: 000f.0102.0030, 3154
Ftag Proxy Root: 000f.0100.0030
```

Step 2 Verify the FabricPath IS-IS database.

Example:

```
leaf #  show fabricpath isis database
Fabricpath IS-IS domain: default LSP database
  LSPID                 Seq Number   Checksum  Lifetime   A/P/O/T
  sn1.00-00           * 0x00000008   0xF4BB    707        0/0/0/1
  sn2.00-00             0x00000008   0x4B7E    705        0/0/0/1
  ln1.00-00             0x00000005   0x1C01    711        0/0/0/1
  ln2.00-00             0x00000006   0x0173    766        0/0/0/1
  ln3.00-00             0x00000004   0x0C46    766        0/0/0/1
sn1# term len 0
```

Step 3 Display the FabricPath IS-IS database detail and verify that the uRIB and uFIB are properly populated.

Example:

```
sn1 # show fabricpath isis database detail
Fabricpath IS-IS domain: default LSP database
  LSPID                 Seq Number   Checksum  Lifetime   A/P/O/T
  sn1.00-00           * 0x00000009   0xF2BC    1029       0/0/0/1
    Instance      :  0x00000009
    Area Address  :  00
    NLPID         :  0xCC 0x8E 0xC0
    Hostname      :  sn1                Length : 3
    Extended IS   :            ln3.00             Metric : 400
    Extended IS   :            ln2.00             Metric : 400
    Extended IS   :            ln1.00             Metric : 400
    Capability    : Device Id: 3479 Base Topology
      Version         :
       Version: 1 Flags: 0
      Nickname        :
```

```
     Priority: 0 Nickname: 3479 BcastPriority: 64
    IP to MAC Mapping :
     MAC: 000e.0100.0030, # IPv4 addr 1, # IPv6 addr 1
       IPv4 address: 110.1.1.1
       IPv6 address: 2110::1
    Nickname Migration :
     Swid: 3479 Sec. Swid: 0
    Base Topo Trees :
     Trees desired: 2  Trees computed: 2  Trees usable: 2
  Digest Offset :  0
sn2.00-00             0x00000009   0x497F    1059       0/0/0/1
  Instance      :  0x00000004
  Area Address  :  00
  NLPID         :  0xCC 0x8E 0xC0
  Hostname      :  sn2                Length : 3
  Extended IS   :            ln3.00             Metric : 400
  Extended IS   :            ln2.00             Metric : 400
  Extended IS   :            ln1.00             Metric : 400
  Capability    : Device Id: 1599 Base Topology
    Version         :
     Version: 1 Flags: 0
    Nickname        :
     Priority: 0 Nickname: 1599 BcastPriority: 64
    IP to MAC Mapping :
     MAC: 000e.0101.0030, # IPv4 addr 1, # IPv6 addr 1
       IPv4 address: 110.1.1.2
       IPv6 address: 2110::2
    Nickname Migration :
     Swid: 1599 Sec. Swid: 0
    Base Topo Trees :
     Trees desired: 2  Trees computed: 2  Trees usable: 2
  Digest Offset :  0
ln1.00-00             0x00000006   0x1A02    1068       0/0/0/1
  Instance      :  0x00000003
  Area Address  :  00
  NLPID         :  0xCC 0x8E 0xC0
  Hostname      :  ln1                Length : 3
  Extended IS   :            sn2.00             Metric : 400
  Extended IS   :            sn1.00             Metric : 400
  Capability    : Device Id: 374 Base Topology
    Version         :
     Version: 1 Flags: 0
```

What to Do Next

Verify that leafs and borders leafs have rendezvous point (RP) reachability.

Verifying That Leafs and Border Leafs Have RP Reachability

You can verify that the leafs and border leafs have rendezvous point (RP) reachability.

Note If you are running Source Specific Multicast (SSM) only, you can skip this step.

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.

- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port-channel (vPC) peers.
- Verify the port and vPC port channel status.
- Verify the Routing Information Base (RIB) entry.
- Verify that the remote host appears in both the RIB and the BGP RIB.
- Verify the proper FabricPath intermediate-system-to-intermediate-system (IS-IS) adjacency.
- Verify the FabricPath IS-IS topology and database.

Step 1 Verify that RP information exists on leafs and border leafs. In this example, vpn1 is the name of the vrf that you are investigating and RP 19.19.19.19 corresponds to a sparse mode group range 239.0.0.0/16.

Example:

