

**Contrail™**

---

# Contrail Fabric Lifecycle Management and Bare Metal Servers Guide

Published  
2020-11-20

Release  
5.1

Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

*Contrail™ Contrail Fabric Lifecycle Management and Bare Metal Servers Guide*  
5.1

Copyright © 2020 Juniper Networks, Inc. All rights reserved.

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

## YEAR 2000 NOTICE

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

## END USER LICENSE AGREEMENT

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

# Table of Contents

## About the Documentation | vi

Documentation and Release Notes | vi

Documentation Conventions | vi

Documentation Feedback | ix

Requesting Technical Support | ix

Self-Help Online Tools and Resources | x

Creating a Service Request with JTAC | x

1

## Overview

### Understanding Underlay Management | 12

Benefits of Underlay Management | 12

### Fabric Lifecycle Management | 13

2

## Intent Driven Automation

### Support for Intent Driven Automation Functionality using Ansible | 15

### Providing Intent Driven Automation Capabilities on Physical Network Elements | 16

Image Management | 16

Upload a New Device Image | 17

Upgrade an Existing Device Image | 19

Fabric Management | 20

Create a Fabric | 21

Delete a Fabric | 26

Discover a Device | 27

Assign Role to a Device | 27

Manage Device Configuration | 28

View Node Profile Information | 29

### Provisioning Fabric Devices Using End-to-End ZTP | 30

## Managing Data Center Devices

### Data Center Interconnect | 52

Understanding Data Center Interconnect | 52

Data Center Interconnect Deployment Topologies | 53

DCI using EBGp | 53

DCI using IBGP | 54

Creating Data Center Interconnect | 54

Create a Fabric | 55

Create Virtual Network | 59

Create Logical Routers | 61

Create Data Center Interconnect Objects | 63

### Configuring QFX10000 as a Data Center Gateway | 65

Discover a Fabric | 66

Add Bare Metal Server | 69

Create Tenant Virtual Network | 70

Add TSN Nodes | 71

Enable VXLAN Routing | 72

Create Logical Router | 73

Verification | 74

### Edge-Routed Bridging for QFX Series Switches | 75

Benefits of ERB | 76

### Hitless Software Upgrade of Data Center Devices Overview | 77

Benefits of Hitless Software Upgrade | 78

### Performing Hitless Software Upgrade on Data Center Devices | 79

### Creating Layer 3 PNF Service Chains for Inter-LR Traffic | 88

Create a Fabric | 88

Create PNF Service Template | 92

Create PNF Service Instance | 94

View Service Appliance Sets and Service Appliances | 95

**Running Generic Device Operations Commands In Contrail Command | 97**

**Certificate Lifecycle Management Using Red Hat Identity Management | 101**

Fully Qualified Domain Names | 102

Performing Lifecycle Management of Certificates using Identity Management | 102

**Virtual Port Groups | 106**

**Configuring Virtual Port Groups | 107**

**Supported Hardware Platforms and Associated Roles | 109**

## 4

**Extending Contrail to Bare Metal Servers**

**Bare Metal Server Management | 115**

Understanding Bare Metal Server Management | 115

Features of the Bare Metal Server Management Framework | 117

**How Bare Metal Server Management Works | 119**

Administrative Workflow | 119

Tenant Workflow | 122

**LAG and Multihoming Support | 123**

**Adding Bare Metal Server to Inventory | 125**

**Launching a Bare Metal Server | 127**

**Onboarding and Discovery of Bare Metal Servers | 128**

Onboarding of Bare Metal Servers | 129

Discovery of Bare Metal Servers | 129

Manual Discovery | 129

Auto Discovery | 129

**Launching and Deleting a Greenfield Bare Metal Server | 130**

**Troubleshooting Bare Metal Servers | 131**

# About the Documentation

## IN THIS SECTION

- Documentation and Release Notes | vi
- Documentation Conventions | vi
- Documentation Feedback | ix
- Requesting Technical Support | ix

Use this guide to understand Contrail underlay management and intent driven automation. This guide also provides information on how to manage data center devices and to extend Contrail to bare metal servers.

## Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at <https://www.juniper.net/documentation/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <https://www.juniper.net/books>.

## Documentation Conventions

[Table 1 on page vii](#) defines notice icons used in this guide.

Table 1: Notice Icons







Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page vii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
<b>Bold text like this</b>	Represents text that you type.	To enter configuration mode, type the <b>configure</b> command:  user@host> <b>configure</b>
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> <b>show chassis alarms</b>  No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> <li>Introduces or emphasizes important new terms.</li> <li>Identifies guide names.</li> <li>Identifies RFC and Internet draft titles.</li> </ul>	<ul style="list-style-type: none"> <li>A policy <i>term</i> is a named structure that defines match conditions and actions.</li> <li><i>Junos OS CLI User Guide</i></li> <li>RFC 1997, <i>BGP Communities Attribute</i></li> </ul>

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name:  [edit] root@# <b>set system domain-name</b> <i>domain-name</i>
<b>Text like this</b>	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> <li>To configure a stub area, include the <b>stub</b> statement at the [edit <b>protocols ospf area area-id</b>] hierarchy level.</li> <li>The console port is labeled <b>CONSOLE</b>.</li> </ul>
< > (angle brackets)	Encloses optional keywords or variables.	<b>stub</b> <default-metric <i>metric</i> >;
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	<b>broadcast   multicast</b>  ( <i>string1</i>   <i>string2</i>   <i>string3</i> )
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<b>rsvp { # Required for dynamic MPLS only</b>
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	<b>community name members [ <i>community-ids</i> ]</b>
Indentation and braces ( { } )	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	

## GUI Conventions



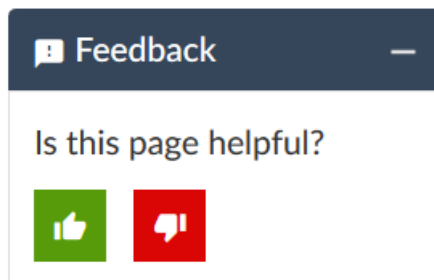
Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
<b>Bold text like this</b>	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"> <li>In the Logical Interfaces box, select <b>All Interfaces</b>.</li> <li>To cancel the configuration, click <b>Cancel</b>.</li> </ul>
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select <b>Protocols&gt;Ospf</b> .

## Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the [Juniper Networks TechLibrary](#) site, and do one of the following:



- Click the thumbs-up icon if the information on the page was helpful to you.
- Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto:techpubs-comments@juniper.net). Include the document or topic name, URL or page number, and software version (if applicable).

## Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are

covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <https://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

## Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <https://www.juniper.net/customers/support/>
- Search for known bugs: <https://prsearch.juniper.net/>
- Find product documentation: <https://www.juniper.net/documentation/>
- Find solutions and answer questions using our Knowledge Base: <https://kb.juniper.net/>
- Download the latest versions of software and review release notes: <https://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <https://www.juniper.net/company/communities/>
- Create a service request online: <https://myjuniper.juniper.net>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://entitlementsearch.juniper.net/entitlementsearch/>

## Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit <https://myjuniper.juniper.net>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <https://support.juniper.net/support/requesting-support/>.

# 1

CHAPTER

## Overview

---

Understanding Underlay Management | 12

Fabric Lifecycle Management | 13

---

# Understanding Underlay Management

An underlay network is the physical infrastructure on which an overlay network is built. An overlay network is built on top of and is supported by an underlay network's physical infrastructure. Starting with release 5.0.1, Contrail supports underlay network management. The existing Contrail configuration node can provide intent driven automation capabilities on physical network elements such as ToR and EoR switches, Spines, SDN gateway, and VPN gateways in the data center. In addition, you can perform basic device management functions such as image upgrade, device discovery, device underlay configuration, assigning roles to devices, and viewing node profile information from the node.

## Benefits of Underlay Management

- Enables basic device management functions from the Contrail configuration node.
- Enables underlay network automation.
- Supports zero-touch-provisioning (ZTP) of factory-default devices to form an IP Clos network.

**NOTE:** ZTP allows you to provision new devices in your network automatically, with minimal manual intervention.

### RELATED DOCUMENTATION

[Support for Intent Driven Automation Functionality using Ansible | 15](#)

[Providing Intent Driven Automation Capabilities on Physical Network Elements | 16](#)

# Fabric Lifecycle Management

You can onboard, configure, and manage a set of devices, bare metal servers (BMS), and physical network functions (PNF) in Contrail as a fabric. A fabric is a set of devices, BMS, and PNFs that fall under the same data center administrator responsibility area. The fabric is linked to different role-based access control (RBAC) profiles for ease of administration and management.

Contrail helps you provision both greenfield and brownfield devices to form IP Clos networks. You can bring up all factory-default greenfield devices using zero-touch-provisioning to form an operational IP Clos network with underlay connectivity. However, unlike greenfield devices, brownfield devices are manually provisioned before device onboarding.

## RELATED DOCUMENTATION

---

[Understanding Underlay Management | 12](#)

---

[Understanding Bare Metal Server Management | 115](#)

---

[Configuring QFX10000 as a Data Center Gateway | 65](#)

# 2

CHAPTER

## Intent Driven Automation

---

Support for Intent Driven Automation Functionality using Ansible | 15

Providing Intent Driven Automation Capabilities on Physical Network Elements | 16

Provisioning Fabric Devices Using End-to-End ZTP | 30

---

# Support for Intent Driven Automation Functionality using Ansible

Starting with Release 5.0.1, Contrail supports the intent driven automation functionality using Ansible. Basic device management functions such as image upgrade, device discovery, device underlay configuration, assigning roles to devices, and viewing node profile information are implemented as Ansible playbooks and are triggered by an action URL (**/execute-job**) on the API server. Custom Ansible python modules such as Contrail Ansible Modules and Vendor Specific Ansible Modules are developed for interactions with the intent driven automation functionality. These python modules are also developed to interact with Virtualized Network Services (VNS) data models and User-Visible Entity (UVE) or Object Log operational data models.

The intent driven automation functionality using Ansible provides the following benefits:

- Multi-vendor support
- Extensible configuration management and automation with a set of plugins
- Option to customize underlay and overlay configuration templates

**NOTE:** All configurations are added by using Jinja templates and therefore can be customized.

You can use ansible to:

- onboard a new vendor, new device family, and hardware platform device
- provide vendor-specific, device family-specific, or hardware platform-specific underlay and overlay device configurations
- customize onboarding of an existing device
- customize device routing-bridging roles
- add new generic device operations with Contrail Release 5.1

## RELATED DOCUMENTATION

[Understanding Underlay Management | 12](#)

[Providing Intent Driven Automation Capabilities on Physical Network Elements | 16](#)

# Providing Intent Driven Automation Capabilities on Physical Network Elements

## IN THIS SECTION

- [Image Management | 16](#)
- [Fabric Management | 20](#)

This topic describes how you can extend the existing Contrail configuration node to provide intent driven automation capabilities on physical network elements such as ToR and EoR switches, Spines, SDN gateway, and VPN gateways in the data center.

## Image Management

## IN THIS SECTION

- [Upload a New Device Image | 17](#)
- [Upgrade an Existing Device Image | 19](#)

You can upload a new device image to the Contrail fabric or upgrade an existing device image of any device in the Contrail fabric.



## Upload a New Device Image

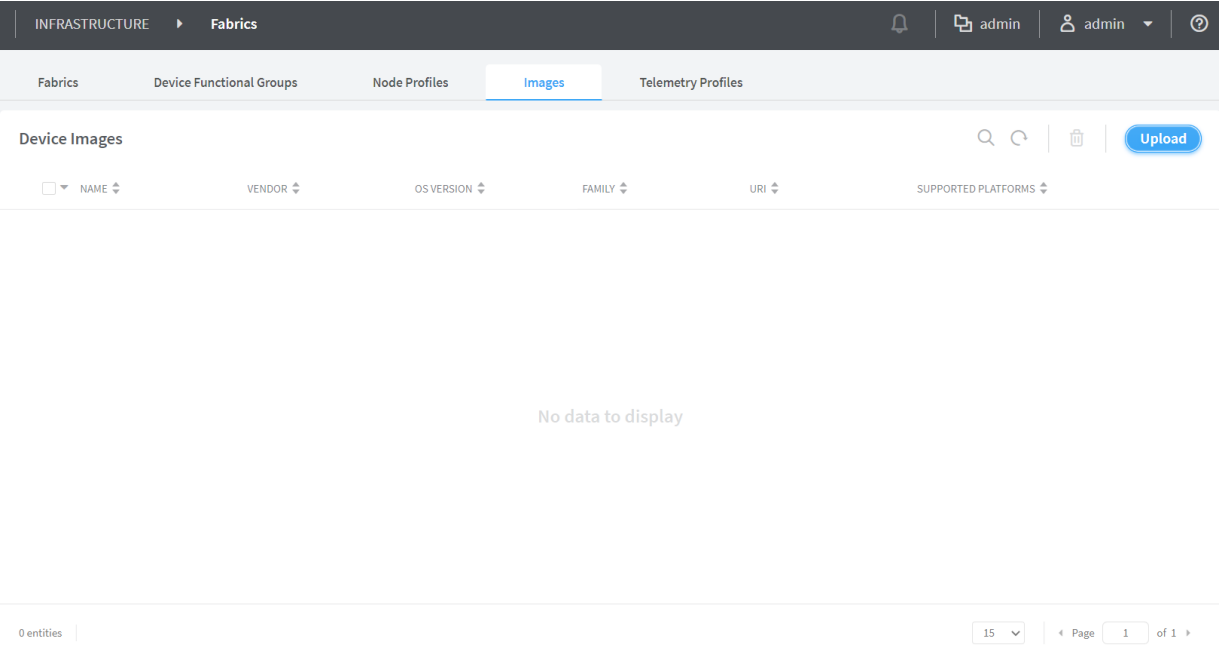
### Upload a New Device Image

Follow these steps to upload a new device image:

- 1. Click **Infrastructure>Fabrics>Images**.

The Device Images page is displayed. See [Figure 1 on page 17](#).

Figure 1: Device Images



- 2. Click **Upload**.

The Upload Image pop-up is displayed. See [Figure 2 on page 18](#).

Figure 2: Upload Image

### Upload Image

Device Image

Tags

Permissions

Name \*

Pick a File \* ?

Drag file here or [browse](#)

Vendor Name \* ?

juniper

Device Family \* ?

Supported Platforms \* ?

Os Version \* ?

Image MD5 ?

Image SHA1 ?

Cancel

Upload

3. Enter the following information given in [Table 3 on page 18](#).

Table 3: Upload Image Details

Field	Action
Name	Enter a name for the device image.
Pick a file	Click <b>Upload</b> and navigate to the local directory and select the device image file.  Click <b>Open</b> to confirm selection.

Table 3: Upload Image Details (*continued*)

Field	Action
Vendor Name	Displays the name of the vendor.
Device Family	Select the device family from the list.
Supported platforms	Select the supported platforms from the list.
OS version	Enter the OS version.
Image MD5	(Optional) Enter MD5 checksum value.
Image SHA1	(Optional) Enter SHA1 checksum value.

- Click **Upload** to begin uploading the device image file.

You are redirected to the Device Images page. When the image upload is complete, the device image is listed in Device Images page.

## Upgrade an Existing Device Image

Follow these steps to upgrade an existing device image:

- Click the **Upgrade** icon.

The Image Upgrade pop-up is displayed.

- From the Compatible Devices pane, select the device you want to upgrade by clicking the device display name.

**NOTE:** You can select more than one device.

The device you select is then moved to the Selected devices to image upgrade pane.

**NOTE:** Devices that are compatible, based on device name and device family, are displayed in the Compatible Devices pane.

- Click **Upgrade** to start device image upgrade.

## Fabric Management

### IN THIS SECTION

- [Create a Fabric | 21](#)
- [Delete a Fabric | 26](#)
- [Discover a Device | 27](#)
- [Assign Role to a Device | 27](#)
- [Manage Device Configuration | 28](#)
- [View Node Profile Information | 29](#)

You can manage a set of devices, bare metal servers (BMS), and physical network functions (PNF) in Contrail as a fabric. A fabric is a set of devices, and BMS and PNFs that fall under the same data center admin responsibility area. The fabric is linked to different role-based access control (RBAC) profiles for ease of administration and management.

You can provision greenfield devices and brownfield devices by using the Contrail Command user interface (UI).

**Greenfield devices**—You can provision new devices to form an IP Clos network. These devices are connected to a management network that is provisioned before device onboarding. The greenfield fabric workflow then zero-touch-provisions all factory-default devices to form an operational IP Clos network with underlay connectivity.

This greenfield fabric workflow includes playbooks that automate the fabric data model creation in the database, DHCP server configuration, generating device bootstrap configuration, uploading device bootstrap configuration to TFTP server, device discovery, node profile auto-assignment, device role assignment, and role-based auto configuration.

**Brownfield devices**—You can provision legacy devices or existing devices to form an IP Clos network. Unlike greenfield devices, brownfield devices are manually provisioned before device onboarding. The brownfield fabric workflow includes playbooks that automate the fabric data model creation in the database. You can perform basic device management functions such as image upgrade, device discovery, device underlay configuration, assign roles to devices, and view node profile information.

You can use the Contrail Command UI to:

### Create a Fabric

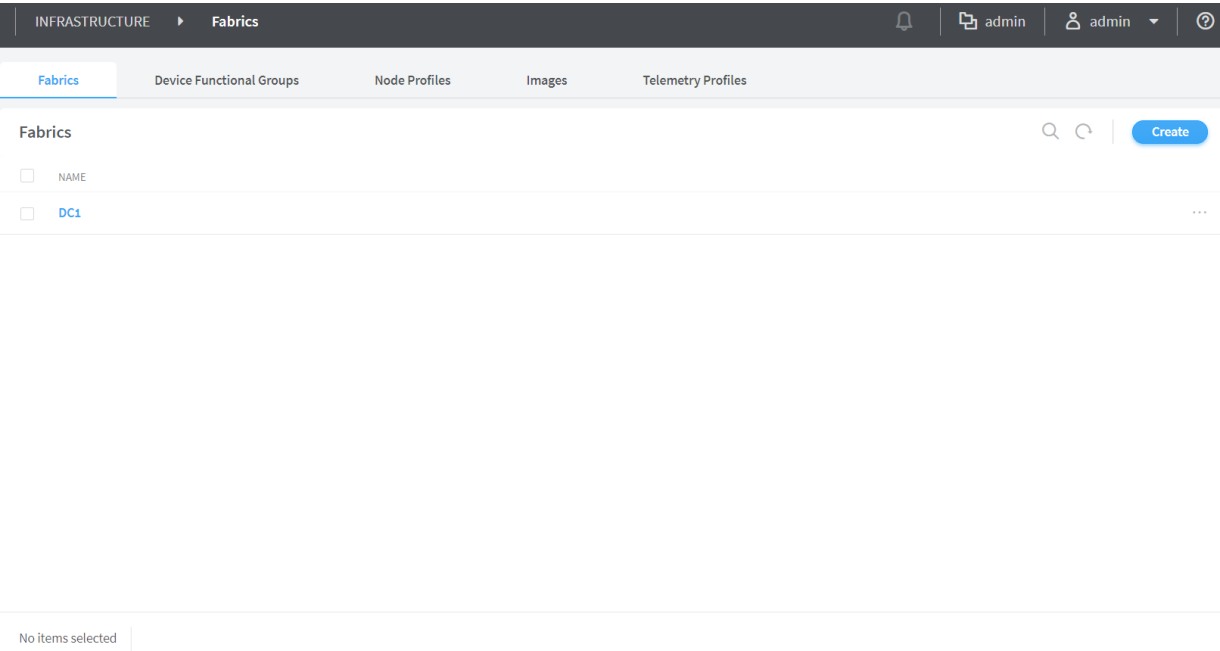
You can use zero-touch-provisioning (ZTP) to create a new fabric by using the Contrail Command UI.

Follow these steps to create a new fabric:

- 1. Click **Infrastructure>Fabrics**.

The Fabrics page is displayed. See [Figure 3 on page 21](#).

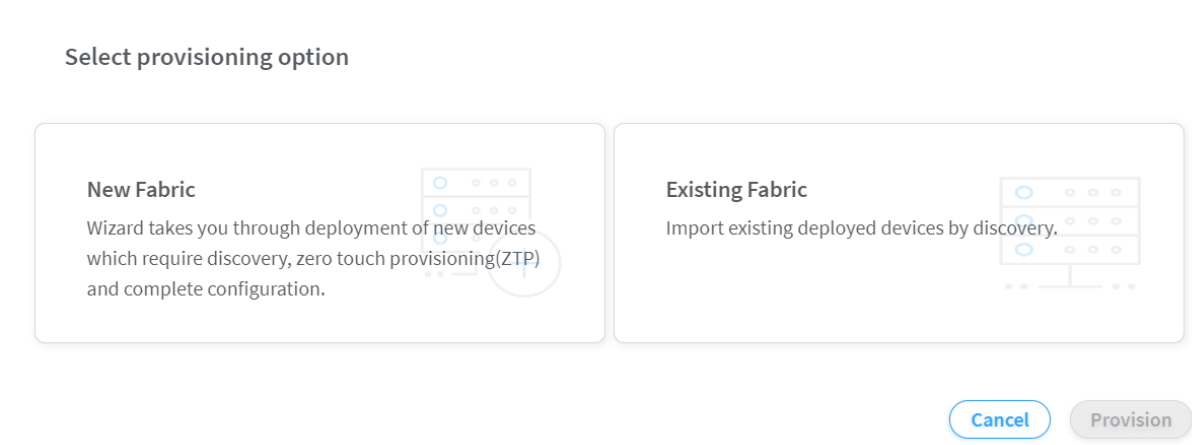
Figure 3: Fabrics Page



- 2. Click **Create**.

You are prompted to select a provisioning option. See [Figure 4 on page 22](#).

Figure 4: Select Provisioning Option



- Click **New Fabric** to deploy new (greenfield) devices. See [Figure 5 on page 24](#).
- Click **Existing Fabric** to import existing (brownfield) devices by discovery. See [Figure 6 on page 26](#).

Click **Provision**.

The Create Fabric page is displayed.

If you select **New Fabric** as the provisioning option, see [“Provisioning Option - New Fabric” on page 22](#).

If you select **Existing Fabric** as the provisioning option, see [“Provisioning Option - Existing Fabric” on page 24](#).

**Provisioning Option - New Fabric**

Enter the information as given in [Table 4 on page 22](#).

Table 4: Provisioning Option - New Fabric

Field	Action
Name	Enter a name for the fabric.
Device credentials	Enter root user password.
Overlay ASN (iBGP)	Enter an autonomous system (AS) number.  The AS number can be in the range from 1 through 65535.
Device Info	Upload device information file.  Navigate to the local directory and select the device information file. Click <b>Open</b> to confirm.

Table 4: Provisioning Option - New Fabric *(continued)*

Field	Action
Minimum devices to be ZTP'ed	Enter the minimum number of devices you want zero-touch-provisioned (ZTP'ed).
Node profiles	<p>Add node profiles.</p> <p>You can add more than one node profile.</p> <p>All preloaded node profiles are added to the fabric by default. You can remove a node profile by clicking <b>X</b> on the node profile. For more information, see <a href="#">“View Node Profile Information” on page 29</a>.</p>
Management subnets	<p>Enter the following information to auto-assign management IP addresses to devices:</p> <p><b>CIDR</b>—Enter CIDR address.</p> <p><b>Gateway</b>—Enter gateway address.</p>
Underlay ASNs (eBGP)	<p>Enter autonomous system (AS) number in the range from 1 through 65535.</p> <ul style="list-style-type: none"> <li>• Enter minimum value in <b>ASN From</b> field.</li> <li>• Enter maximum value in <b>ASN To</b> field.</li> </ul>
Fabric subnets (CIDR)	<p>Enter fabric CIDR address.</p> <p>Fabric subnets are used to assign IP addresses to interfaces that connect to leaf or spine devices.</p>
Loopback subnets (CIDR)	<p>Enter loopback subnet address.</p> <p>Loopback subnets are used to auto-assign loopback IP addresses to the fabric devices.</p>
PNF Servicechain subnets (CIDR)	<p>Enter PNF device CIDR address.</p> <p>Starting in Contrail Release 5.1, enter the subnet for allocating IP addresses in the <b>PNF Servicechain subnets</b> field to establish eBGP session between PNF device and SPINE switch.</p>

Figure 5: Deploy Greenfield Devices

CONTRAIL  
COMMAND

INFRASTRUCTURE ▸ Fabrics ▸ Create Fabric

Default ▸ ctest-TestQos-49173782 admin

Servers

Cluster

Fabrics

Public Cloud

Networks

STEP 1  
Create Fabric

STEP 2  
Device discovery

STEP 3  
Assign the roles

STEP 4  
Autoconfigure

Name\*

Device credentials\*

root user password

Overlay ASN (iBGP)\*

64512

Minimum devices to be ZTP'ed\*

1

Node profiles\*

Management subnets

CIDR\*

Enter valid CIDR

Gateway\*

Enter valid IPv4

Cancel

Next

Click **Next**.

The Discovered devices page is displayed.

**Provisioning Option - Existing Fabric**

Enter the information as given in [Table 5 on page 24](#).

Table 5: Provisioning Option - Existing Fabric

Field	Action
Name	Enter a name for the fabric.
Username	Enter a username for the device.
Password	Enter a password for the device.
Overlay ASN (iBGP)	Enter an autonomous system (AS) number.  The AS number can be in the range from 1 through 65535.



Table 5: Provisioning Option - Existing Fabric (*continued*)

Field	Action
Node profiles	<p>Add node profiles.</p> <p>You can add more than one node profile.</p> <p>All preloaded node profiles are added to the fabric by default. You can remove a node profile by clicking <b>X</b> on the node profile. For more information, see <a href="#">“View Node Profile Information” on page 29</a>.</p>
Management subnets	<p>Enter the following information:</p> <p><b>CIDR</b>—Enter CIDR network address.</p> <p><b>Gateway</b>—Enter gateway address.</p> <p><b>NOTE:</b> You enter the CIDR address range in the <b>Management subnets</b> field to search for devices. Any device that has a previously configured management IP on the subnet is discovered.</p>
Underlay ASNs (eBGP)	<p>Enter autonomous system (AS) number in the range from 1 through 65535.</p> <ul style="list-style-type: none"> <li>• Enter minimum value in <b>ASN From</b> field.</li> <li>• Enter maximum value in <b>ASN To</b> field.</li> </ul>
Fabric subnets (CIDR)	<p>Enter fabric CIDR address.</p> <p>Fabric subnets are used to assign IP addresses to interfaces that connect to leaf or spine devices.</p>
Loopback subnets (CIDR)	<p>Enter loopback address.</p> <p>Loopback subnets are used to auto-assign loopback IP addresses to the fabric devices.</p>
PNF Servicechain subnets (CIDR)	<p>Enter PNF device CIDR address.</p> <p>Starting in Contrail Release 5.1, enter the subnet for allocating IP addresses in the <b>PNF Servicechain subnets</b> field to establish eBGP session between PNF device and SPINE switch.</p>

Figure 6: Import Brownfield Devices

STEP 1 Create Fabric

STEP 2 Device discovery

STEP 3 Assign the roles

STEP 4 Autoconfigure

STEP 5 (optional) Assign Telemetry Profiles

Name \*

Overlay ASN (BGP) \*

64512

Node profiles \*

device-functional-gr... x

juniper-mx x

juniper-qb10k x

juniper-qb10k-lean x

juniper-qb5120 x

juniper-qb5k x

juniper-qb5k-lean x

juniper-ax x

☐ Disable VLAN-VN Uniqueness Check

☒ VLAN-ID Fabric-Wide Significance

Expand All Collapse All

Device credentials

Username \*

Password \*

Cancel Next

Click **Next**.

The Discovered devices page is displayed.

## Delete a Fabric

You can delete a fabric by using the Contrail Command UI. Follow these steps to delete a fabric:

1. Click **Fabrics**.

The Fabrics page is displayed.

2. Select the fabric you want removed by selecting the check box next to the name of fabric.

**NOTE:** Contrail Release 5.0.1 does not support bulk deletion of fabric.

3. Click the **Delete** icon at the end of the row to delete a fabric.

The Delete confirmation pop-up is displayed.

4. Click **Delete** to confirm.

## Discover a Device

Device discovery is initiated as soon as you click **Next** on the Fabrics page. The **Device discovery progress** bar on the Discovered devices page displays the progress of the device discovery job. The list of devices discovered is listed in the Discovered devices page.

You can add a discovered device to the fabric by following these steps:

1. Select the device you want to add by selecting the check box next to the device name.

**NOTE:** You can select more than one device.

2. Click **Add**.

The device is added to the fabric.

Click **Next** to assign roles.

The Assign to devices page is displayed.

## Assign Role to a Device

You can assign roles to the devices from the Assign to devices page.

Follow these steps to assign roles to devices:

1. Select the device you want to assign a role to by selecting the check box next to the device name.
2. Click the **Assign** icon at the end of the row to assign roles.

**NOTE:** To assign roles to more than one device at a time, select the devices by selecting the check box next to the device name and then click **Assign Role**.

The Assign role to devices pop-up is displayed.

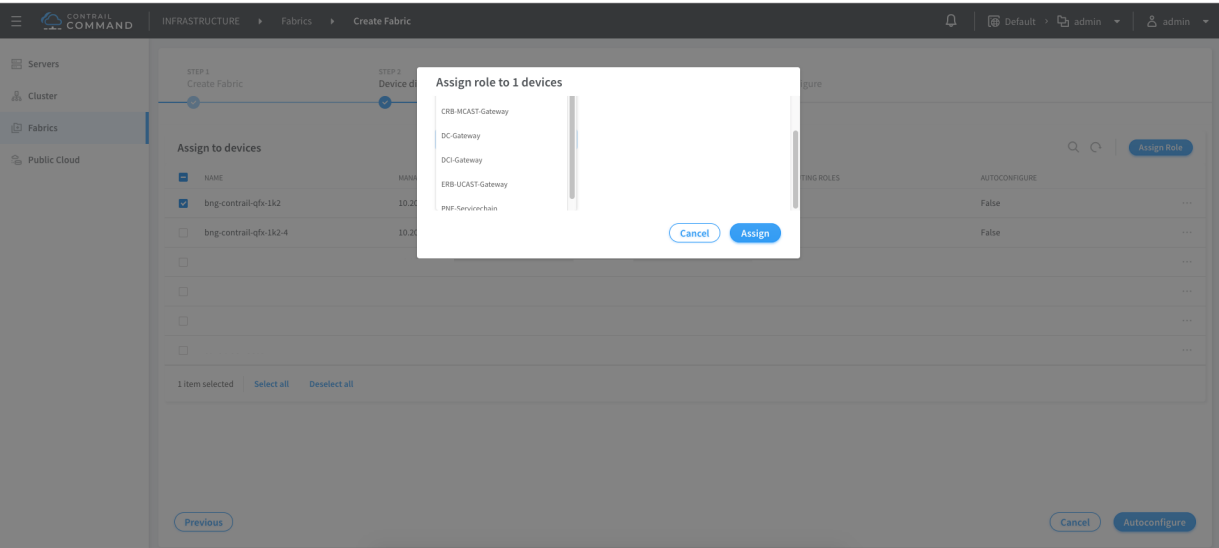
3. Select a physical role type from the **Physical Role** list.

**NOTE:** Contrail Release 5.0.1 supports only leaf and spine physical underlay roles.

- 4. Select a routing bridging role from the **Routing Bridging Role** list.

**NOTE:** Contrail Release 5.0.1 supports CRB-Access, CRB-Gateway, and DC-Gateway overlay roles. Contrail Release 5.1 supports the ERB-UCAST-Gateway and CRB-MCAST-Gateway roles. For more information, see [Centrally-Routed Bridging Overlay Design and Implementation](#).

Figure 7: Assign Roles to Devices



- 5. Click **Assign** to confirm.
- 6. Click **Autoconfigure** to initiate the auto-configuration job.

The Autoconfigure page is displayed.

**Manage Device Configuration**

After you assign device roles, you initiate the auto-configuration job by clicking **Autoconfigure** on the Assign to devices page. The **Autoconfigure progress** bar on the Discovered devices page displays the progress of the auto-configuration job.

Once the auto-configuration job is completed, click **Finish**.

View Node Profile Information

You can view basic device information, vendor information, vendor hardware information, supported routing bridging roles, supported physical roles, assigned devices, and node permission information of a node on the Node Profiles page of the Contrail Command UI.

Follow these steps to view node profiles:

- 1. Click **Infrastructure>Fabrics>Node Profiles**.  
The Node Profiles page is displayed. See [Figure 8 on page 29](#).

Figure 8: Node Profiles

Node Profiles		
NAME	DEVICE FAMILY	VENDOR
▶ <a href="#">juniper-mx</a>	junos	Juniper
▶ <a href="#">juniper-qfx10k</a>	junos-qfx	Juniper
▶ <a href="#">juniper-qfx10k-lean</a>	junos-qfx	Juniper
▶ <a href="#">juniper-qfx5k</a>	junos-qfx	Juniper
▶ <a href="#">juniper-qfx5k-lean</a>	junos-qfx	Juniper
▶ <a href="#">juniper-srx</a>	junos	Juniper

- 2. Select the node profile you want to view by clicking the arrow next to the node profile name.  
The details and permissions of the node profile are displayed.

By default, all preloaded node profiles are available for devices in a fabric.

RELATED DOCUMENTATION

Understanding Underlay Management   12
Support for Intent Driven Automation Functionality using Ansible   15

# Provisioning Fabric Devices Using End-to-End ZTP

From Contrail Networking Release 5.1, you can provision fabric devices using Zero Touch Provisioning (ZTP).

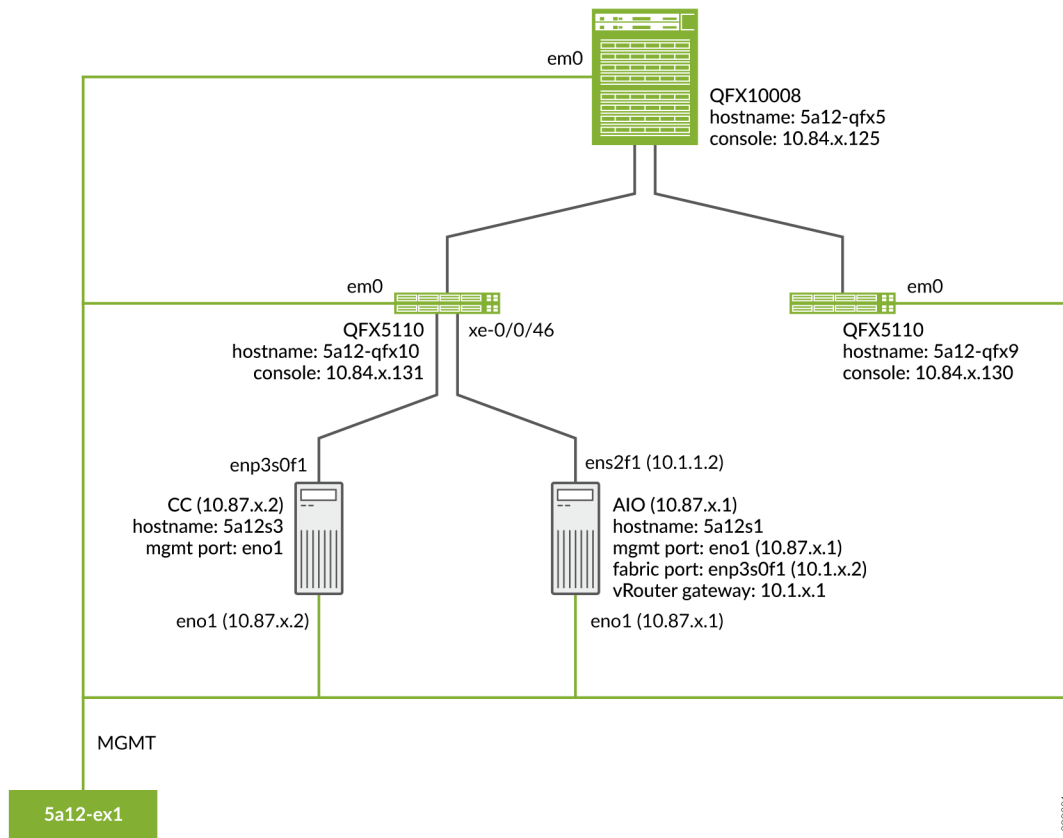
ZTP allows you to provision new Juniper Networks devices in your network automatically, with minimal manual intervention.

This topic provides steps to provision fabric devices using ZTP and configure underlay network via Contrail Command UI.

**NOTE:** You must complete *Installing Contrail Command* before proceeding.

**NOTE:** The minimum required version of Junos OS for QFX5000 and QFX10000 Series devices is 18.1R3-S5 or higher.

## Sample Topology



## Prerequisites

- 5a12s3-node1:
  - Install CentOS 7.6.
  - Configure `eno1` port with the static IP **10.87.x.2/27**.

```
HWADDR=ac:xx:xx:xx:xx:88
NM_CONTROLLED=no
BOOTPROTO=none
DEVICE=enp2s0f0
ONBOOT=yes
IPADDR=10.87.x.2
NETMASK=255.255.255.224
GATEWAY=10.87.6.30
```

- 5a12s1-node1:
  - Install CentOS 7.6.

- Configure *eno1* port with the static IP **10.87.x.1/27**.

```
HWADDR=0c:xx:xx:xx:xx:4a
NM_CONTROLLED=no
BOOTPROTO=none
DEVICE=eno1
ONBOOT=yes
IPADDR=10.87.x.1
NETMASK=255.255.255.224
GATEWAY=10.87.6.30
```

- Configure *ens2f1* port with the static IP **10.1.x.2/24**.

```
HWADDR=90:xx:xx:xx:xx:a1
NM_CONTROLLED=no
BOOTPROTO=none
DEVICE=ens2f1
ONBOOT=yes
IPADDR=10.1.x.2
NETMASK=255.255.255.0
GATEWAY=10.1.x.1
```

**command\_servers.yml example file:**

```
---
command_servers:
  server1:
    ip: 10.87.x.2
    connection: ssh
    ssh_user: root
    ssh_pass: c0ntrail123
    sudo_pass: c0ntrail123
    ntpserver: x.x.x

    # Specify either container_path
    # or registry details and container_name
    container_registry: x.x.x:5010
    container_name: contrail-command
    container_tag: master-720
    config_dir: /etc/contrail

    # contrail command container configurations given here go to
    /etc/contrail/contrail.yml
    contrail_config:
```



```

# Database configuration. MySQL/PostgreSQL supported
database:
    # MySQL example
    host: localhost
    user: root
    password: contrail123
    name: contrail_test
    type: postgres
    dialect: postgres

    # Max Open Connections for DB Server
    max_open_conn: 100
    connection_retries: 10
    retry_period: 3s

# Log Level
log_level: debug

# Server configuration
server:
    enabled: true
    read_timeout: 10
    write_timeout: 5
    log_api: true
    address: ":9091"
    enable_vnc_replication: true

# TLS Configuration
tls:
    enabled: true
    key_file: /usr/share/contrail/ssl/cs-key.pem
    cert_file: /usr/share/contrail/ssl/cs-cert.pem

# Enable GRPC or not
enable_grpc: false

# Static file config
# key: URL path
# value: file path. (absolute path recommended in production)
static_files:
    /: /usr/share/contrail/public

# API Proxy configuration
# key: URL path

```

```

# value: String list of backend host
#proxy:
#    /contrail:
#    - http://localhost:8082

notify_etcd: false

# Keystone configuration
keystone:
  local: true
  assignment:
    type: static
  data:
    domains:
      default: &default
      id: default
      name: default
    projects:
      admin: &admin
      id: admin
      name: admin
      domain: *default
      demo: &demo
      id: demo
      name: demo
      domain: *default
    users:
      admin:
        id: admin
        name: Admin
        domain: *default
        password: contrail123
        email: admin@x.com
        roles:
          - id: admin
            name: Admin
            project: *admin
      bob:
        id: bob
        name: Bob
        domain: *default
        password: bob_password
        email: bob@x.com
        roles:

```

```

        - id: Member
          name: Member
          project: *demo
    store:
      type: memory
      expire: 3600
      insecure: true
      authurl: https://localhost:9091/keystone/v3

    # disable authentication with no_auth true and comment out keystone
    configuration.
    #no_auth: true
    insecure: true

    etcd:
      endpoints:
        - localhost:2379
      username: ""
      password: ""
      path: contrail

    watcher:
      enabled: false
      storage: json

    client:
      id: admin
      password: contrail123
      project_id: admin
      domain_id: default
      schema_root: /
      endpoint: https://localhost:9091

    compilation:
      enabled: false
      # Global configuration
      plugin_directory: 'etc/plugins/'
      number_of_workers: 4
      max_job_queue_len: 5
      msg_queue_lock_time: 30
      msg_index_string: 'MsgIndex'
      read_lock_string: "MsgReadLock"
      master_election: true

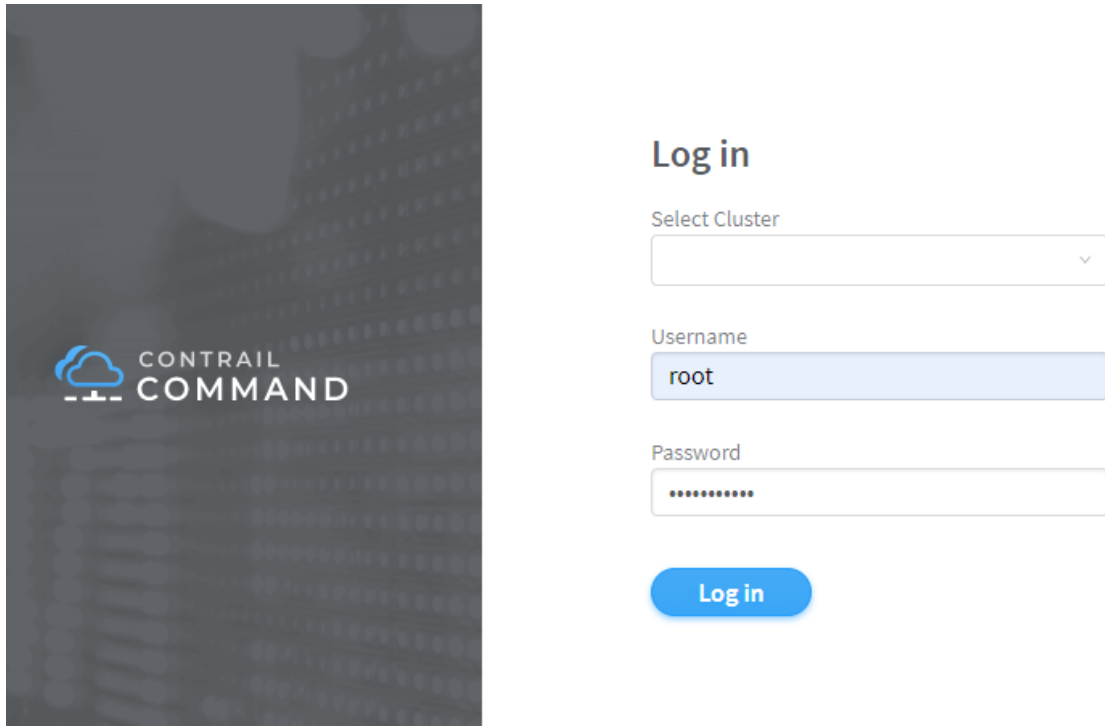
```

```
        # Plugin configuration
        plugin:
            handlers:
                create_handler: 'HandleCreate'
                update_handler: 'HandleUpdate'
                delete_handler: 'HandleDelete'

        agent:
            enabled: true
            backend: file
            watcher: polling
            log_level: debug
    cache:
        enabled: true
        timeout: 10s
        # how long revision deleted event preserved.
        max_history: 100000
    rdbms:
        enabled: true
```

To provision fabric devices using ZTP via Contrail Command UI:

1. Login to Contrail Command UI as a *super user* using credentials **root** for username and **contrail123** for password.