```
source_leaf # show ip pim rp vrf vpn1
PIM RP Status Information for VRF "vpn1"
BSR disabled
Auto-RP disabled
BSR RP Candidate policy: None
BSR RP policy: None
Auto-RP Announce policy: None
Auto-RP Discovery policy: None
RP: 19.19.19.19, (0), uptime: 4d20h, expires: never,
  priority: 0, RP-source: (local), group ranges:
      239.0.0.0/16

receiver_leaf # show  ip pim rp vrf vpn1
PIM RP Status Information for VRF "vpn1"
BSR disabled
Auto-RP disabled
BSR RP Candidate policy: None
BSR RP policy: None
Auto-RP Announce policy: None
Auto-RP Discovery policy: None
RP: 19.19.19.19, (0), uptime: 4d20h, expires: never,
  priority: 0, RP-source: (local), group ranges:
      239.0.0.0/16
```

Step 2 Verify that you can ping the RP from leafs and border leafs.

Example:

```
source_leaf # ping 19.19.19.19 vrf vpn1 source 199.0.0.2
PING 19.19.19.19 (19.19.19.19) from 199.0.0.2: 56 data bytes
64 bytes from 19.19.19.19: icmp_seq=0 ttl=254 time=1.829 ms
64 bytes from 19.19.19.19: icmp_seq=1 ttl=254 time=1.092 ms
64 bytes from 19.19.19.19: icmp_seq=2 ttl=254 time=1.075 ms
64 bytes from 19.19.19.19: icmp_seq=3 ttl=254 time=1.078 ms
64 bytes from 19.19.19.19: icmp_seq=4 ttl=254 time=1.07 ms

Receiver_leaf # ping 19.19.19.19 vrf vpn1 source 188.0.0.2
PING 19.19.19.19 (19.19.19.19) from 188.0.0.2: 56 data bytes
64 bytes from 19.19.19.19: icmp_seq=0 ttl=254 time=2.037 ms
64 bytes from 19.19.19.19: icmp_seq=1 ttl=254 time=1.028 ms
64 bytes from 19.19.19.19: icmp_seq=2 ttl=254 time=1.025 ms
```

```
64 bytes from 19.19.19.19: icmp_seq=3 ttl=254 time=1.015 ms
64 bytes from 19.19.19.19: icmp_seq=4 ttl=254 time=0.954 ms
```

What to Do Next

Verify that multicast routes are properly propagated.

Verifying That Multicast Routes Are Properly Propagated

You can verify that multicast routes have been properly propagated.

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both vPC peers.
- Verify the port and vPC port-channel status.
- Verify the Routing Information Base (RIB) entry.
- Verify that the remote host appears in both the RIB and the BGP RIB.
- Verify the proper FabricPath intermediate-system-to-intermediate-system (ISIS) adjacency.
- Verify the FabricPath IS-IS topology and database.
- Verify that leafs and borders leafs have rendezvous point (RP) reachability.

For SSM streams, verify the mroute states on leafs connect to both the source and the receiver.

Note In this example, the host-facing switch virtual interface (SVI) is VLAN 110, and the fabric-facing SVI is VLAN 10.

Step 1 Verify the mroute on the source.

Example:

```
source_leaf #  show ip mroute vrf vpn1
IP Multicast Routing Table for VRF "vpn1"
(*, 232.0.0.0/8), uptime: 4d21h, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)
(199.0.0.1/32, 232.1.1.1/32), uptime: 18:40:02, fabric_mcast ip pim
  Incoming interface: Vlan110, RPF nbr: 199.0.0.1
  Outgoing interface list: (count: 1) (Fabric OIF)
    Vlan10, uptime: 18:40:02, fabric_mcast
```

For the leaf that attaches to the source, verify that incoming interface is the host-facing SVI, while the outgoing interface is the fabric-facing SVI.

Step 2 Verify the mroute on the receiver.

Example:

```
receiver_leaf #  show ip mroute vrf vpn1
IP Multicast Routing Table for VRF "vpn1"
(*, 232.0.0.0/8), uptime: 4d21h, pim ip
  Incoming interface: Null, RPF nbr: 0.0.0.0
  Outgoing interface list: (count: 0)
(199.0.0.1/32, 232.1.1.1/32), uptime: 18:40:10, igmp ip pim
  Incoming interface: Vlan10, RPF nbr: 1.1.0.1
  Outgoing interface list: (count: 1)
    Vlan110, uptime: 18:40:10, igmp feature interface-vlan
```

For the leaf that attaches to the receiver, verify that the incoming interface is the fabric-facing SVI, while the outgoing interface is the host-facing SVI.

For sparse mode streams, which source actively sending traffic, check the mroute states on both leafs that connect to the source and receiver.

Step 3 Verify the mroute on the source.

Example:

```
source_leaf # show ip mroute vrf vpn1
IP Multicast Routing Table for VRF "vpn1"
(199.0.0.1/32, 239.0.0.1/32), uptime: 00:00:11, ip pim fabric_mcast
  Incoming interface: Vlan110, RPF nbr: 199.0.0.1
  Outgoing interface list: (count: 1) (Fabric OIF)
    Vlan10, uptime: 00:00:11, fabric_mcast
For the leaf attaching to the source, make sure the incoming interface is the host facing SVI, while
 the outgoing interface is the fabric facing SVI.
```

Step 4 Use the **show ip mroute vrf** command to verify the mroute on the receiver.

Example:

```
receiver_leaf # show ip mroute vrf vpn1
IP Multicast Routing Table for VRF "vpn1"
(*, 239.0.0.1/32), uptime: 00:00:29, igmp ip pim
  Incoming interface: Vlan10, RPF nbr: 1.1.0.4
  Outgoing interface list: (count: 1)
    Vlan110, uptime: 00:00:29, igmp
(199.0.0.1/32, 239.0.0.1/32), uptime: 00:00:19, ip mrib pim
  Incoming interface: Vlan10, RPF nbr: 1.1.0.1
  Outgoing interface list: (count: 1)
    Vlan110, uptime: 00:00:19, mrib
```

For the leaf attaching to the receiver, make sure the incoming interface is the fabric facing SVI, while the outgoing interface is the host facing SVI.

For bidirectional streams, check the mroute states on leafs that connect to the source and the receiver.

Step 5 Verify the mroute on the source .

Example:

```
Source_leaf #  show ip mroute vrf vpn1
IP Multicast Routing Table for VRF "vpn1"
(*, 238.0.0.0/16), bidir, uptime: 4d22h, pim ip
  Incoming interface: Vlan10, RPF nbr: 1.1.0.4
  Outgoing interface list: (count: 1)
    Vlan10, uptime: 4d22h, pim, (RPF)
```

For the leaf that attaches to the source, you do not see the exact group the receiver joined. Verify that (*,G/m) which matches the joined group. Both incoming and outgoing interfaces should be the fabric- facing SVI.

Step 6 Verify the mroute on the receiver.

Example:

```
receiver_leaf # show ip mroute vrf vpn1
IP Multicast Routing Table for VRF "vpn1"
(*, 238.0.0.0/16), bidir, uptime: 4d22h, pim ip
  Incoming interface: Vlan10, RPF nbr: 1.1.0.4
  Outgoing interface list: (count: 1)
    Vlan10, uptime: 4d22h, pim, (RPF)
(*, 238.0.0.1/32), bidir, uptime: 00:00:06, igmp ip pim
  Incoming interface: Vlan10, RPF nbr: 1.1.0.4
  Outgoing interface list: (count: 2)
    Vlan10, uptime: 00:00:06, pim, (RPF)
    Vlan110, uptime: 00:00:06, igmp
```

For the leaf that attaches to the receiver, verify that the same (*,G/m) entry exist. Also, verify the receiver joined group exists. For this (*,G), the incoming interface is the fabric-facing SVI, while the outgoing interface are both the host-facing SVI and fabric-facing SVI.

What to Do Next

Verify that PIM is autoenabled and that the Designated Router (DR)/Designated Forwarder (DF) is on the SVIs.

Verifying the PIM on the SVI and the DR and DF on the Host-Facing SVI Are Autoenabled

You can verify that the Protocol Independent Multicast (PIM) on the Switch Virtual Interfaces (SVIs) are configured properly and that the Designated Router (DR) and Designated Forwarder (DF) on the host-facing SVI are autoenabled.

Before You Begin

- Verify that unicast connectivity is established.
- Verify that Border Gateway Protocol (BGP) sessions are established.
- Verify that the correct virtual network identifier (VNI) is being used.
- Verify that the host is learned from the Address Resolution Protocol (ARP) and that the adjacency table is properly updated.
- Verify that the Hot Standby Route Protocol (HSRP) is up and ARP entries for the host are updated on both virtual port-channel (vPC) peers.
- Verify the port and vPC port channel status.
- Verify the Routing Information Base (RIB) entry.
- Verify that the remote host appears in both the RIB and the BGP RIB.
- Verify the proper FabricPath intermediate-system to intermediate-system (IS-IS) adjacency.
- Verify the FabricPath IS-IS topology and database.
- Verify that leafs and borders leafs have rendezvous point (RP) reachability.

- Verify that multicast routes are properly propagated.

Step 1 Verify the PIM on the source leaf.

Example:

```
source_leaf #  show ip pim int vrf vpn1
PIM Interface Status for VRF "vpn1"
Vlan10, Interface status: protocol-up/link-up/admin-up
  IP address: 1.1.0.1, IP subnet: 1.1.0.0/24
  PIM DR: 1.1.0.1, DR's priority: 1
  PIM neighbor count: 0
  PIM hello interval: 30 secs, next hello sent in: 0.000000
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x2bf9d0c3
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: yes
  PIM VPC SVI: no
  PIM Auto Enabled: yes
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 0/0 (early: 0), JPs: 0/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 0/0, DF-Winners: 0/0, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 0
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0
Vlan110, Interface status: protocol-up/link-up/admin-up
  IP address: 199.0.0.2, IP subnet: 199.0.0.0/24
  PIM DR: 199.0.0.2, DR's priority: 1
  PIM neighbor count: 0
  PIM hello interval: 30 secs, next hello sent in: 0.000000
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x0ce78912
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: yes
  PIM VPC SVI: no
  PIM Auto Enabled: yes
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 14330/13827 (early: 0), JPs: 831/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
```

```
      DF-Offers: 0/0, DF-Winners: 13616/0, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 595
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0
```

Step 2 Verify the PIM on the receiver leaf.

Example:

```
receiver_leaf #  show ip pim int vrf vpn1
PIM Interface Status for VRF "vpn1"
Vlan10, Interface status: protocol-up/link-up/admin-up
  IP address: 1.1.0.2, IP subnet: 1.1.0.0/24
  PIM DR: 1.1.0.2, DR's priority: 1
  PIM neighbor count: 0
  PIM hello interval: 30 secs, next hello sent in: 0.000000
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x26bf8eb9
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
  PIM BFD enabled: no
  PIM passive interface: yes
  PIM VPC SVI: no
  PIM Auto Enabled: yes
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 0/0 (early: 0), JPs: 0/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 0/0, DF-Winners: 0/0, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 0
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0
Vlan110, Interface status: protocol-up/link-up/admin-up
  IP address: 188.0.0.2, IP subnet: 188.0.0.0/24
  PIM DR: 188.0.0.2, DR's priority: 1
  PIM neighbor count: 0
  PIM hello interval: 30 secs, next hello sent in: 0.000000
  PIM neighbor holdtime: 105 secs
  PIM configured DR priority: 1
  PIM configured DR delay: 3 secs
  PIM border interface: no
  PIM GenID sent in Hellos: 0x1f81e5a1
  PIM Hello MD5-AH Authentication: disabled
  PIM Neighbor policy: none configured
  PIM Join-Prune inbound policy: none configured
  PIM Join-Prune outbound policy: none configured
  PIM Join-Prune interval: 1 minutes
  PIM Join-Prune next sending: 1 minutes
```