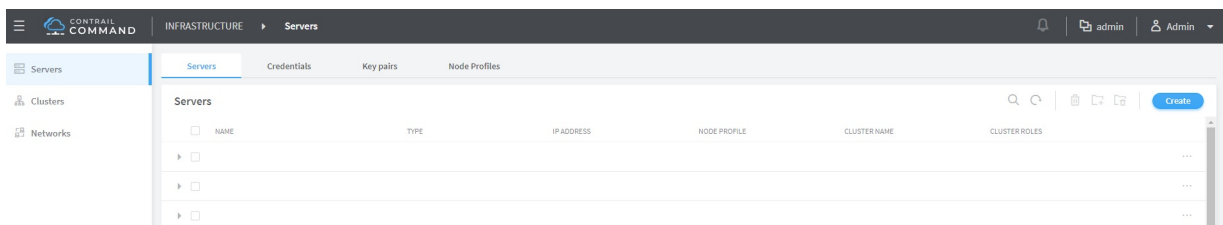


The image shows the Contrail Command UI login page. On the left is a dark sidebar with the Contrail Command logo. The main area is titled 'Log in' and contains a 'Select Cluster' dropdown menu, a 'Username' field with 'root' entered, a 'Password' field with masked characters, and a blue 'Log in' button.

2. Install bootstrap server.

Click **Servers**.

- a. Click **Create**.



- b. Enter the required details.

- c. Click **Create**.

Choose Mode\* ☐ Express ☒ Detailed ☐ Bulk Import (csv)

Select workload type this server will be used for ☒ Physical/Virtual Node ☐ Baremetal

Hostname\*  Management IP\*  Management Interface

Credentials

MAC Address

Disk Partition(s)

Network Interfaces

Name*	IP Address*	
<input type="text" value="eno1"/>	<input type="text" value="x.x.x.x"/>	▼ ▲ 🗑
Name*	IP Address*	
<input type="text" value="ens2f1"/>	<input type="text" value="x.x.x.x"/>	▼ ▲ 🗑

+ Add

- Port *eno1* is connected to management VLAN.
  - Port *ens2f1* is connected to QFX ToR.
3. Create cluster by entering the required details.
    - a. Click **Cluster**.
    - b. Click **Add Cluster**.
    - c. Enter the required details including **Inventory**, **Cloud Manager**, **Infrastructure Networks**, **Overcloud**, etc.

Check **Enable ZTP** checkbox.

- **Default Vrouter Gateway** is the QFX ToR IRB IP. The IP is used for provisioning the network.

*CONTROLLER\_NODES* and *CONTROL\_NODES* are a part of Contrail Configuration.

- *CONTROLLER\_NODES* IP is a static IP configured on port *eno1*.
- *CONTROL\_NODES* IP is a static IP configured on port *ens2f1*.

STEP 1  
Inventory

STEP 2  
Cloud Manager

STEP 3  
Infrastructure Networks

STEP 4 (optional)  
Overcloud

STEP 5 (optional)  
Undercloud Nodes

STEP 6 (optional)  
Jumphost Nodes

STEP 7  
Control Nodes

STEP 8  
Orchestrator Nodes

STEP 9 (optional)  
Compute Nodes

STEP 10 (optional)  
Contrail Service Nodes

STEP 11 (optional)  
Appformix Nodes

Choose Provisioning Manager\*

☐ RHOSP Manager ☒ Contrail Cloud Manager

Cluster Name\*

Container Registry\*

opencontrailnightly

☐ Insecure

Container Registry Username\*

Container Registry Password\*

Contrail Version\*

latest

Provisioner Type

Ansible

Domain Suffix

local

NTP Server

Default Vrouter Gateway

Encapsulation Priority

MPLSoGRE,MPLSoUDP,...

☒ Enable ZTP ⓘ

► Contrail Configuration

STEP 1  
Inventory

STEP 2  
Cloud Manager

STEP 3 (optional)  
Infrastructure Networks

STEP 4 (optional)  
Overcloud

STEP 5 (optional)  
Undercloud Nodes

STEP 6 (optional)  
Jumphost Nodes

STEP 7  
Control Nodes

☐ High availability mode

Available servers

Search servers

Add all

HOSTNAME	IP ADDRESS	DISK PARTITION
Add servers to your inventory		

Assigned Control nodes

Search servers

Remove all

HOSTNAME	IP ADDRESS	DISK PARTITION
5a12s1-node1	10.87.6.1	

Roles\*

contrail\_config\_node ×

contrail\_config\_database\_node ×

contrail\_analytics\_node ×

contrail\_analytics\_alarm\_node ×

contrail\_analytics\_snmp\_node ×

contrail\_analytics\_database\_node ×

**NOTE:** Set **enable\_swift** to **yes** if the cluster will be used for any image management tasks on the fabric devices. Otherwise, set **enable\_swift** to **no**.

- **enable\_ironic** is used for life cycle management of Bare Metal Servers (BMS).
- **enable\_swift** is used to provision Swift containers (object storage). All the images used during different fabric related tasks are stored in these containers.
- **enable\_haproxy** is used when OpenStack controllers are set up in high availability (HA) mode.

CONTRAIL  
COMMAND

SETUP

STEP 1  
Inventory

STEP 2  
Cloud Manager

STEP 3 (optional)  
Infrastructure Networks

STEP 4 (optional)  
Overcloud

STEP 5 (optional)  
Undercloud Nodes

STEP 6 (optional)  
Jumphost Nodes

STEP 7  
Control Nodes

STEP 8  
Orchestrator Nodes

STEP 9 (optional)  
Compute Nodes

STEP 10 (optional)  
Contrail Service Nodes

STEP 11 (optional)  
Appformix Nodes

STEP 12  
Summary

STEP 13  
Provisioning

Orchestrator type\*

Openstack

Show Advanced

Container Registry

default

Openstack Release

queens

Control & Data Network Virtual IP address

Enter valid IPv4

Management Network Virtual IP address

Enter valid IPv4

Customize configuration ⓘ

Place customized configuration...

Kolla Globals

Key	Value	
enable_ironic	no	
Key	Value	
enable_swift	no	
Key	Value	
enable_haproxy	no	

+ Add

Kolla Passwords

+ Add

Available servers

Search servers

Add all

HOSTNAME	IP ADDRESS	DISK PARTITION
Add servers to your inventory		

Previous

Assigned Openstack nodes

Search servers

Remove all

HOSTNAME	IP ADDRESS	DISK PARTITION
5a12s1-node1	10.1.1	

Roles\*

openstack\_control\_node ×

openstack\_network\_node ×

Next



STEP 1  
Inventory

STEP 2  
Cloud Manager

STEP 3 (optional)  
Infrastructure Networks

STEP 4 (optional)  
Overcloud

STEP 5 (optional)  
Undercloud Nodes

STEP 6 (optional)  
Jumphost Nodes

STEP 7  
Control Nodes

STEP 8  
Orchestrator Nodes

STEP 9 (optional)  
Compute Nodes

STEP 10 (optional)  
Contrail Service Nodes

Available servers

Search servers

Add all

<

HOSTNAME	IP ADDRESS	DISK PARTITION
Add servers to your inventory		

Assigned Service nodes

Search servers

Remove all

HOSTNAME	IP ADDRESS	DISK PARTITION
5a12s1-node1	10.1.1.1	
Default Vrouter Gateway*		
10.1.1.1		

Cluster overview

Display name	AIO
Container registry	repo:5010 (insecure)
Contrail version	master-550
Provisioner type	ansible
Domain Suffix	local
NTP server	ntp.juniper.net
Default Vrouter Gateway	
Encapsulation priority	VXLAN,MPLSoUDP,MPLSoGRE
Enable ZTP	true
▸ Contrail configuration	
High availability mode	false
Orchestrator	openstack
Openstack release	queens
Openstack internal virtual IP	-
Openstack external virtual IP	-
Openstack registry	default
▸ Kolla globals	
Kolla passwords	-

Nodes overview

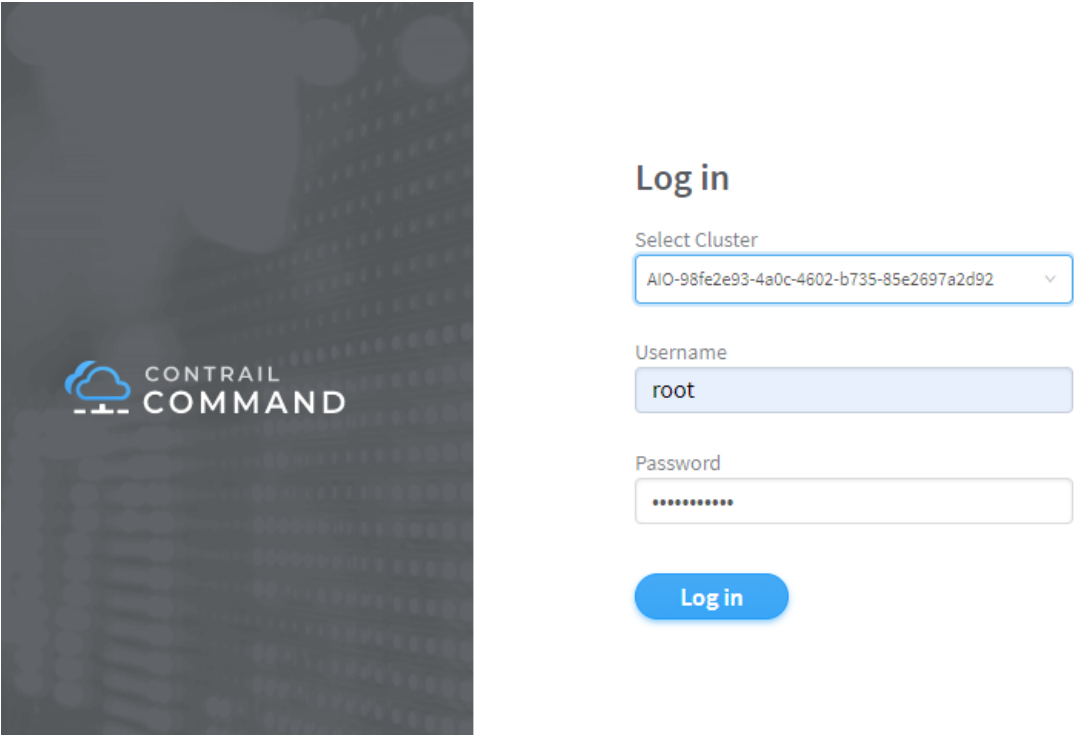


All cluster nodes	Control nodes	Compute nodes	Openstack nodes	Service nodes
NAME	TYPE	IP ADDRESS	NODE PROFILE	ROLES
5c10s7-node4	physical/virtual node			Control node; Compute n...

Previous

Provision

- d. Click **Create**.
4. After creating the cluster, login to the cluster using credentials **root** for username and **cOntrail123** for password.



- 5. Run fabric ZTP workflow to onboard the fabric devices
  - a. Click **Fabrics**.
  - b. Click **Create** .
  - c. Click **New Fabric**.
  - d. Click **Provision**.
  - e. Enter the required details.

Table 6: Required Fields for creating Fabric

Field	Details
Overlay ASN (iBGP)	iBGP ASN pool for Contrail overlay network. List of the ASN pools that can be used to configure the iBGP peers for the IP fabric
Underlay ASNs (eBGP)	eBGP ASN pool for fabric underlay network. List of the ASN pools that can be used to configure the eBGP peers for the IP fabric
Management subnet	List of the management network subnets for the fabric

Table 6: Required Fields for creating Fabric *(continued)*

Field	Details
Fabric subnet	List of subnet prefixes that can be used for the P2P networks between fabric devices
Loopback subnet	List of the subnet prefixes that can be allocated to fabric device loopback IPs

**Sample device\_info.yml file**

```

supplemental_day_0_cfg:
  - name: "cfg1"
    cfg: |
      set system ntp server 167.99.20.98
device_to_ztp:
  - serial_number: "74035760356"
    supplemental_day_0_cfg: "cfg1"
  - serial_number: "55674325815"
    supplemental_day_0_cfg: "cfg1"
  - serial_number: "11675330144"
  - serial_number: "74656088411"

```

STEP 1

Create Fabric

STEP 2

Device discovery

STEP 3

Assign the roles

STEP 4

Autoconfigure

Name \*

Device credentials \*

root user password

Overlay ASN (iBGP) \*

64512

Device Info \*

Download Template: [\(?\) \(\\*.yaml\)](#)

[\(?\) Upload .yaml or .yaml](#)

Node profiles \*

juniper-mx ×

juniper-qfx10k ×

juniper-qfx5k ×

juniper-qfx5k-lean ×

juniper-srx ×

Management subnets

CIDR \*

Enter valid CIDR

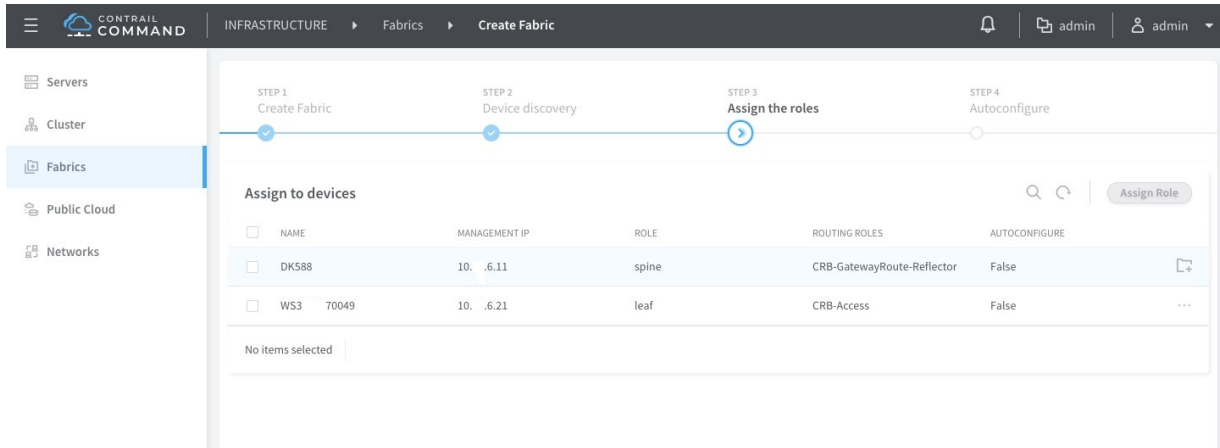
Gateway \*

Enter valid IPv4

Cancel

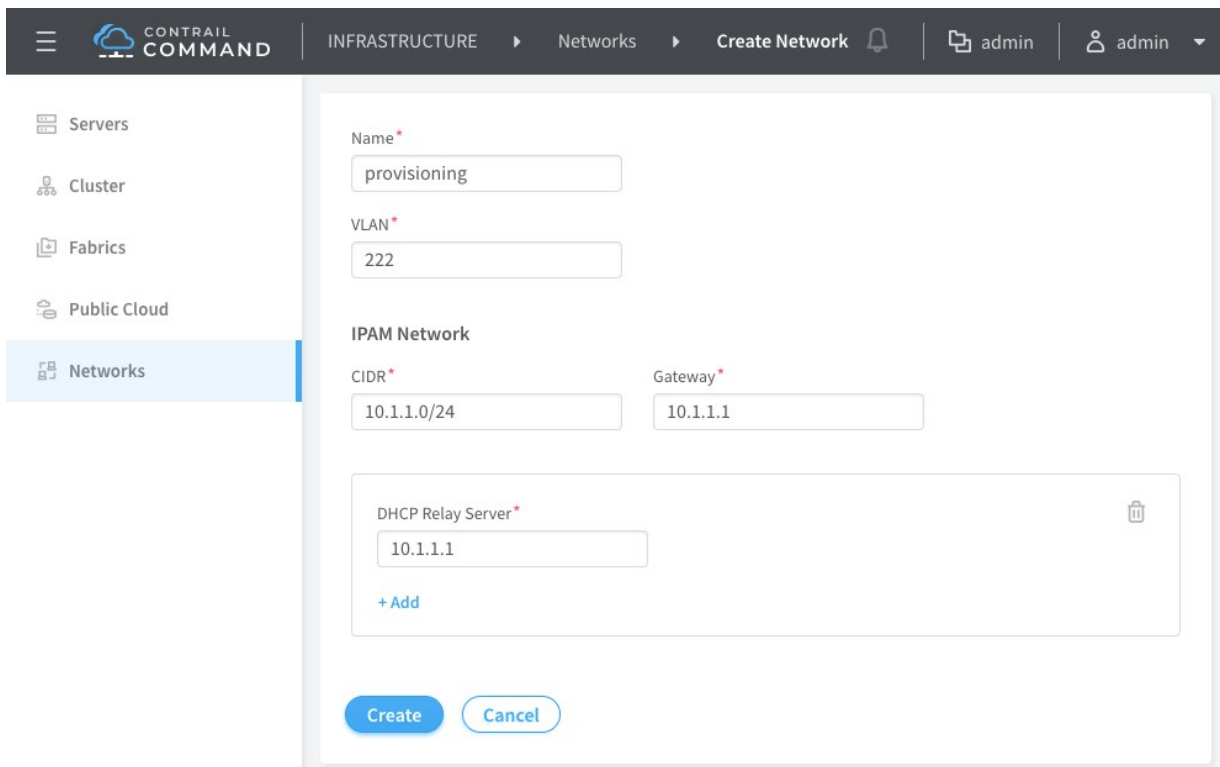
Next

- f. Assign the roles to the fabric devices.
- DK588 as Spine with CRB-Gateway and Route-Reflector roles.
  - WS3XXXX0049 as Leaf with CRB-Access role.



To configure underlay network via Contrail Command UI:

1. Create provisioning infrastructure network.
  - a. Click **Networks**.
  - b. Create a network by entering the required details.



2. Import server topology.
  - a. Click **Servers**.


- b. Click **Import**.
- c. Upload the **server topology** file.

### Import Server

To import a Server, please upload a file (\*.json or \*.yaml) from your computer

Download Template: [📄 \(\\*.json\)](#) [📄 \(\\*.yaml\)](#)

Drag a file here, or [browse](#)

 server\_01.yaml

Cancel

Import

Sample server topology yaml file:

```
nodes:
- name: 5a12s1-node1
  type: baremetal
  ports:
    - name: ens2f1
      mac_address: 90:xx:xx:xx:xx:a1
      switch_name: WS37XXX049
      port_name: xe-0/0/46
      switch_id: 3c:61:04:63:0e:80
```

Table 7: Required Fields for server topology yaml file

Field	Details
name	Name of the infrastructure BMS node

Table 7: Required Fields for server topology yaml file (*continued*)

Field	Details
type	Type of the infrastructure BMS node. It must be "baremetal"
ports	List of the ports of BMS node connected to the TOR switch
name	Name of the BMS port
switch_name	TOR switch name
port_name	TOR port name

### 3. Import server node profile.

You must create server node profile for the Contrail Controller server.

- a. Click **Servers**.
- b. Click **Node Profiles**.
- c. Click **Import**.
- d. Upload the **server node profile** file.

Table 8: Required fields for Server Node Profile

Field	Details
kind	Resource type
name	Name of a resource
fq_name	Fully Qualified name of a resource
parent_type	Node profile parent resource type. It must be "global-system-config"
node_profile_vendor	Node Profile vendor name
node_profile_type	Node profile type. It must be "end-system" for servers
hardware_refs	List of references to the hardware models supported by the node profile



Table 8: Required fields for Server Node Profile (*continued*)

Field	Details
card_refs	List of references to the interface cards

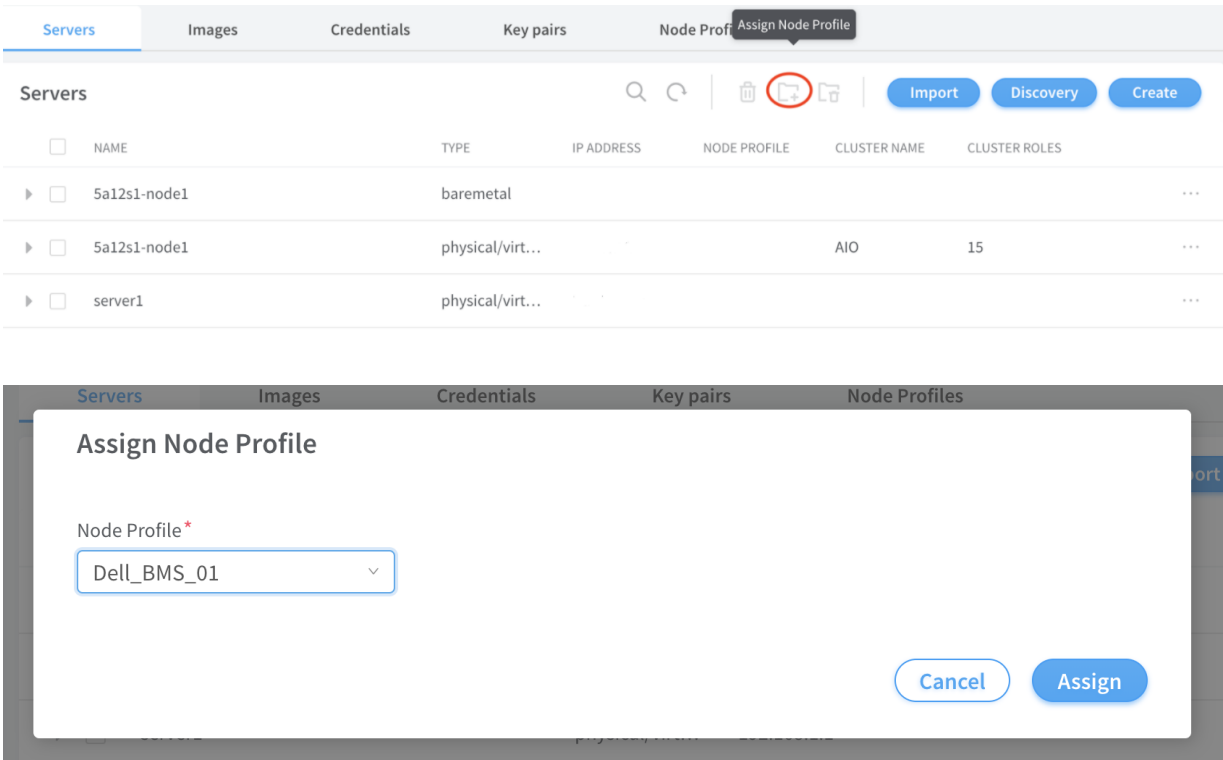
**Sample server node profile json file:**

```
{
  "resources": [
    {
      "kind": "card",
      "data": {
        "name": "dell-bms-card",
        "fq_name": ["dell-bms", "dell-bms-card"],
        "interface_map": {
          "port_info": [{"name": "ens2f1", "labels": ["provisioning"]}]}
      }
    },
    {
      "kind": "hardware",
      "data": {
        "name": "dell-bms",
        "fq_name": ["dell-bms"],
        "card_refs": [{"to": ["dell-bms", "dell-bms-card"]}]}
    },
    {
      "kind": "node_profile",
      "data": {
        "hardware_refs": [{"to": ["dell-bms"]}]}],
        "parent_type": "global-system-config",
        "name": "Dell_BMS_01",
        "fq_name": ["default-global-system-config", "Dell_BMS_01"],
        "node_profile_vendor": "Dell",
        "node_profile_type": "end-system"
      }
    }
  ]
}
```

## 4. Assign node profile to the server.

a. Click **Servers**.

- b. Select the required server from the list.
- c. Click **Assign Node Profile**.



Once the above procedure is completed, change the default route from *management* port to the *access* port.

RELATED DOCUMENTATION

| *Installing Contrail Command*

# 3

CHAPTER

## Managing Data Center Devices

---

Data Center Interconnect | **52**

Configuring QFX10000 as a Data Center Gateway | **65**

Edge-Routed Bridging for QFX Series Switches | **75**

Hitless Software Upgrade of Data Center Devices Overview | **77**

Performing Hitless Software Upgrade on Data Center Devices | **79**

Creating Layer 3 PNF Service Chains for Inter-LR Traffic | **88**

Running Generic Device Operations Commands In Contrail Command | **97**

Certificate Lifecycle Management Using Red Hat Identity Management | **101**

Virtual Port Groups | **106**

Configuring Virtual Port Groups | **107**

Supported Hardware Platforms and Associated Roles | **109**

---

# Data Center Interconnect

## IN THIS SECTION

- [Understanding Data Center Interconnect | 52](#)
- [Data Center Interconnect Deployment Topologies | 53](#)
- [Creating Data Center Interconnect | 54](#)

Starting in Contrail Release 5.1, you can automate data center interconnect (DCI) of two different data centers.

These topics provide information on data center interconnect deployment topologies and how you can create a data center interconnect.

## Understanding Data Center Interconnect

You can automate data center interconnect (DCI) of two different data centers. Multiple tenants connected to a logical router in a data center can exchange routes with tenants connected to a logical router in another data center. All BGP routers in a data center should peer with local route reflectors and not with BGP routers on another fabric. Contrail Release 5.1 supports interconnect of data centers that exist in different fabrics. Contrail Networking defines elements (spine switch and leaf switch) that belong to a data center.

A single Contrail Networking cluster can manage multiple data center pods that are composed of two-tier IP fabric. These data center pods are used to provision overlay layer 2 and layer 3 networking services as virtual networks and logical routers.

Contrail Networking automates the interconnection of logical routers (Layer 3 VRF) in each pod. A DCI object represents the extension of a logical router from one data center pod to another by using EVPN VXLAN Type 5 routes. These logical routers that are extended to the devices in each fabric are assigned DCI-Gateway role. The routing policies are configured on both pods to ensure EVPN type 5 routes are exchanged across the data center. For more information, see [“Creating Data Center Interconnect” on page 54](#).

**NOTE:** The gateway devices must support DCI-Gateway routing bridging role.

## Data Center Interconnect Deployment Topologies

### IN THIS SECTION

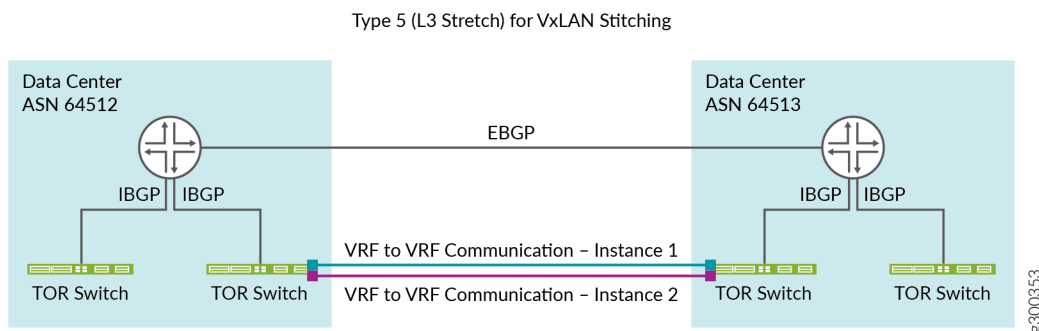
- [DCI using EBGP | 53](#)
- [DCI using IBGP | 54](#)

Contrail Release 5.1 supports “[DCI using EBGP](#)” on page 53 and “[DCI using IBGP](#)” on page 54 data center interconnect deployment topologies.

EVPN Type 5 routes are used in a DCI context to ensure inter-data center traffic between data centers using different IP address subnetting schemes can be exchanged. Routes are exchanged between spine devices in different data centers to allow for the passing of traffic between data centers. Physical connectivity between the data centers is required before EVPN Type 5 messages can be sent between data centers. For more information, see [Data Center Interconnect Design and Implementation](#).

### DCI using EBGP

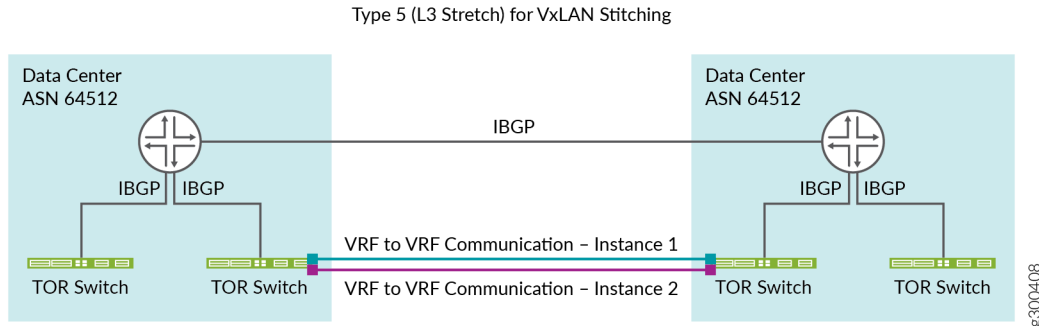
Figure 9: DCI using EBGP Connection



DCI using EBGP connection establishes an EBGP connection between two data centers. The data centers are configured with two different autonomous system (AS) numbers as depicted in [Figure 9](#) on page 53. Contrail Release 5.1 supports interconnect of data centers between two shared VRF instances.

## DCI using IBGP

Figure 10: DCI using IBGP Connection



DCI using IBGP connection establishes an IBGP connection between two data centers. The data centers are configured with the same autonomous system (AS) numbers as depicted in [Figure 10 on page 54](#).

## Creating Data Center Interconnect

### IN THIS SECTION

- [Create a Fabric | 55](#)
- [Create Virtual Network | 59](#)
- [Create Logical Routers | 61](#)
- [Create Data Center Interconnect Objects | 63](#)

These topics provide step-by-step instructions to create data center interconnect.

### Prerequisites

Before you start creating data center interconnect, ensure that:

- Junos OS 18.1 is installed
- Logical routers and client networks are connected
- Logical routers are extended to data center interconnect gateway
- There is a route reflector on each data center

- DCI-Gateway role is assigned to participating DCI gateways
- VxLAN Routing is enabled by editing the project from **IAM>Projects**

To enable VxLAN Routing, click **IAM>Projects** and edit the project in use.

Follow these steps to create a data center interconnect.

## Create a Fabric

Follow these steps to create a fabric with brownfield devices from the Contrail Command user interface (UI):

1. Click **Fabrics**.

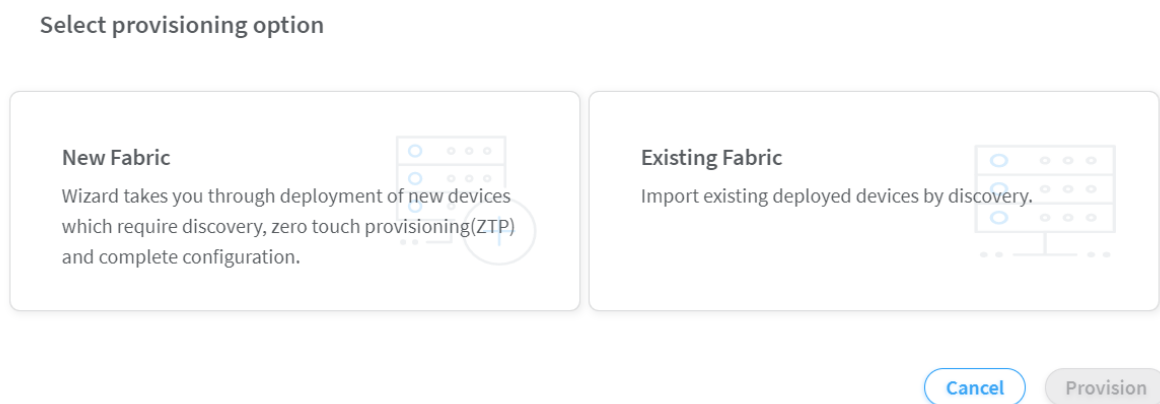
The Fabrics page is displayed.

2. Click **Create**.

You are prompted to select a provisioning option.

3. Click **Existing Fabric** to import existing (brownfield) devices by discovery.

**Figure 11: Select Provisioning Option**



4. Click **Provision**.

The Create Fabric page is displayed.

Figure 12: Create Fabric Page

STEP 1 Create Fabric    STEP 2 Device discovery    STEP 3 Assign the roles    STEP 4 Autoconfigure    STEP 5 (optional) Assign Telemetry Profiles

Name \*

Overlay ASN (iBGP) \*

64512

Node profiles \*

- device-functional-gr...
- juniper-mx
- juniper-qb10k
- juniper-qb10k-lean
- juniper-qb5120
- juniper-qb5k
- juniper-qb5k-lean
- juniper-arx

☐ Disable VLAN-VN Uniqueness Check

☒ VLAN-ID Fabric-Wide Significance

Expand All   Collapse All

Device credentials \*

Username \*   Password \*

Cancel   Next

5. Enter the following information:

Table 9: Provision Existing Fabric

Field	Action
<b>Name</b>	Enter a name for the fabric.
<b>Username</b>	Enter a username for the device.
<b>Password</b>	Enter a password for the device.
<b>Overlay ASN (iBGP)</b>	Enter an autonomous system number (ASN).  The AS number can be in the range from 1 through 65,535.
<b>Node profiles</b>	Add node profiles.  You can add more than one node profile.  All preloaded node profiles are added to the fabric by default. You can remove a node profile by clicking <b>X</b> on the node profile.



Table 9: Provision Existing Fabric (*continued*)

Field	Action
Management subnets	<p>Enter the following information:</p> <p><b>CIDR</b>—Enter CIDR network address.</p> <p><b>Gateway</b>—Enter gateway address.</p> <p><b>NOTE:</b> You enter the CIDR address range in the <b>Management subnets</b> field to search for devices. Any device that has a previously configured management IP on the subnet is discovered.</p>
Underlay ASNs (eBGP)	<p>Enter autonomous system number (ASN) in the range from 1 through 65,535.</p> <ul style="list-style-type: none"> <li>• Enter minimum value in <b>ASN From</b> field.</li> <li>• Enter maximum value in <b>ASN To</b> field.</li> </ul>
Fabric subnets (CIDR)	<p>Enter fabric CIDR address.</p> <p><b>NOTE:</b> Fabric subnets are used to assign IP addresses to interfaces that connect to leaf or spine devices.</p>
Loopback subnets (CIDR)	<p>Enter loopback address.</p> <p><b>NOTE:</b> Loopback subnets are used to auto-assign loopback IP addresses to the fabric devices.</p>

6. Click **Next**.

The Discovered devices page is displayed.

The **Device discovery progress** bar on the Discovered devices page displays the progress of the device discovery job.

Figure 13: Device Discovery Progress Bar

### Device discovery progress



The list of devices discovered are listed in the Discovered devices page.

7. Select the device(s) you want to add to the fabric and then click **Add**.

The device is added to the fabric.

8. Click **Next** to assign roles.

The Assign to devices page is displayed.

9. Click the **Assign** icon at the end of the row to assign roles.

The Assign role to devices pop-up is displayed.

10. Assign physical roles and routing bridging roles.

**For Spine Devices:**

- Select **spine** from the Physical Role list.
- Select **DCI-Gateway** from the Routing Bridging Roles list.

Figure 14: Assign Role to Spine Devices

### Assign role to 1 devices

Physical Role

spine ▼

Routing Bridging Roles

DCI-Gateway × ▼

Cancel

Assign

**For Leaf Devices:**

- Select **leaf** from the Physical Role list.
- Select **DCI-Gateway** from the Routing Bridging Roles list.

Figure 15: Assign Role to Leaf Devices

### Assign role to 1 devices

Physical Role

Routing Bridging Roles

Cancel

Assign

11. Click **Assign** to confirm selection and then click **Autoconfigure** to initiate the auto-configuration job.

The Autoconfigure page is displayed.

### Create Virtual Network

Follow these steps to create a Virtual Network from the Contrail Command user interface (UI).

1. Click **Overlay>Virtual Networks**.

The All Networks page is displayed.

2. Click **Create** to create a network.

The Create Virtual Network page is displayed.

Figure 16: Create Virtual Network Page

OVERLAY ▶ Virtual Networks ▶ Create Virtual Network

Network Tags Permissions

Name\* ⓘ

VN Fabric Type ⓘ

Network Policies ⓘ  
 ▼

Allocation Mode ⓘ  
 ▼

VxLAN Network Identifier ⓘ

Subnets  
[+ Add](#)

3. Enter a name for the network in the **Name** field.
4. Select network policies from the **Network Policies** list. You can select more than one network policy.
5. Select any one of the following preferred allocation mode.
  - Flat subnet only
  - Flat subnet preferred
  - (Default) User defined subnet only
  - User defined subnet preferred

An allocation mode indicates how you choose a subnet. You select **Flat subnet only** or **Flat subnet preferred** allocation mode when the subnet is shared by multiple virtual networks. However, you select **(Default) User defined subnet only** or **User defined subnet preferred** allocation mode when you want to define a subnet range.

6. The VXLAN ID is populated by default and is displayed in the **VxLAN Network Identifier** field.
7. Enter valid IPv4 subnet or mask in the **CIDR** field.
8. Enter valid IPv4 address in the **Gateway** field.
9. Click **Create**.

The All Networks page is displayed. The virtual networks that you created are displayed in this page.

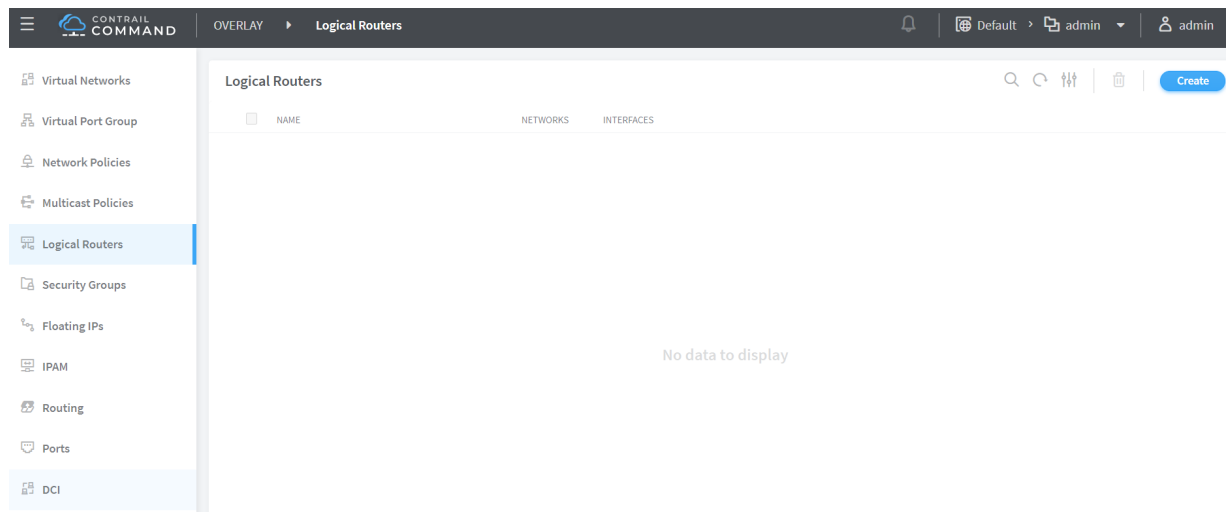
## Create Logical Routers

Follow these steps to create a logical router (LR).

1. Click **Overlay>Logical Routers**.

The Logical Routers page is displayed.

**Figure 17: Logical Routers Page**



2. Click **Create**.

The Create Logical Router page is displayed.

Figure 18: Create Logical Router Page

The screenshot displays the 'Create Logical Router' page in the Contrail Command interface. The sidebar on the left lists various network configuration options, with 'Logical Routers' currently selected. The main panel features three tabs: 'Logical Router', 'Tags', and 'Permissions'. The 'Logical Router' tab is active, showing a form with the following fields and options:

- Name:** A text input field.
- Admin State:** Radio buttons for 'Up' (selected) and 'Down'.
- Extend to Physical Router:** A dropdown menu with 'None' selected.
- Connected networks:** A dropdown menu.
- Public Logical Router:** An unchecked checkbox.
- NAT:** A checked checkbox.
- VxLAN Network Identifier:** A text input field containing '1-16777215'.
- Route Target(s):** A section with a '+Add' button for adding route targets.

At the bottom of the form are 'Create' and 'Cancel' buttons.

## 3. Enter the following information.

Field	Action
<b>Name</b>	Enter a name for the Logical Router.
<b>Admin State</b>	Select <b>Up</b> .
<b>Extend to Physical Router</b>	Select the routers from the list.
<b>Connected Networks</b>	Select the networks from the list.
<b>Public Logical Router</b>	(Optional) Select this check box if you want the logical router to function as a public logical router.
<b>VxLAN Network Identifier</b>	Enter VxLAN network identifier. Range: 1 through 16,777,215
<b>Route Target(s)</b>	Click <b>+Add</b> to add route targets. <ul style="list-style-type: none"> <li>Enter ASN in the ASN field. Range: 1 through 65,535</li> <li>Enter route target in the Target field. Range: 0 through 4,294,967,295</li> </ul>

4. Click **Create** to create the logical router.

The Logical Routers page is displayed.

5. Repeat Step 3 and Step 4 to create another logical router.

## Create Data Center Interconnect Objects

A data center interconnect (DCI) object is a

- collection of data centers
- collection of logical routers (LRs) connected to one or more DCI-Gateways on each data center
- collection of virtual networks connected to logical routers
- DCI-Gateway router

### Creating Data Center Interconnect

Follow these steps to create a DCI of two different data centers from the Contrail Command user interface (UI).

1. Click **Overlay > DCI**.

The DCI page is displayed.

2. Click **Create**.

The Create DCI page is displayed.

Figure 19: Create DCI Page

OVERLAY

Interconnects

Create Data Center Interconnect

DCI name\* ?

DCI Mode ?

L2

L3

Connections ?

Select logical router\*

Fabric

Extend to Physical Router (RB role = DCI-Gateway)\* ?

Select logical router\*

Fabric

Extend to Physical Router (RB role = DCI-Gateway)\* ?