```
  PIM BFD enabled: no
  PIM passive interface: yes
  PIM VPC SVI: no
  PIM Auto Enabled: yes
  PIM Interface Statistics, last reset: never
    General (sent/received):
      Hellos: 14264/13767 (early: 0), JPs: 0/0, Asserts: 0/0
      Grafts: 0/0, Graft-Acks: 0/0
      DF-Offers: 0/0, DF-Winners: 13558/0, DF-Backoffs: 0/0, DF-Passes: 0/0
    Errors:
      Checksum errors: 0, Invalid packet types/DF subtypes: 0/0
      Authentication failed: 0
      Packet length errors: 0, Bad version packets: 0, Packets from self: 0
      Packets from non-neighbors: 0
          Packets received on passiveinterface: 652
      JPs received on RPF-interface: 0
      (*,G) Joins received with no/wrong RP: 0/0
      (*,G)/(S,G) JPs received for SSM/Bidir groups: 0/0
      JPs filtered by inbound policy: 0
      JPs filtered by outbound policy: 0

For both leafs attaching to source and receiver, make sure both fabric and host facing SVIs are DR,
  PIM passive, and PIM Auto Enabled.
```

Step 3 Verify that both the host and fabric SVIs are DF Winners.

Example:

```
source_leaf # show ip pim df vrf vpn1
Bidir-PIM Designated Forwarder Information for VRF "vpn1"
RP Address (ordinal)   RP Metric        Group Range
18.18.18.18 (2)        [200/0]          238.0.0.0/16
  Interface            DF Address       DF State   DF Metric    DF Uptime
  Vlan10               1.1.0.1          Winner     [0/0]        4d23h     (RPF)
  Vlan110              199.0.0.2        Winner     [200/0]      4d23h

Receiver_leaf #  show ip pim df vrf vpn1
Bidir-PIM Designated Forwarder Information for VRF "vpn1"
RP Address (ordinal)   RP Metric        Group Range
18.18.18.18 (2)        [200/0]          238.0.0.0/16
  Interface            DF Address       DF State   DF Metric    DF Uptime
  Vlan10               1.1.0.2          Winner     [0/0]        4d23h     (RPF)
  Vlan110              188.0.0.2        Winner     [200/0]      4d23h
```

Verifying That Unique IP Address Per Leaf is Configured

The profile configures a unique IP address per leaf on each VRF and advertises it via MP-BGP to all the leaf nodes. For troubleshooting, any host can ping any leaf in that VRF and vice-versa.

The following is an example for vrf-common-loopback-universal configuration profile:

```
configure profile vrf-common-loopback-universal interface loopback
interface loopback $system_auto_loopbackId
vrf member $vrfName
  ip address $system_auto_backboneIpAddress/32 tag 12345
vrf context $vrfName
vni $include_vrfSegmentId
rd auto
ip route 0.0.0.0/0 $include_serviceNodeIpAddress
address-family ipv4 unicast
  route-target both auto
address-family ipv6 unicast
  route-target both auto
router bgp $asn
vrf $vrfName
```

```
address-family ipv4 unicast
 redistribute hmm route-map FABRIC-RMAP-REDIST-HOST
 redistribute direct route-map FABRIC-RMAP-REDIST-SUBNET
 maximum-paths ibgp 2
address-family ipv6 unicast
 redistribute hmm route-map FABRIC-RMAP-REDIST-V6HOST
 redistribute direct route-map FABRIC-RMAP-REDIST-SUBNET
 maximum-paths ibgp 2
```