+ Add

Create

Cancel

3. Enter the following information.

Field	Action
DCI name	Enter a name for the DCI.
BGP Hold Time	Modify BGP hold time. <i>This field is optional.</i>
BGP Address Family	Modify the existing BGP address family by selecting BGP address family from the <b>BGP Address Family</b> list. You can select more than one option from the list. <i>This field is optional.</i>



Field	Action
Connections	<p>Follow these steps to connect two logical routers.</p> <p>a. Select logical router from the <b>Select logical router</b> list.</p> <p>b. Select fabric from the <b>Select fabric</b> list.</p> <p>c. Select the physical router you want to extend the connection to from the <b>Extend to Physical Router</b> list.</p> <p>Repeat the above steps to create the next connection.</p>

4. Click **Create**.
- The connections you create are listed in the DCI page.

RELATED DOCUMENTATION

- [Data Center Interconnect Design and Implementation](#)
- [VXLAN Data Center Interconnect Using EVPN Overview](#)

# Configuring QFX10000 as a Data Center Gateway

IN THIS SECTION

- [Discover a Fabric | 66](#)
- [Add Bare Metal Server | 69](#)
- [Create Tenant Virtual Network | 70](#)
- [Add TSN Nodes | 71](#)
- [Enable VXLAN Routing | 72](#)
- [Create Logical Router | 73](#)
- [Verification | 74](#)

Starting in Contrail Release 5.0.1, you can use a QFX10000 switch as a Data Center Gateway (DC-GW). DC-GW is an overlay role that is assigned to a QFX10000 switch to:

- Extend private network
- Extend public routable network

You can extend private network and extend public routable network with EVPN type 5.

These topics provide instructions to configure QFX10000 switch as a DC-GW:

## Discover a Fabric

Follow these steps to discover a fabric by using the Contrail Command user interface (UI):

1. Click **Fabrics**.

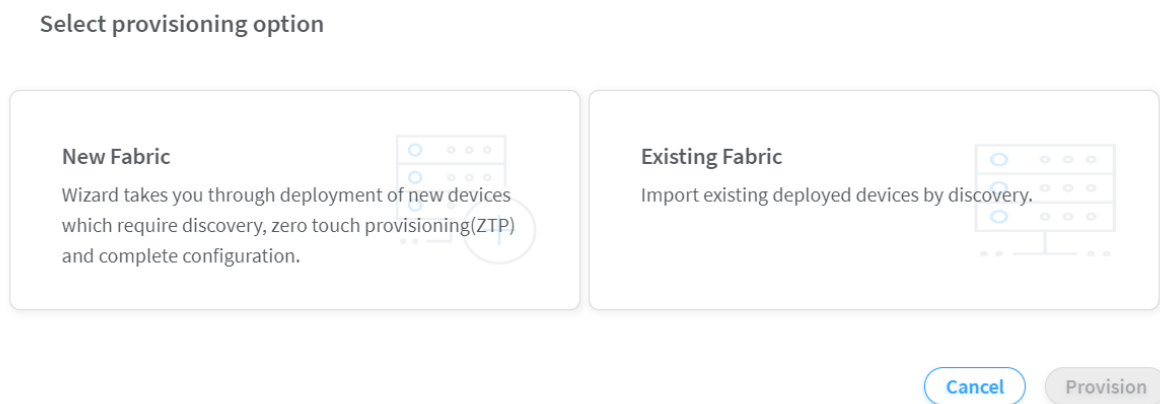
The Fabrics page is displayed.

2. Click **Create**.

You are prompted to select a provisioning option.

3. Click **Existing Fabric** to import existing (brownfield) devices by discovery.

**Figure 20: Select Provisioning Option**



4. Click **Provision**.

The Create Fabric page is displayed.

5. Enter the following information:

Table 10: Provision Existing Fabric

Field	Action
Name	Enter a name for the fabric.
Username	Enter a username for the device.
Password	Enter a password for the device.
Overlay ASN (iBGP)	<p>Enter an autonomous system (AS) number.</p> <p>The AS number can be in the range from 1 through 65,535.</p>
Node profiles	<p>Add node profiles.</p> <p>You can add more than one node profile.</p> <p>All preloaded node profiles are added to the fabric by default.</p> <p>You can remove a node profile by clicking <b>X</b> on the node profile.</p>
Management subnets	<p>Enter the following information:</p> <p><b>CIDR</b>—Enter CIDR network address.</p> <p><b>Gateway</b>—Enter gateway address.</p> <p><b>NOTE:</b> You enter the CIDR address range in the <b>Management subnets</b> field to search for devices. Any device that has a previously configured management IP on the subnet is discovered.</p>
Underlay ASNs (eBGP)	<p>Enter autonomous system (AS) number in the range from 1 through 65,535.</p> <ul style="list-style-type: none"> <li>• Enter minimum value in <b>ASN From</b> field.</li> <li>• Enter maximum value in <b>ASN To</b> field.</li> </ul>
Fabric subnets (CIDR)	<p>Enter fabric CIDR address.</p> <p><b>NOTE:</b> Fabric subnets are used to assign IP addresses to interfaces that connect to leaf or spine devices.</p>
Loopback subnets (CIDR)	<p>Enter loopback address.</p> <p><b>NOTE:</b> Loopback subnets are used to auto-assign loopback IP addresses to the fabric devices.</p>

6. Click **Next**.

The Discovered devices page is displayed. The **Device discovery progress** bar on the Discovered devices page displays the progress of the device discovery job. The list of devices discovered is listed in the Discovered devices page.

7. Select the device you want to add to the fabric and then click **Add**.

The device is added to the fabric.

8. Click **Next** to assign roles.

The Assign to devices page is displayed.

9. Click the **Assign** icon at the end of the row to assign roles.

The Assign role to devices pop-up is displayed.

10. Assign physical roles and routing bridging roles.

- To configure centrally-routed bridging (CRB):

**For Spine Devices:**

- Select **spine** from the Physical Role list.
- Select **CRB-Gateway** from the Routing Bridging Role list.

**For Leaf Devices:**

- Select **leaf** from the Physical Role list.
- Select **CRB-Access** from the Routing Bridging Role list.

- To configure edge-routed bridging (ERB):

**For Spine Devices:**

- Select **spine** from the Physical Role list.
- Select **CRB-MCAST-Gateway** from the Routing Bridging Role list.

**For Leaf Devices:**

- Select **leaf** from the Physical Role list.
- Select **ERB-UCAST-Gateway** from the Routing Bridging Role list.

**NOTE:** Contrail Release 5.0.1 supports CRB-Access, CRB-Gateway, and DC-Gateway overlay roles. Contrail Release 5.1 supports the ERB-UCAST-Gateway and CRB-MCAST-Gateway roles. For more information, see [Centrally-Routed Bridging Overlay Design and Implementation](#).

11. Assign a DC-Gateway Role to the spine device.
    - Select **spine** from the Physical Role list.
    - Select **DC-Gateway** from the Routing Bridging Role list.
  12. Click **Assign** to confirm selection and then click **Autoconfigure** to initiate the auto-configuration job.
- The Autoconfigure page is displayed.

## Add Bare Metal Server

Follow these steps to add an existing bare metal server (BMS) by using the Contrail Command UI:

1. Click **Workloads>Instances**.  
The Instances page is displayed.
2. Click **Create** to create a new instance.  
The Create Instance page is displayed.
3. Select **Existing Baremetal Server** as the Server Type.
4. Enter the following information in the Create Existing Baremetal Server pane:

**Table 11: Add Existing Bare Metal Server Information**

Field	Action
<b>Instance Name</b>	Displays the name of the BMS instance.
<b>Baremetal Node</b>	Select a bare metal node.
<b>Interface</b>	Select an interface from the list.
<b>IP Address</b>	Enter IP address of the instance.

Table 11: Add Existing Bare Metal Server Information (*continued*)

Field	Action
VLAN ID	Enter VLAN ID.
Virtual Network	Select a virtual network from the list.
Select Security Groups	Select <b>default</b> security group from the list.

Figure 21: Existing Bare Metal Server

The screenshot shows the 'Create Instance' page in a cloud management console. The breadcrumb navigation at the top reads 'WORKLOADS > Instances > Create Instance'. The user is logged in as 'admin'. Under 'Server Type', the 'Existing Baremetal Server' option is selected. The 'Create Existing Baremetal Server' section contains the following fields:

- Instance Name\***: bms-qfx3
- Baremetal Node\***: bms-2

Below these is an 'Associate interfaces' section with a trash icon:

- Interface\***: p5p2-00:e0:ed:26:28:ce
- IP Address**: (empty field)
- VLAN ID\***: 0
- Virtual Network\***: vn-public
- Select Security Groups**: default x

There are '+ Add' buttons at the bottom of the interface list and below the entire configuration section. At the bottom of the page are 'Create' and 'Cancel' buttons.

5. Click **Create** to confirm.

## Create Tenant Virtual Network

A virtual network in a EVPN VXLAN data center corresponds to a bridge domain for one tenant in a multi-tenant data center fabric.

Follow these steps to create a tenant virtual network by using the Contrail Command UI:

1. Click **Overlay>Virtual Networks**.

The All networks pane is displayed.

2. Click **Create**.

The Network page is displayed.

3. Enter the following information:

**Table 12: Add Tenant Virtual Network Information**

Field	Action
<b>Name</b>	Enter a name for the virtual network.
<b>Network Policies</b>	Select network policy from the list.
<b>Allocation Mode</b>	Select <b>User defined subnet only</b> as the allocation mode.
<b>VxLAN Network Identifier</b>	Enter VXLAN Network Identifier in the range from 1 through 16,777,215.
<b>Subnets</b>	Click <b>+Add</b> to add subnets.
<b>Network IPAM</b>	Select the default Network IPAM.
<b>CIDR</b>	Enter CIDR address.
<b>Allocation Pools</b>	Enter allocation pool information.
<b>Gateway</b>	Enter gateway IP address.
<b>Auto Gateway</b>	Select Auto Gateway check box.
<b>DHCP</b>	Select DHCP check box.

4. Click **Create** to confirm.

## Add TSN Nodes

Follow these steps to add TSN Nodes to the fabric by using the Contrail Command UI:

Navigate to the EVPN fabric you provisioned.

1. Click the fabric name, and then click the fabric device.

The Fabric Device page is displayed.

2. Enter the following information:

**Table 13: Add TSN Node to Fabric Device Information**

Field	Action
Management IP	Enter management IP address.
VTEP Address	Enter VTEP address.
Loopback IP	Enter loopback IP address.
BGP Router	Select BGP router from the list.
Virtual Router Type	Select virtual router type from the list.
Existing TSN	Select existing TSN from the list.

3. Click **Save** to confirm changes to fabric device.

## Enable VXLAN Routing

Follow these steps to enable VXLAN routing:

1. Click **IAM>Projects**.

The Projects page is displayed.

2. Click the name of the project you want to edit.
3. Enable VXLAN Routing from the Settings pane.



## Create Logical Router

Follow these steps to create a logical router:

1. Click **Overlay>Logical Routers**.

The Logical Routers page is displayed.

2. Click **Create**.

The Create Logical Router page is displayed.

3. Enter the following information:

Table 14: Add Logical Router Information

Field	Action
Name	Enter a name for the logical router.
Admin State	Select <b>Up</b> as the state.
Extend to Physical Router	Select extend to physical router information from the list.
Connected networks	Select connected network information from the list.
NAT	Enable NAT by selecting the NAT check box.
VxLAN Network Identifier	Edit VXLAN network identifier.

**Figure 22: Create Logical Router**

The screenshot shows a web interface for creating a logical router. The breadcrumb navigation at the top reads 'OVERLAY > Logical Routers > Create Logical Router'. On the right, there is a notification bell icon and a user profile icon labeled 'admin'. Below the navigation, there are three tabs: 'Logical Router' (which is selected and underlined in blue), 'Tags', and 'Permissions'. The form contains the following fields and controls:

- Name:** A text input field containing 'LR100'.
- Admin State:** Radio buttons for 'Up' (selected) and 'Down'.
- Extend to Physical Router:** A dropdown menu showing '5b11-qfx5' with a close icon.
- External Gateway:** A dropdown menu showing 'None'.
- Connected networks:** A dropdown menu showing 'vn-public' with a close icon.
- NAT:** A checkbox that is checked.
- VxLAN Network Identifier:** A text input field containing '100'.
- Route Target(s):** A section header with a right-pointing triangle icon.
- Buttons:** 'Create' (solid blue) and 'Cancel' (outlined blue) buttons at the bottom left.

4. Click **Create**.

The Logical Routers page is displayed.

## Verification

EVPN type 5 configuration is pushed to QFX10000 switch as a DC-GW.

Figure 23: EVPN Type 5 Configuration

```

set groups _contrail_overlay_evpn_ interfaces irb gratuitous-arp-reply
set groups _contrail_overlay_evpn_ interfaces irb unit 5 proxy-macip-advertisement
set groups _contrail_overlay_evpn_ interfaces irb unit 5 family inet address 10.87.78.165/28 virtual-gateway-address 10.87.78.161
set groups _contrail_overlay_evpn_ protocols evpn vni-options vni 5 vrf-target target:64512:8000004
set groups _contrail_overlay_evpn_ protocols evpn encapsulation vxlan
set groups _contrail_overlay_evpn_ protocols evpn extended-vni-list all
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail_vn-public-12-5-import term t1 from community target_64512_8000004
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail_vn-public-12-5-import term t1 then accept
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail_vn-public-12-5-export term t1 then community add target_64512_8000004
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail_vn-public-12-5-export term t1 then accept
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-import term t1 from community target_64512_8000005
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-import term t1 then accept
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-export term t1 then community add target_64512_8000005
set groups _contrail_overlay_evpn_ policy-options policy-statement _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-export term t1 then accept
set groups _contrail_overlay_evpn_ policy-options community target_64512_8000004 members target:64512:8000004
set groups _contrail_overlay_evpn_ policy-options community target_64512_8000005 members target:64512:8000005
set groups _contrail_overlay_evpn_ switch-options vtep-source-interface lo0.0
set groups _contrail_overlay_evpn_ switch-options route-distinguisher 5.5.5.3:1
set groups _contrail_overlay_evpn_ switch-options vrf-import _contrail_vn-public-12-5-import
set groups _contrail_overlay_evpn_ switch-options vrf-import _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-import
set groups _contrail_overlay_evpn_ switch-options vrf-export _contrail_vn-public-12-5-export
set groups _contrail_overlay_evpn_ switch-options vrf-export _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-export
set groups _contrail_overlay_evpn_ switch-options vrf-target target:64512:1
set groups _contrail_overlay_evpn_ vlans bd-5 vlan-id 5
set groups _contrail_overlay_evpn_ vlans bd-5 l3-interface irb.5
set groups _contrail_overlay_evpn_ vlans bd-5 vxlan vni 5
set groups _contrail_overlay_evpn_types_ interfaces lo0 unit 1006 family inet address 127.0.0.1/32
set groups _contrail_overlay_evpn_types_ forwarding-options family inet filter input redirect_to_public_vrf_filter
set groups _contrail_overlay_evpn_types_ protocols evpn default-gateway no-gateway-community
set groups _contrail_overlay_evpn_types_ firewall family inet filter redirect_to_public_vrf_filter term term-100 then routing-instance _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6
set groups _contrail_overlay_evpn_types_ firewall family inet filter redirect_to_public_vrf_filter term default-term then accept
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 instance-type vrf
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 interface lo0.1006
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 interface irb.5
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 vrf-import _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-import
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 vrf-export _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6-export
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 routing-options static route 0.0.0.0/0 next-table inet.0
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 protocols evpn ip-prefix-routes advertise direct
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 protocols evpn ip-prefix-routes encapsulation vxlan
set groups _contrail_overlay_evpn_types_ routing-instances _contrail__contrail_lr_internal_vn_23640071-2302-4728-8424-5528df330ae8_-13-6 protocols evpn ip-prefix-routes vni 100

```

## RELATED DOCUMENTATION

[Understanding Underlay Management | 12](#)

[Support for Intent Driven Automation Functionality using Ansible | 15](#)

[Providing Intent Driven Automation Capabilities on Physical Network Elements | 16](#)

[Edge-Routed Bridging for QFX Series Switches | 75](#)

# Edge-Routed Bridging for QFX Series Switches

Starting with Contrail Release 5.1, the edge-routed bridging (ERB) for QFX series switches feature configures the inter-VN unicast traffic routing to occur at the leaf (ToR) switches in an IP CLOS with underlay connectivity topology. The ERB feature introduces the **ERB-UCAST-Gateway** and **CRB-MCAST-Gateway** roles in release 5.1. ERB is supported on the following devices running Junos OS release 18.1R3 and later:

- QFX5110-48S
- QFX5110-32Q
- QFX10002-36Q
- QFX10002-72Q

- QFX10008
- QFX10016

Contrail supports assigning physical roles and routing bridging (overlay) roles to a networking device like a switch. The roles define the routing and bridging responsibilities of the device in the data center. A device can have one physical role and one or more routing bridging roles. In releases prior to release 5.1, Contrail supports centrally-routed bridging (CRB) roles on data center devices. In CRB, when you configure the logical router to allow traffic to flow between Ethernet virtual network instances, the routing occurs at the spine device. Traffic is routed from the leaf to the spine and back. IRB interfaces are configured in the overlay at each spine device to route traffic between virtual networks. Contrail Release 5.1, supports the **ERB-UCAST-Gateway** role in which the routing occurs at the leaf switch. The IRB interfaces are configured at the leaf switch to enable unicast traffic routing at the leaf switch.

Traffic is routed in lesser hops when routed at the leaf switches. For example, consider two bare metal servers belonging to two separate VNs. Unicast traffic between the VNs are routed at the leaf switch and doesn't need to flow to the spine and back. Traffic is routed through the shortest path.

When you configure the **ERB-UCAST-Gateway** role on the leaf switches, it is recommended that you also configure the **CRB-MCAST-Gateway** role for multicast traffic on the corresponding spine devices. The **CRB-MCAST-Gateway** role is also supported from Contrail Release 5.1. While unicast traffic can be routed at the leaf switches, multicast traffic routing still occurs at the spine devices. The existing **CRB-Gateway** role is capable of routing both unicast and multicast traffic at the spine devices. However, in ERB, if leaf switches route the unicast traffic, configuring the **CRB-Gateway** role on the spine is unnecessary since unicast traffic will never reach the spine device. Instead, you must configure the spine devices with the **CRB-MCAST-Gateway** role to route multicast traffic when required.

## Benefits of ERB

- Traffic is routed through the shortest path.
- When you extend a logical router to a physical router, you can extend the logical router to leaf switches as well. Previously, logical routers could only be extended to the spine devices.

### RELATED DOCUMENTATION

[Edge-Routed Bridging Overlay Design and Implementation](#)

[Providing Intent Driven Automation Capabilities on Physical Network Elements | 16](#)

[Configuring QFX10000 as a Data Center Gateway | 65](#)

# Hitless Software Upgrade of Data Center Devices

## Overview

Contrail Controller supports the automation of basic device management functions such as software image upgrade on the devices in the data center fabric. In Contrail Release 5.1, you can perform Contrail Controller-assisted maintenance activities such as a software image upgrade on a data center fabric device managed by Contrail, with zero packet loss. Hitless software upgrade on the leafs and spines of the data center fabric is supported.

Software image upgrade on a networking device in a data center is a time consuming task and might include rebooting the device. During upgrade, if user traffic is being routed through the device then the packets are lost which adversely affects the data center fabric performance.

In release 5.1, during hitless upgrade, the devices are placed in a new mode called maintenance mode for the duration of the maintenance activity. The following sequence of steps are performed during hitless upgrade.

- **Initial Verification**

- Verifying that traffic can be routed from the selected device to another equally capable device. If no such device is present, then hitless upgrade cannot be performed because there will be traffic loss.
- Verifying that the selected upgrade image is compatible with the devices.
- Performing health checks on devices. Health checks are pre-configured parameters against which the devices are checked. If the health checks for devices fail, then the upgrade process for that device is terminated by default. However, you can change the default setting to not terminate upgrade upon health check failure.

If all the checks in the initial verification are cleared, Contrail Controller places the device in the maintenance mode and performs the software upgrade.

- **Maintenance Mode**

- Before the device or devices are placed in maintenance mode, Contrail Controller captures a snapshot of the existing state of the device. This snapshot is used to verify the operational state of the device when the maintenance activity or software upgrade is completed.
- The traffic flowing through the device is rerouted through another equally capable device and the Controller verifies that there is no traffic flowing through the device.
- The device is then taken offline and placed in the maintenance mode.
- The Controller upgrades the software image to the required version on the device.

- **Final Verification**

- The device is taken out of the maintenance mode and traffic is routed through it again.

- Contrail Controller captures a snapshot of the operational state of the device to verify against the snapshot taken previously.

**NOTE:** For hitless software upgrade to work as per design and for zero packet loss, all devices must be redundantly connected. If any device is not redundantly connected, then you will have connectivity and packet loss when the device reboots.

## Benefits of Hitless Software Upgrade

- Maintenance activities can be performed on devices in a data center without a maintenance window.
- No user traffic is lost during image upgrade on devices in the data center.

### RELATED DOCUMENTATION

| [Performing Hitless Software Upgrade on Data Center Devices](#) | 79

# Performing Hitless Software Upgrade on Data Center Devices

Perform the following steps to upgrade the software image on the devices in a data center fabric with no loss of user traffic.

To perform hitless software upgrade on data center devices.

- 1. Upload the software images to which you want to upgrade your devices.
  - a. Navigate to the **Infrastructure > Fabrics** page in Contrail Command. A list of fabrics is displayed in the **Fabrics** tab.
  - b. Click the **Upload** button in the **Images** tab. The **Upload Image** page appears.
  - c. Enter the required software image details and click **Upload**. [Table 15 on page 80](#) lists all the mandatory parameters that must be entered to upload a software image.

Upload Image

Device Image

Tags

Permissions

Name \*

Pick a File \* ⓘ

Drag file here or [browse](#)

Vendor Name \* ⓘ

juniper

Device Family \* ⓘ

Supported Platforms \* ⓘ

Os Version \* ⓘ

Image MD5 ⓘ

Cancel

Upload

Table 15: Upload Image Fields

Field	Description
-------	-------------

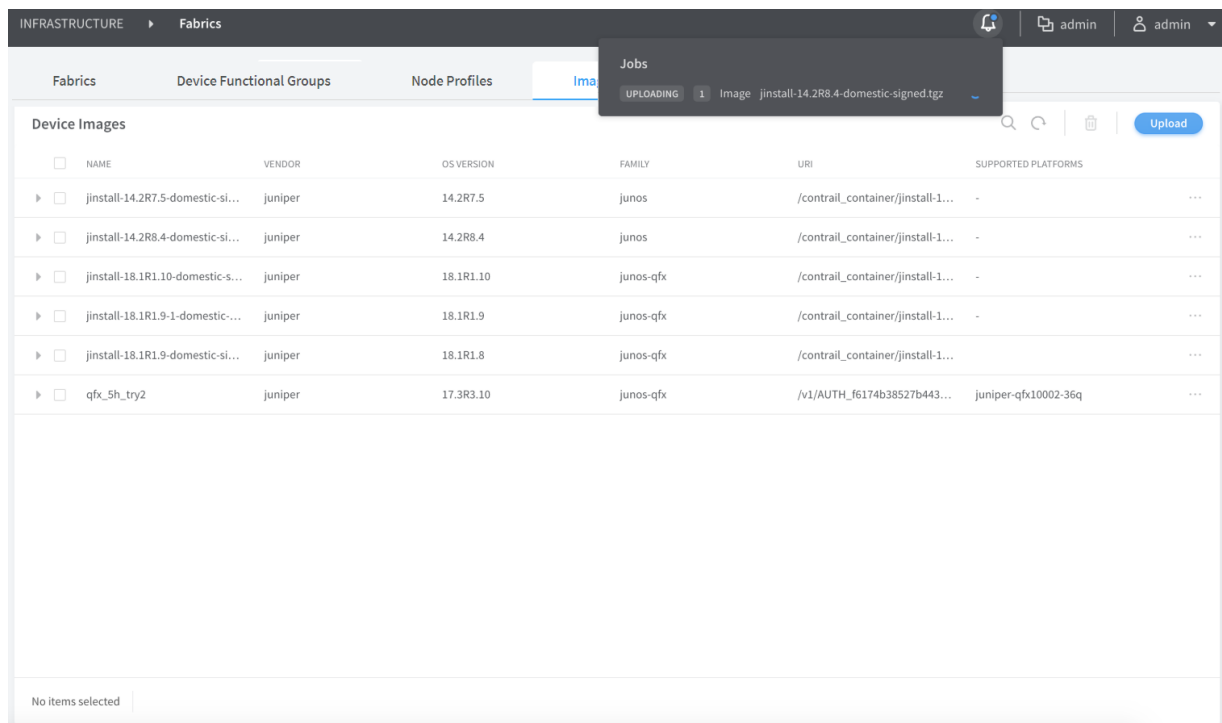


Table 15: Upload Image Fields (*continued*)

Name	Enter a name for the software image. This name cannot be changed once the image has been uploaded.
Pick a File	Select the actual image file to be uploaded.
Vendor name	Enter the image vendor name. For example, Juniper, Arista, and so on.
Device Family	Enter the device family. For example, junos, junos-qfx, and so on.
Supported Platforms	Enter all the device platforms that the image is compatible on.
Os Version	Enter the OS version of the image. For example, 18.1R2.

- d. Upon successful image upload, the **Images** tab appears listing the newly uploaded software image. Apart from the image name, you can edit image details at any time.

The same list of device images is available for image upgrade in [3](#).



2. Click the **Fabrics** tab and select a data center fabric.

The list of devices connected in a spine and leaf topology and corresponding details of each device in the selected fabric is displayed. The roles assigned to the devices are also displayed.

3. Click **Action > Image Upgrade**. The **Select Device** page appears. The list of images available to be upgraded to is displayed.
4. Select the image and the compatible devices to be upgraded to that image in the **Assign Devices** tab.  
You can select one or more devices in the fabric. You can also select multiple images.

Figure 24: Select Device > Assign Devices

INFRASTRUCTURE > Fabrics > fab01 > Upgrade

STEP 1 Select Device (active) | STEP 2 Testing | STEP 3 Upgrade

Assign Devices | Parameters

Images			
NAME	VENDOR	DEVICE FAMILY	DEVICES
▶ jinstall-14.2R7...	juniper	junos-qfx	vqfx1, vqfx2 1 more
▶ jinstall-14.2R8...	juniper	junos-qfx	vqfx5, vqfx3 1 more
▶ jinstall-host-qf...	juniper	junos-qfx	

Devices for image		
DEVICE NAME	VENDOR	DEVICE FAMILY
No data to display		

Cancel Next

5. Select the health check parameters for each device in the **Parameters** tab.

The health check parameters confirm that the devices and the network as a whole are stable to perform hitless image upgrade. By default, if health check fails for a particular device, then image upgrade is terminated. You can deselect the **Abort on health check failure** check box to continue upgrade on a device even if the health check fails.

Figure 25: Select Device &gt; Parameters

INFRASTRUCTURE > Fabrics > fab01 > Upgrade

STEP 1 Select Device | STEP 2 Testing | STEP 3 Upgrade

Assign Devices | **Parameters**

☒ Abort on health check failure

Devices to upgrade simultaneously

4

**BGP**

Flaps allowed for BGP neighbors: 4

Down peers allowed: 0

Check: ☒ Flap count ☒ Down peer count ☒ Peer state

**Alarm**

Check: ☐ System alarm ☒ Chassis alarm

**Interface**

Check: ☒ Error ☒ Drop ☒ Carrier transition

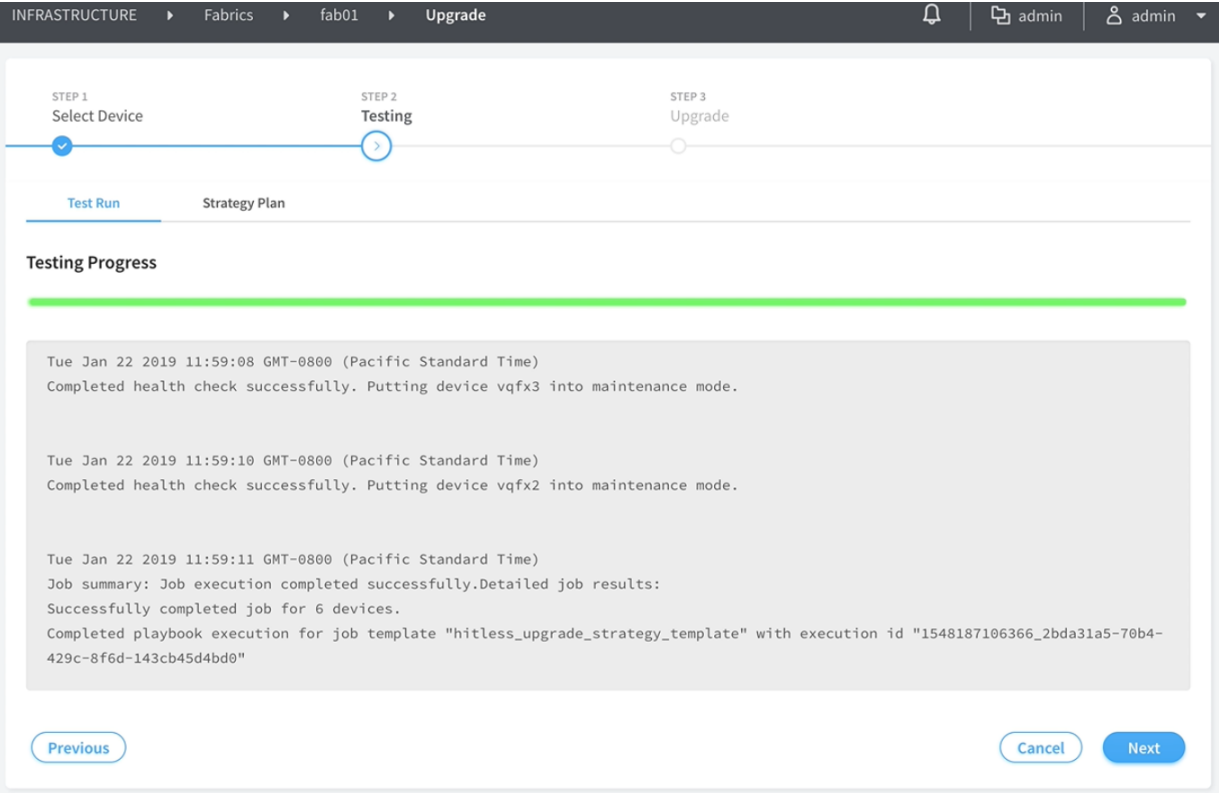
**Routing Engine**

Cancel Next

6. Click **Next**. The **Testing** page appears.

The **Test Run** tab checks that the devices selected for upgrade are not already running the selected software version. The **Test Run** tab also displays the result of the health check on the devices for the parameters selected previously in the **Parameters** tab. If health check fails for the selected parameters, then you can go back to the previous page by clicking **Previous** and either changing the value of the health check parameter or disabling the parameter altogether. You can perform this step multiple times until health check passes for the device or you are able to determine that upgrade on the devices is feasible. Alternatively, you can click **Previous** and deselect the **Abort on health check failure** check box in the **Parameters** tab to continue upgrade on a device even if health check fails.

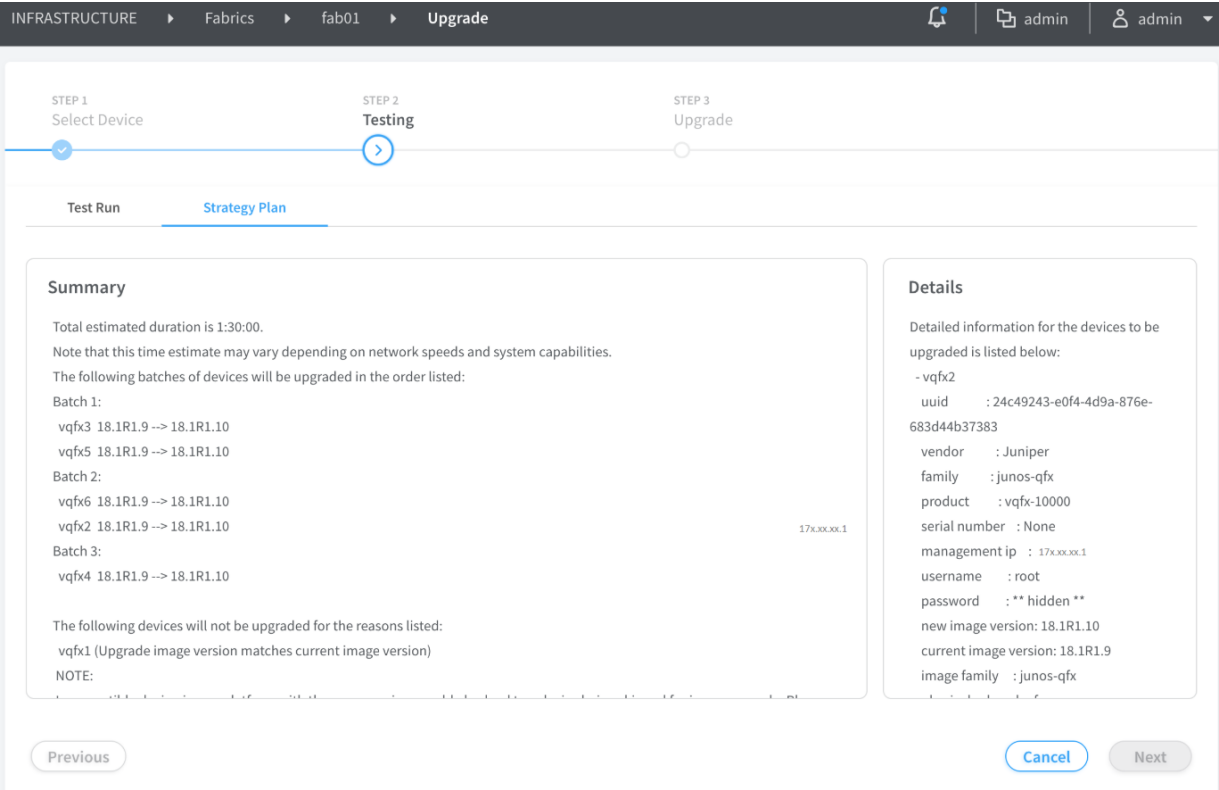
Figure 26: Testing > Test Run



- 7. Click the **Strategy Plan** tab. The **Strategy Plan** tab displays the strategy used to upgrade the images on the selected devices. Image upgrades occurs in batches, where multiple devices are upgraded at one go. The default maximum size of a batch is four devices.

The leafs are upgraded first and in a separate batch from their corresponding spines. If multihoming is configured on a BMS, the corresponding devices are upgraded in different batches. The batches are formed so as to have backup devices in a separate batch to the devices being upgraded in order to make the upgrade hitless. You can view the summary of the strategy used to upgrade the devices at the top and you can scroll down to view complete details of the devices. The estimated time for image upgrade per batch is also displayed.

Figure 27: Testing > Strategy Plan

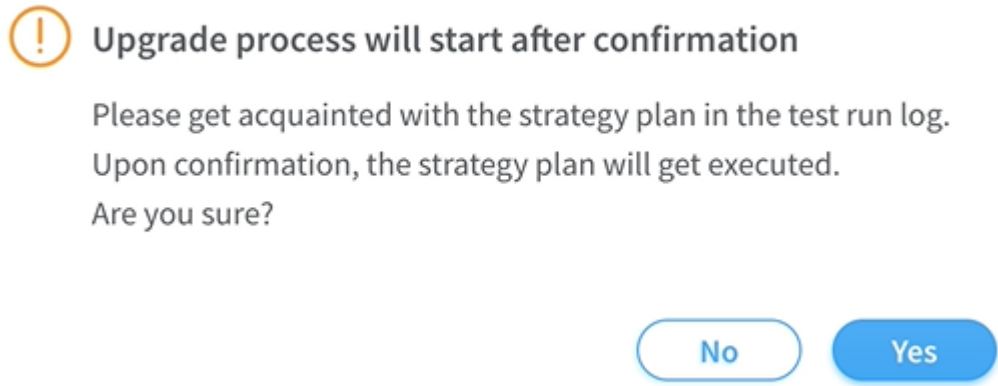


8. Click **Next**. A confirmation page requesting confirmation of the image upgrade process is displayed.

9. Click **Yes** to confirm that you want to continue with the image upgrade. The **Upgrade** page appears displaying the status of the image upgrade progress for each device. The cumulative list of devices is displayed and the upgrade process happens according the batches determined in the strategy plan. The overall progress of all the devices is also displayed.

Alternatively, click **No** to go back to the previous page.

Figure 28: Testing &gt; Strategy Plan Confirmation



10. Click on each device to view the image upgrade progress for that device. Click the device again to toggle back to display the overall image upgrade progress of all devices. [Table 16 on page 86](#) displays the states displayed during the course of the upgrade.

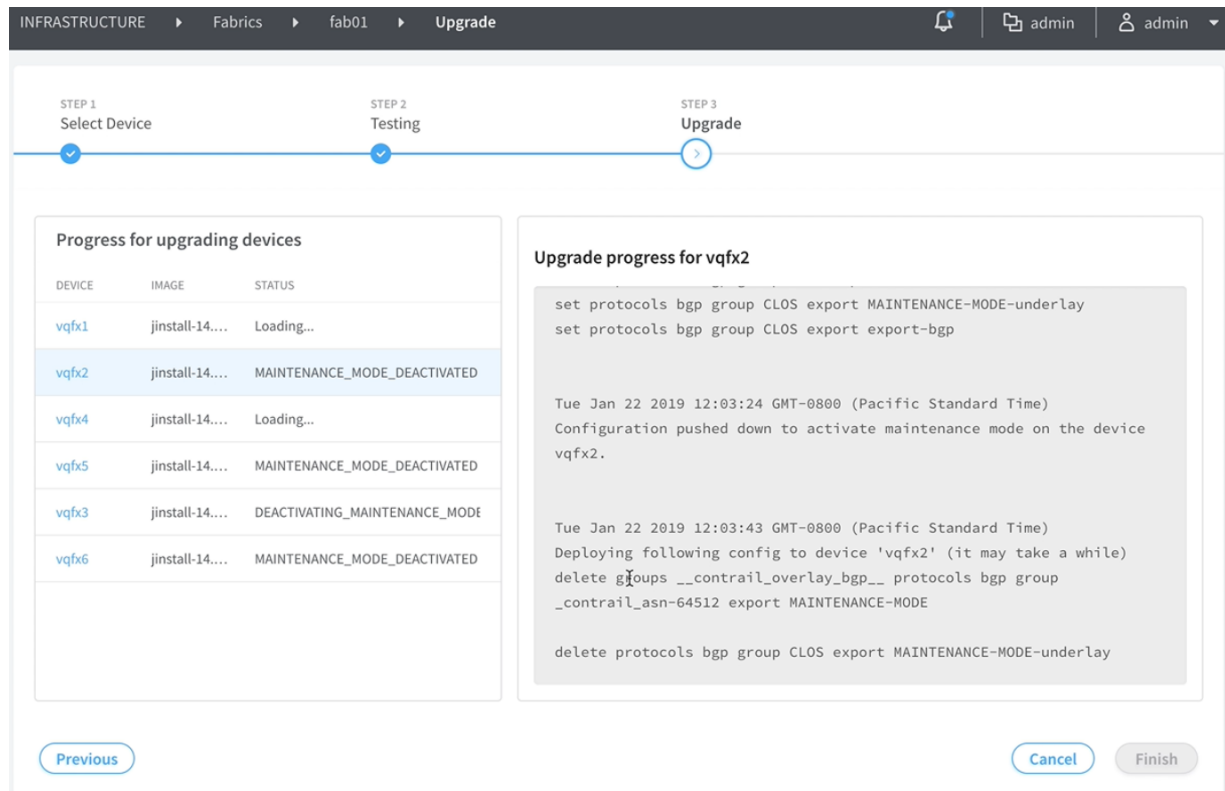
Table 16: Image Upgrade Progress States

State	Description
Loading Validating	The devices are prepping for the upgrade by running health checks.
Health Check Failed	Health check on the device has failed. You can click Previous and go back the <b>Parameters</b> page to either change the health check parameter value or disable the parameter.
Activating Maintenance Mode	The device has passed health check and the device is being placed under maintenance mode.
Deactivating Maintenance Mode	Removing maintenance mode configuration from device and exiting maintenance mode.
Maintenance Mode Activated	Maintenance mode is active on the device.
Maintenance Mode Deactivated	Deactivating maintenance mode is complete and maintenance mode configuration is successfully removed from the device.
Maintenance Mode Failure	Internal error detected during maintenance mode activation or deactivation.

Table 16: Image Upgrade Progress States (continued)

Hitless Image Upgrade Successful	Device image is successfully upgraded.
Hitless Image Upgrade Failed	Device image is not upgraded.
Skipped	Attempted to upgrade to the same image version or the device family does not support hitless upgrade.

Figure 29: Upgrade



11. Click **Finish** when all the devices have been upgraded.

Alternatively, to cancel the upgrade process, click **Cancel**. The **Infrastructure > Fabrics** page is displayed.

**NOTE:** You can re-enter the upgrade workflow if you exit at any point in the process. Also, in case of any failure, the reason is available in the device logs.

## RELATED DOCUMENTATION

[Hitless Software Upgrade of Data Center Devices Overview | 77](#)

## Creating Layer 3 PNF Service Chains for Inter-LR Traffic

### IN THIS SECTION

- [Create a Fabric | 88](#)
- [Create PNF Service Template | 92](#)
- [Create PNF Service Instance | 94](#)
- [View Service Appliance Sets and Service Appliances | 95](#)

Starting with Release 5.1, Contrail provides layer 3 physical network functions (PNF) support to create service chains for inter-LR (logical router) traffic. Contrail Release 5.1 automates configuration of QFX and SRX devices to allow movement of inter-LR traffic between bare metal servers through layer 3 PNF.

These topics provide instructions to create service chains for inter-LR traffic.

### Create a Fabric

Follow these steps to create a fabric with brownfield devices by using the Contrail Command UI:

1. Click **Fabrics**.

The Fabrics page is displayed.

2. Click **Create**.

You are prompted to select a provisioning option.

3. Click **Existing Fabric** to import existing (brownfield) devices by discovery.




Figure 30: Select Provisioning Option

Select provisioning option

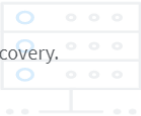
New Fabric

Wizard takes you through deployment of new devices which require discovery, zero touch provisioning(ZTP) and complete configuration.



Existing Fabric

Import existing deployed devices by discovery.



Cancel

Provision

4. Click **Provision**.

The Create Fabric page is displayed.

Figure 31: Create Fabric Page

STEP 1  
Create Fabric

STEP 2  
Device discovery

STEP 3  
Assign the roles

STEP 4  
Autoconfigure

STEP 5 (optional)  
Assign Telemetry Profiles

Name \*

Overlay ASN (BGP) \*

64512

Node profiles \*

device-functional-gr...

juniper-mx

juniper-qfx10k

juniper-qfx10k-lean

juniper-qfx5120

juniper-qfx5k

juniper-qfx5k-lean

juniper-srx

☐ Disable VLAN-VN Uniqueness Check

☒ VLAN-ID Fabric-Wide Significance

Expand All

Collapse All

Device credentials

Username \*

Password \*

Cancel

Next

5. Enter the following information:

Table 17: Provision Existing Fabric

Field	Action
Name	Enter a name for the fabric.

Table 17: Provision Existing Fabric (*continued*)

Field	Action
<b>Username</b>	Enter a username for the device.
<b>Password</b>	Enter a password for the device.
<b>Overlay ASN (iBGP)</b>	Enter an autonomous system number (ASN).  The AS number can be in the range from 1 through 65,535.
<b>Node profiles</b>	Add node profiles.  You can add more than one node profile.  All preloaded node profiles are added to the fabric by default. You can remove a node profile by clicking <b>X</b> on the node profile.
<b>Management subnets</b>	Enter the following information:  <b>CIDR</b> —Enter CIDR network address.  <b>Gateway</b> —Enter gateway address.  <b>NOTE:</b> You enter the CIDR address range in the <b>Management subnets</b> field to search for devices. Any device that has a previously configured management IP on the subnet is discovered.
<b>Underlay ASNs (eBGP)</b>	Enter autonomous system number (ASN) in the range from 1 through 65,535.  <ul style="list-style-type: none"> <li>• Enter minimum value in <b>ASN From</b> field.</li> <li>• Enter maximum value in <b>ASN To</b> field.</li> </ul>
<b>Fabric subnets (CIDR)</b>	Enter fabric CIDR address.  <b>NOTE:</b> Fabric subnets are used to assign IP addresses to interfaces that connect to leaf or spine devices.
<b>Loopback subnets (CIDR)</b>	Enter loopback address.  <b>NOTE:</b> Loopback subnets are used to auto-assign loopback IP addresses to the fabric devices.

Table 17: Provision Existing Fabric (*continued*)

Field	Action
PNF Servicechain subnets (CIDR)	<p>Enter PNF device CIDR address.</p> <p><b>NOTE:</b> Starting in Contrail Release 5.1, enter the subnet for allocating IP addresses in the <b>PNF Servicechain subnets</b> field to establish eBGP session between PNF device and SPINE switch.</p>

6. Click **Next**.

The Discovered devices page is displayed.

The **Device discovery progress** bar on the Discovered devices page displays the progress of the device discovery job. The list of devices discovered are listed in the Discovered devices page.

7. Select the Juniper SRX device you want to add to the fabric and then click **Add**.

**NOTE:** With Contrail Release 5.1, Juniper SRX devices are discovered as well.

The device is added to the fabric.

8. Click **Next** to assign roles.

The Assign to devices page is displayed.

9. Click the **Assign** icon at the end of the row to assign roles.

The Assign role to devices pop-up is displayed.

## 10. Assign physical roles and routing bridging roles.

**For Spine Devices:**

- Select **spine** from the Physical Role list.
- Select **CRB-Gateway** from the Routing Bridging Role list.

**For Leaf Devices:**

- Select **leaf** from the Physical Role list.
- Select **CRB-Access** from the Routing Bridging Role list.

**For PNF Devices:**

- Select **PNF** from the Physical Role list.
- Select **L3PNF** from the Routing Bridging Role list.

11. Click **Assign** to confirm selection and then click **Autoconfigure** to initiate the auto-configuration job.

The Autoconfigure page is displayed.

**NOTE:** The number of PNF instances you can create depends on the subnet mask of the pnf-servicechain-subnet that you provided during fabric onboarding. You can create multiple /29 subnets from the pnf-servicechain-subnet.

For example, if a /24 subnet is provided for the pnf-servicechain-subnet, then, you can create  $2^5 = 32(29-24=5)$  subnets out of it. Each PNF uses a pair of /29 subnets. Thus, for a /24 subnet, you can have a maximum of 16 PNFs.

## Create PNF Service Template

Follow these steps to create a PNF service template by using the Contrail Command UI.

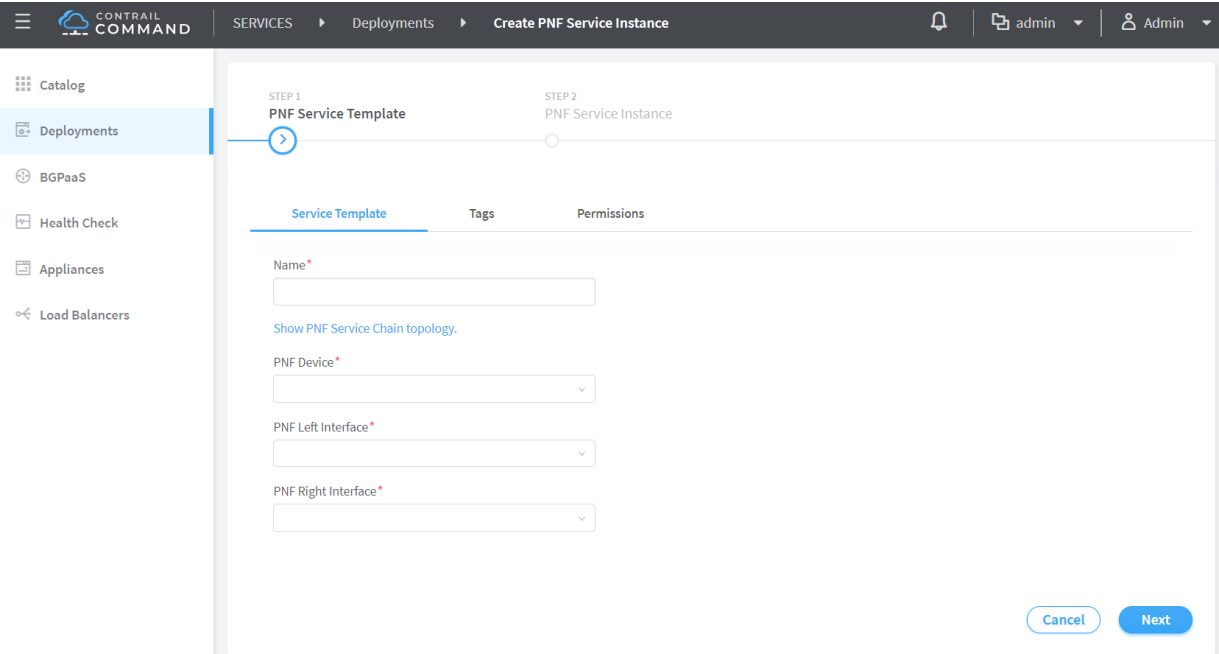
1. Click **Infrastructure>Services>Deployments**.

The VNF Service Instances page is displayed.

2. Click the **PNF** tab.

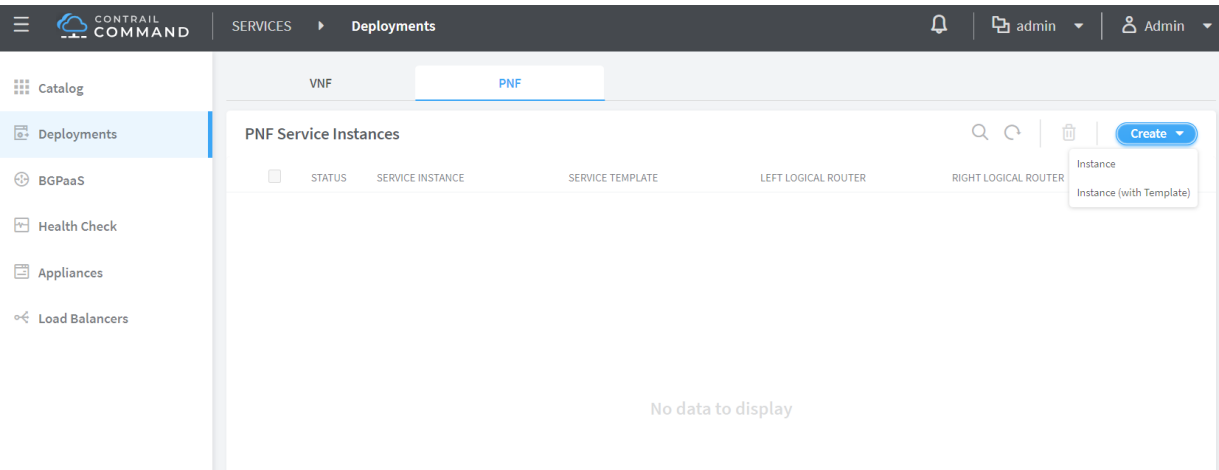
The Create PNF Service Instance page is displayed.

Figure 32: Create PNF Service Instance Page



- 3. Click **Create**.
- 4. Select **Instance (with Template)** from the list that appears as shown in [Figure 33 on page 93](#).

Figure 33: PNF Tab



The Create PNF Service Instance page is displayed.

- 5. Enter the following information in the PNF Service Template pane:

Table 18: Enter PNF Service Template Information

Field	Action
Name	Enter a name for the PNF template.
PNF Device	Select the Juniper SRX device from the list.
PNF Left Interface	Select the Juniper SRX device left interface from the list.
PNF Left Fabric	Select the fabric the SRX device left interface belongs to.
PNF Left Attachment Points	Select the Juniper QFX device from the <b>Physical Router</b> list. Select the left interface from the <b>Left Interface</b> list.
PNF Right Interface	Select the Juniper SRX right interface from the list.
PNF Right Fabric	Select the fabric the SRX device right interface belongs to.
PNF Right Attachment Points	Select the Juniper QFX device from the <b>Physical Router</b> list. Select the right interface from the <b>Right Interface</b> list.

- Click **Next** to confirm.

The PNF Service Instance pane is displayed.

## Create PNF Service Instance

Follow these steps to create a PNF Service Instance by using the Contrail Command UI:

- Enter a name for the service instance in the **Name** field.
- Select the service template you created from the **Service Template** list.

**NOTE:** You can create and launch multiple layer 3 PNF service instances in a topology. The service instances that you launch must be associated to a single service template.

- Enter the PNF device ASN in the **PNF eBGP ASN** field.

**NOTE:** With Contrail Release 5.1, only Juniper SRX devices are supported as PNF devices.

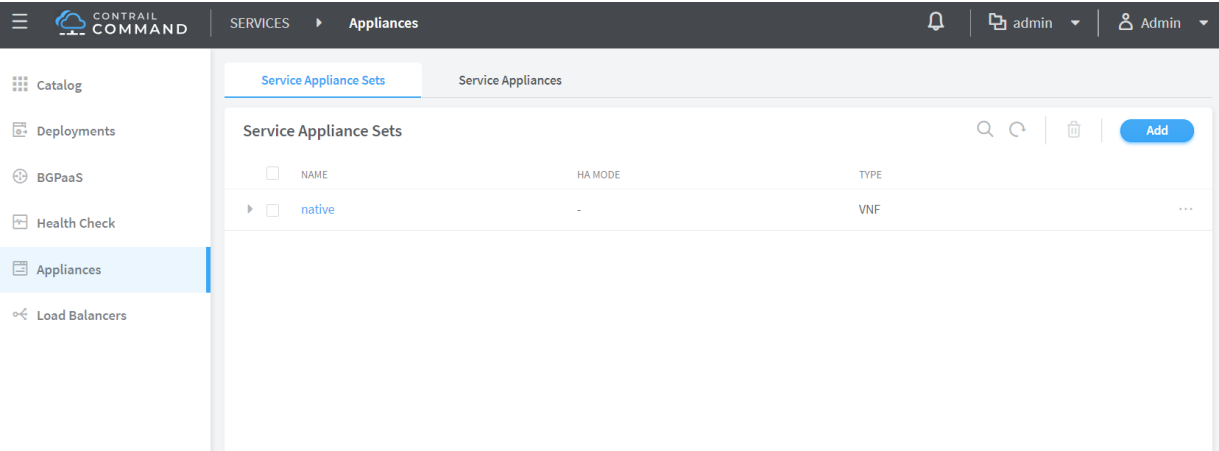
4. Enter the following information for the **left** interface:
  - a. Select left-LR from the **Left Tenant Logical Router** list.  
Left-LR is where the service chain starts.
  - b. Enter the BGP ASN for the Service Instance.
  - c. Enter left VLAN information in the **Left Service VLAN** field.
5. Enter the following information for the **right** interface:
  - a. Select right-LR from the **Right Tenant Logical Router** list.  
Right-LR is where the service chain ends.
  - b. Enter the BGP ASN for the Service Instance.
  - c. Enter right VLAN information in the **Right Service VLAN** field.
6. Click **Finish** to create the service chain.
7. Click **Create** to confirm.

## View Service Appliance Sets and Service Appliances

(Optional) Follow these steps to view Service Appliance Sets and Service Appliances by using the Contrail Command UI:

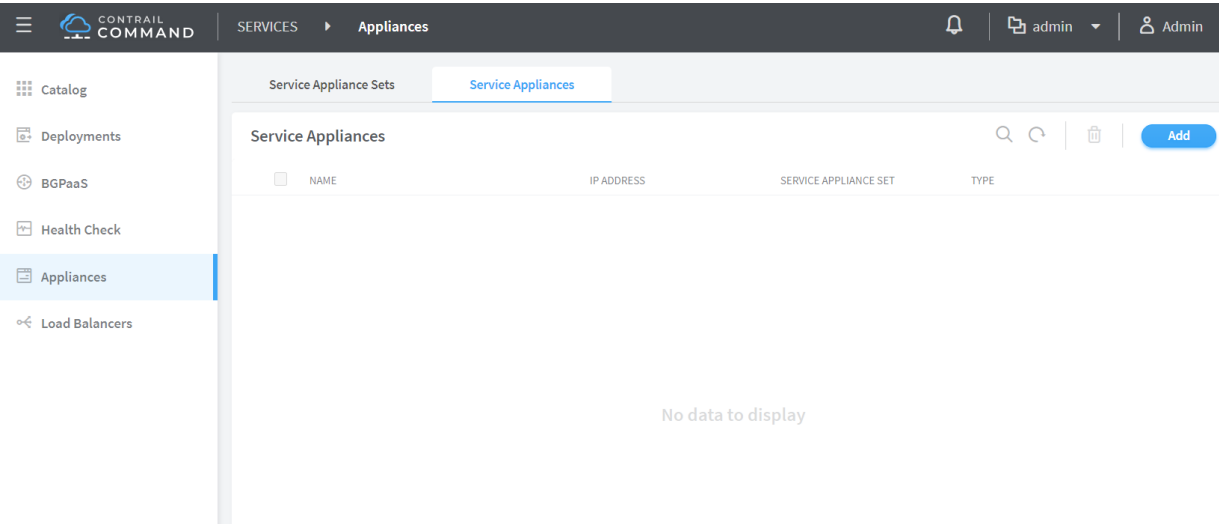
1. Click **Services > Appliances**.  
The Appliances page is displayed.
2. Click **Service Appliance Sets** tab to view the list of available service appliance sets.

Figure 34: View Service Appliance Sets



3. Click **Service Appliance** tab to view the list of available service appliances.

Figure 35: View Service Appliances



RELATED DOCUMENTATION

Providing Intent Driven Automation Capabilities on Physical Network Elements | 16



# Running Generic Device Operations Commands In Contrail Command

Contrail Release 5.1 and later enables you to obtain device information, such as interface information, like input rate or output rate, or search for the name of an interface by providing its MAC address or IP address from the Contrail Command UI. You can run a specific generic device operations command on multiple devices at a time. A job template is defined for each generic device operations command. After you select the devices and specify the parameters defined in the job template, a job is created depending on the generic command you selected. The result of the job is then displayed for the selected device or devices.

You can select a maximum of 20 devices at a time and run a generic device operations command to view information about those devices.

You can run the following generic device operations commands:

- **show interfaces**—Use this generic device operations command to show a list of all runtime interfaces. You can use the filters to select the type of interface, such as physical or logical. You can also view particular types of interfaces using the **regex** filter.
- **show configured interfaces**—Use this generic device operations command to list all the configured interfaces. You can use the filters to select the type of interface, such as physical or logical. You can also view particular types of interfaces using the **regex** filter..
- **show interfaces by names**—Use this generic device operations command to check whether a particular type of interface is present in one or more of the devices selected. This operation is useful when you want to check which among the selected devices has an **xe-0/0/2** interface or an **lo0.0** interface. You can use the filters to select the type of interface, such as physical or logical. You can then enter the interface name you want to search for.
- **search using MAC or IP address**—Use this generic device operations command to identify the interface name if you know the IP address or the MAC address of an interface. This operation is useful to locate the interface by specifying the interface name and information such as name of the originating device and its loopback IP address.

You can create a custom generic device operations command by adding a **job\_template** object type in the **opt/contrail/fabric\_ansible\_playbooks/conf/predef\_payloads.json** file. Follow these best practices when you define a new generic device operations command.

- Make sure that **template\_type** is set to **device\_operation**, which identifies this template as a generic device operation job template.
- Create a new **job\_template** for every generic device command you need to execute. Specify the command name in the **job\_template\_name** field so that it is easy to identify the command.

- Make sure that the generic device operation **job\_templates** references to the playbook **/opt/contrail/fabric\_ansible\_playbooks/operational\_command.yml**.
- Any change to the **predef\_payloads.json** requires a restart of the **config\_api\_1\_xxxx** docker.

To run a generic device operations command:

1. Navigate to **Infrastructure > Fabrics > fabric name**.
2. Select the fabric devices and click the **Run a Custom Action** button as shown in [Figure 36 on page 98](#).

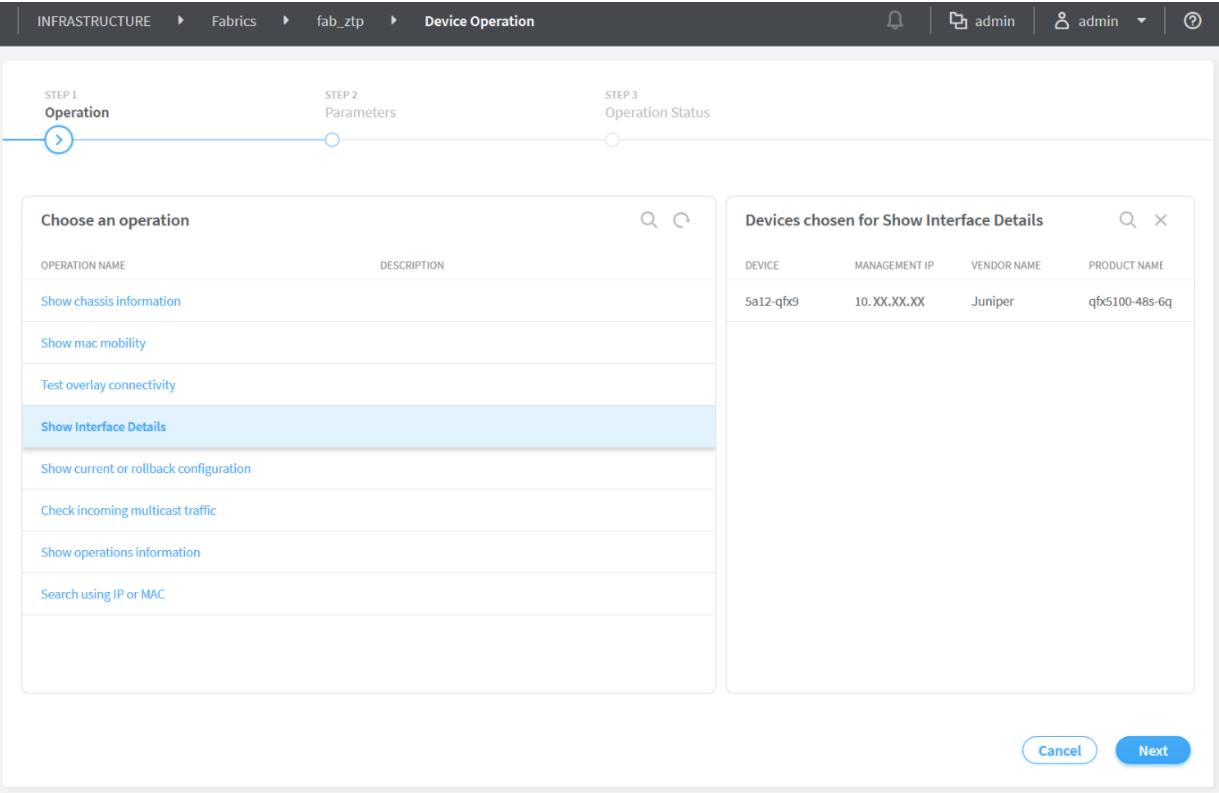
**Figure 36: Select fabric Devices**

The screenshot shows the Contrail Fabric management interface. The breadcrumb navigation at the top indicates the path: **INFRASTRUCTURE > Fabrics > fab\_ztp**. A blue box highlights the **Run a Custom Action** button in the top toolbar. Below this, the **Fabric devices** section contains a table with columns: **STAT**, **NAME**, **MANA**, **LOOP**, **VEND**, **PROD**, **ROLE**, **ROUT**, and **INTEF**. Two devices are listed, both with a green **ACT** status. The first device is a **leaf** role, and the second is a **spine** role. To the right of the table is an **Action** dropdown menu. On the far right, a sidebar displays **Namespaces** with a table of NAME and VALUE, and a section for **Device Credentials**.

NAME	VALUE
overlay_ibgp_asn	64512 ASN
loopback-subnets	10.10.10.0/27 CIDR
fabric-subnets	20.20.20.0/27 CIDR
eBGP-ASN-pool	64000-65000 ASN
management-subnets	10.87.5.128/27 CIDR

3. Click the operation that you want to perform and click **Next**.

Figure 37: Choose an Operation



4. Click an operation. Details of the devices selected are displayed as shown in [Figure 38 on page 100](#).

Figure 38: Devices Selected for the Operation

INFRASTRUCTURE ▸ Fabrics ▸ fab\_ztp ▸ Device Operation

STEP 1 Operation ✓ STEP 2 Parameters STEP 3 Operation Status

**Sub Operation Type**

Choose a sub-operation\*

Show runtime interfaces ^

Show configured interfaces

Show interfaces by names

Show runtime interfaces

**Filter\***

Filter Expression

Filter Type

+ Add

**Interface Details**

Previous Cancel Execute

**Devices chosen for Show Interface Details**

DEVICE	MANAGEMENT IP	VENDOR NAME	PRODUCT NAME
5a12-qfx9	10.XX.XX.XX	Juniper	qfx5100-48s-6q

5. Click **Next** and select the filters.

Figure 39: Select Filters

CONTRAIL COMMAND

INFRASTRUCTURE ▸ Fabrics ▸ PNF ▸ Device Operation

Servers

Cluster

Fabrics

Public Cloud

STEP 1 Operation ✓ STEP 2 Parameters STEP 3 Operation Status

**Interface Type**

physical

**Interface Filters**

**Filter\***

Filter Expression

Filter Type

+ Add

regex

startswith

Previous Cancel Execute

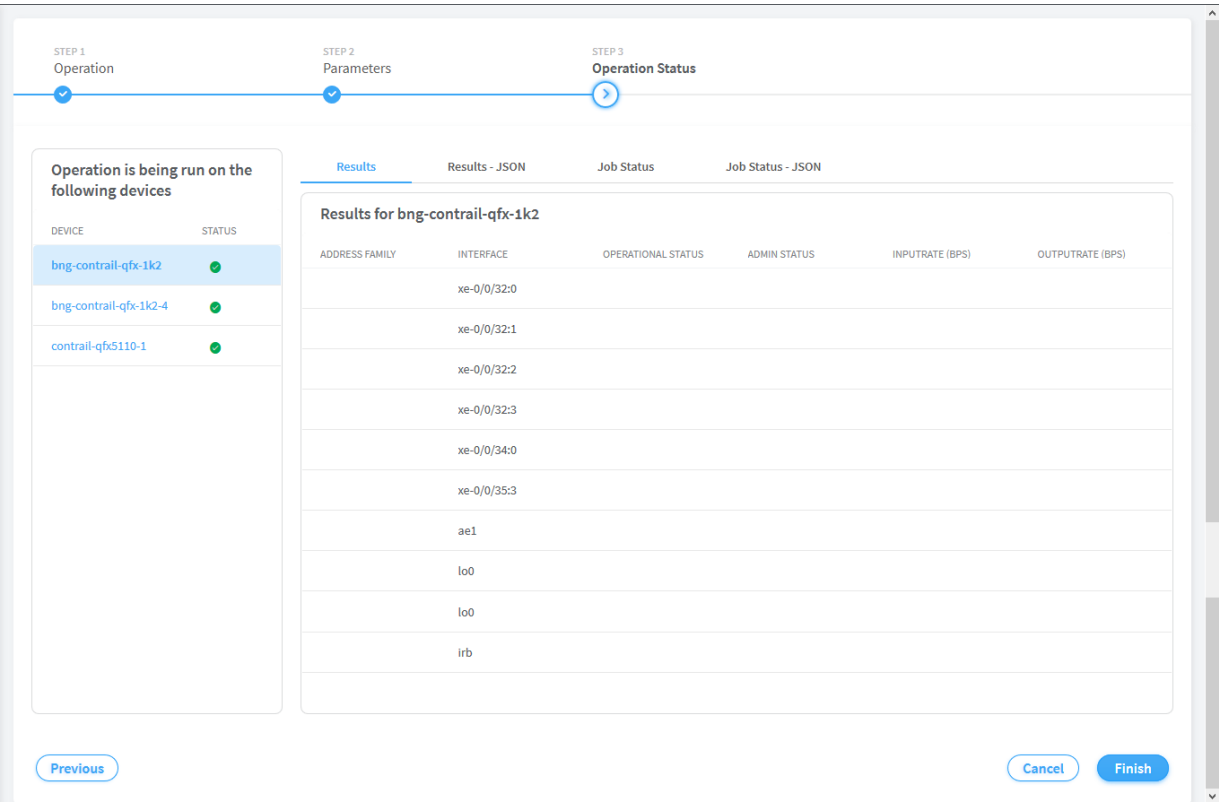
**Devices chosen for Show Configured Interfaces**

DEVICE	MANAGEMENT IP	VENDOR NAME	PRODUCT NAME
bng-contrail-...	10.204.216.155	Juniper	qfx10002-36q
bng-contrail-...	10.204.216.122	Juniper	qfx10002-36q
contrail-qfx51...	10.204.216.131	Juniper	qfx5110-48s-4c

6. Click **Execute**.

Information about the selected device is displayed as shown in [Figure 40 on page 101](#).

Figure 40: Generic Device Operation Command Results



7. Click **Finish** to complete the operation.

# Certificate Lifecycle Management Using Red Hat Identity Management

## IN THIS SECTION

- Fully Qualified Domain Names | 102
- Performing Lifecycle Management of Certificates using Identity Management | 102

Contrail Release 5.1 supports using Transport Layer Security (TLS) with RHOSP to perform lifecycle management, including renewal, expiration, and revocation, of certificates using Red Hat Identity Management (IdM). Because IdM uses fully qualified domain names (FQDNs) to manage endpoints instead of IP addresses, Contrail services are also enhanced to use FQDNs.

Prior to Contrail Networking Release 5.1, lifecycle management of certificates was done manually.

## Fully Qualified Domain Names

Contrail Networking Release 5.1 is integrated with IdM to perform lifecycle management of certificates. Because IdM uses fully qualified domain names (FQDNs) to manage endpoints instead of IP addresses, Contrail services are also enhanced to use FQDNs in the following scenarios:

- Establishing connections between Contrail components
- Input parameters for Contrail Docker container instead of IP addresses
- Contrail TripleO Heat Templates pass FQDNs instead of IP addresses for configuration of Contrail containers using only TLS. You can configure TripleO Heat Templates to pass FQDNs without TLS by setting the **contrail\_nodes\_param\_suffix**: 'node\_names' option.
- Certificates are issued for every Contrail node and stored in the /etc/contrail/ssl folder which is mounted on all Docker containers

## Performing Lifecycle Management of Certificates using Identity Management

Perform the following steps to install the IdM server and manage certificates.

1. Deploy and configure IdM server.

For information on installing an IdM server, see [Installing an IdM Server: Introduction](#).

2. Before deploying the undercloud, set up the **novajoin** plugin on the undercloud node.

```
$ sudo yum install python-novajoin
$ sudo /usr/libexec/novajoin-ipa-setup \
--principal admin \
--password <IdM admin password> \
--server <IdM server hostname> \
--realm <overcloud cloud domain (in upper case)> \
--domain <overcloud cloud domain> \
```

```
--hostname <undercloud hostname> \
--precreate
```

### 3. Prepare the undercloud configuration.

```
[DEFAULT]
enable_novajoin = true
ipa_otp = <otp>    # is returned at previous step
undercloud_hostname = <undercloud FQDN>
undercloud_nameservers = <IdM IP>
overcloud_domain_name = <domain>
...
```

### 4. Check if firewalld is enabled on the IPA (Identity, Policy, Audit) server and the required ports are allowed.

```
rpm -qa | grep firewalld
```

If firewalld is not installed, the undercloud installation will fail. To install firewalld, use the following command:

```
yum install firewalld
firewall-cmd --permanent
--add-port={80/tcp,443/tcp,389/tcp,636/tcp,88/tcp,88/udp,464/tcp,464/udp,53/tcp,53/udp,123/udp}

firewall-cmd --permanent --add-service={freeipa-ldap,freeipa-ldaps,dns}
```

### 5. Deploy the undercloud.

```
$ openstack undercloud install
$ source stack rc
```

### 6. (Optional) Check the following services:

```
(undercloud) [stack@queensa ~]$ systemctl |grep nova
novajoin-notify.service
loaded active running    OpenStack Nova IPA Notification Service
novajoin-server.service
loaded active running    OpenStack Nova IPA Join Service
```

```

    openstack-nova-api.service
loaded active running   OpenStack Nova API Server
    openstack-nova-compute.service
loaded active running   OpenStack Nova Compute Server
    openstack-nova-conductor.service
loaded active running   OpenStack Nova Conductor Server
    openstack-nova-scheduler.service
loaded active running   OpenStack Nova Scheduler Server

```

## 7. Configure overcloud DNS and overcloud domain names.

```
$ openstack subnet set ctlplane-subnet --dns-nameserver <idm_server_address>
```

## 8. Add overcloud domain names to the **contrail-net.yaml** environment file.

```

DnsServers: ["<idm_server_address>"]
CloudDomain: lab.local
CloudName: overcloud.lab.local
CloudNameInternal: overcloud.internalapi.lab.local
CloudNameStorage: overcloud.storage.lab.local
CloudNameStorageManagement: overcloud.storagemgmt.lab.local
CloudNameCtlplane: overcloud.ctlplane.lab.local

```

## 9. Deploy overcloud with the following environment files.

```

$ openstack overcloud deploy --templates ~/tripleo-heat-templates \
-e ~/overcloud_images.yaml \
-e ~/tripleo-heat-templates/environments/network-isolation.yaml \
-e ~/tripleo-heat-templates/environments/contrail/contrail-plugins.yaml \
-e ~/tripleo-heat-templates/environments/contrail/contrail-services.yaml \
-e ~/tripleo-heat-templates/environments/contrail/contrail-net.yaml \
-e ~/tripleo-heat-templates/environments/contrail/contrail-tls.yaml \
-e ~/tripleo-heat-templates/environments/ssl/enable-internal-tls.yaml \
-e ~/tripleo-heat-templates/environments/ssl/tls-everywhere-endpoints-dns.yaml \
\
-e
~/tripleo-heat-templates/environments/services/haproxy-internal-tls-certmonger.yaml \
\
-e
~/tripleo-heat-templates/environments/services/haproxy-public-tls-certmonger.yaml \
\

```



```
--roles-file ~/tripleo-heat-templates/roles_data_contrail_aio.yaml
```

The **contrail-net.yaml**, **enable-internal-tls.yaml**, **tls-everywhere-endpoints-dns.yaml**, **haproxy-internal-tls-certmonger.yaml**, and **haproxy-public-tls-certmonger.yaml** files enable TLS.

10. Check that the host is added to the IPA server.

```
# login to IPA
(undercloud) [stack@undercloud ~]$ kinit admin
(undercloud) [stack@undercloud ~]$ ipa host-find undercloud.my3domain
----- 1 host matched -----
Host name: undercloud.my3domain Description:
Undercloud host Principal name: host/undercloud.my3domain@MY3DOMAIN
Principal alias: host/undercloud.my3domain@MY3DOMAIN
SSH public key fingerprint: SHA256:GAMClAFAgNN709Kb9AcFWfUG30Y06pcR0EdJBWXWIak
    (ssh-rsa), SHA256:KqTDFKQEoKKi7FMzuhBwcO+Y/O9t4rHXQcqPKglJPmI
    (ecdsa-sha2-nistp256), SHA256:QSIBCIiRW03eR6+PPyvDWiWEHXC1MewREAt8hMTUOgU
    (ssh-ed25519)
```

11. View the list of monitored certificates on an overcloud node.

```
[heat-admin@overcloud-novacompute-1 ~]$ sudo getcert list
Number of certificates and requests being tracked: 4.
Request ID 'contrail': status: MONITORING
stuck: no key pair storage:
type=FILE,location='/etc/contrail/ssl/private/server-privkey.pem'
certificate: type=FILE,location='/etc/contrail/ssl/certs/server.pem'
CA: IPA
issuer: CN=Certificate Authority,O=MY3DOMAIN
subject: CN=overcloud-novacompute-1.my3domain,O=MY3DOMAIN
expires: 2021-04-20 14:18:21 UTC
dns:
overcloud-novacompute-1.ctlplane.my3domain,overcloud-novacompute-1.internalapi.my3domain,
overcloud-novacompute-1.tenant.my3domain,overcloud-novacompute-1.my3domain
principal name: contrail/overcloud-novacompute-1.my3domain@MY3DOMAIN
key usage: digitalSignature,nonRepudiation,keyEncipherment,dataEncipherment
eku: id-kp-serverAuth,id-kp-clientAuth
pre-save command:
post-save command: "sudo docker ps -q --filter=name="contrail*" | xargs -i sudo
    docker restart {}"
track: yes
auto-renew: yes
```

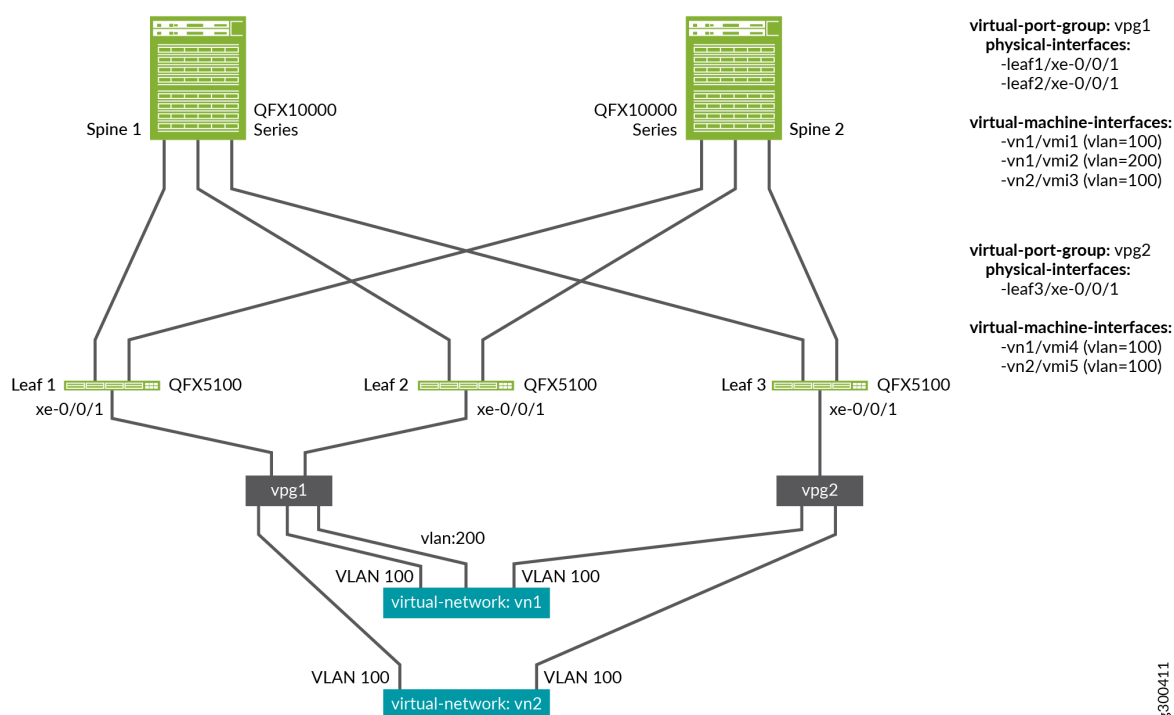
# Virtual Port Groups

A virtual port group (VPG) is a group of one or more physical interfaces attached to one or more virtual machine interfaces (VMI). Each VMI object corresponds to a VLAN ID and is attached to a Virtual Network. You can select multiple interfaces on the same device or on different devices. A VPG is similar to the link aggregation group (LAG) but supports both LAG and multihoming depending on whether you select the interfaces on the same devices or on different devices. A LAG is automatically created if you select more than one interface on the same device.

**NOTE:** DHCP is not supported for virtual port groups.

Figure 41 on page 106 is an illustration of how the interfaces belonging to two devices are grouped using virtual port group. **vpg1** and **vpg2** are two virtual port groups that group the physical interfaces on three QFX5100 devices.

Figure 41: Virtual Port Group



You can view the existing virtual port groups from the Virtual Networks page. You can create virtual port group either when you create a virtual network or from the Virtual Port Group page in Contrail Command. For information about creating virtual port groups, see [“Configuring Virtual Port Groups” on page 107](#).

# Configuring Virtual Port Groups

This topic describes how to create virtual port groups from Contrail Command UI.

To create virtual port groups:

1. Navigate to Overlay > Virtual Port Group > Create Virtual Port Group.

The Create Virtual Port Group page is displayed as shown in [Figure 42 on page 107](#).

**Figure 42: Create Virtual Port Group**

The screenshot shows the 'Create Virtual Port Group' page in the Contrail Command UI. The page is divided into several sections:

- Virtual Port Group Name:** A text input field containing 'vpg-internal-0'.
- VLAN Configuration:**
  - Tagged:** A checkbox that is checked.
  - VLAN id:** A text input field containing '1'.
  - TOR Port VLAN id:** A text input field containing '4094'.
  - Display Name:** A text input field containing 'fa085e76-e0b7-43cb-8ca7-e3c'.
  - Auto Display Name:** A checkbox that is unchecked.
- Network:** A dropdown menu showing 'right\_vn\_1'.
- Security Groups:** A dropdown menu showing 'default'.
- Fabric name:** A dropdown menu showing 'PNF'.
- Available Physical Interface:** A table listing available physical interfaces.
 

DISPLAY NAME	PHYSICAL ROUTER
ge-0/0/45	contrail-qfx5110-7
ge-0/0/6	contrail-qfx5110-6
reth2	contrail-srx5600-2
xe-1/0/9	contrail-srx5600-2
ge-0/0/34	contrail-qfx5110-7
- Assigned Physical Interface:** A table listing assigned physical interfaces.
 

DISPLAY NAME	PHYSICAL ROUTER
ge-0/0/9	contrail-qfx5110-6

At the bottom of the page, there are 'Create' and 'Cancel' buttons, and a pagination bar showing 'Previous 1 2 3 4 Next'.

2. Enter the VLAN ID and network to which the VLAN is associated and select a security group to which the VLAN is to be attached.

You can select multiple VLANs to include in the virtual port group. Based on the need, you can add or remove VLANs from virtual port group by using the **Edit Virtual Port Group** function.

3. Select the fabric from the **Fabric Name** list.

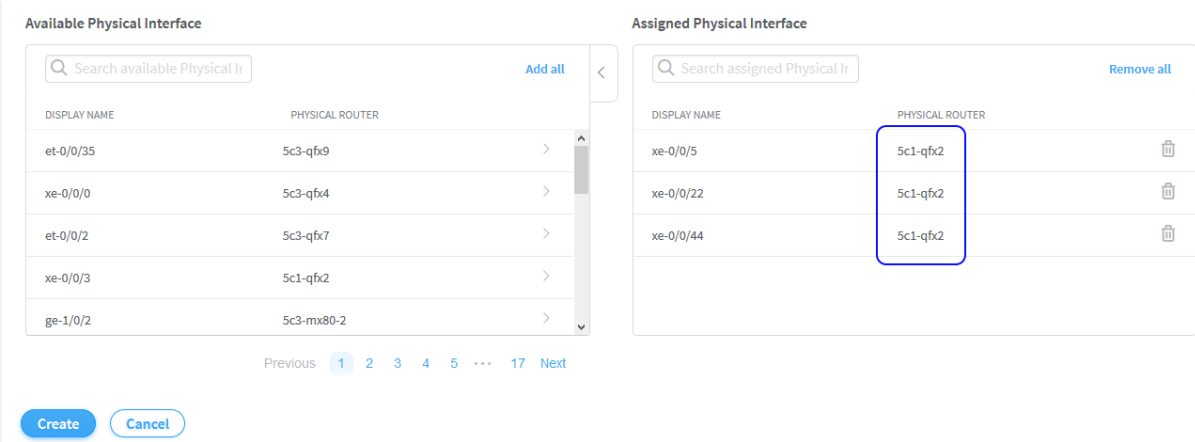
The available physical interfaces on the devices in the selected fabric are listed.

- 4. From the **Available Physical Interface** box, select the physical interfaces to be included in the virtual port group by clicking the arrow next each physical interface. The available physical interfaces are the interfaces available on TORs that are already onboarded.

The selected interfaces are displayed in the **Assigned Physical Interface** box.

If you select more than one interface on the same TOR as shown in [Figure 43 on page 108](#), a link aggregation group (LAG) is automatically created on the device.

Figure 43: Select Interfaces on the Same TOR



- 5. Click **Create**.

The newly created virtual port group is displayed on the Virtual Port Group page with details of the interfaces and the TORs as shown in [Figure 44 on page 109](#).

Figure 44: Virtual Port Groups

NAME	VLAN IDS	TOR PORT VLAN IDS	PHYSICAL INTERFACES	VIRTUAL NETWORK
vpg-internal-0		4094	ge-0/0/0:contrail-qfx5110-6	right_vn_1
vpg-test		4094	fxp0:contrail-srx5600-2 xe-0/0/32:2:bng-contrail-qfx-1... 1 more	left_vn_13

You can delete a virtual port group by clicking the delete icon against the virtual port group. To delete a virtual port group, you must first remove the referenced VMI and the associated BMS instance from the virtual port group.

# Supported Hardware Platforms and Associated Roles

The following tables list the supported hardware platforms and the roles that can be configured on them in Contrail Release 5.1. The starting Contrail release number indicates the first release the devices were supported from and the Junos OS releases indicate the minimum Junos OS release that must be installed on the hardware platforms to configure roles on them.

- For a list of QFX devices, see [Table 19 on page 110](#).
- For a list for MX devices, see [Table 20 on page 111](#).
- For a list of SRX devices, see [Table 21 on page 112](#).

Table 19: Supported QFX Series Switches

QFX Device	Supported from Contrail Release								
	Supported from Junos OS Releases								
	Physical Roles		Special Role	Gateway Roles			Overlay Roles		
	Leaf	Spine	Route Reflector	DC Gateway	DCI Gateway	PNF Service Chaining	CRB-Access	CRB-GW	CRB-M
QFX5110-48S-4C	5.0.2	5.0.2	5.0.2	5.1			5.0.2	5.0.2	
	17.3R3	17.3R3	17.3R3	18.1R3			17.3R3	18.1R3	
QFX5110-32Q	5.0.2	5.0.2	5.0.2	5.1			5.0.2	5.0.2	
	17.3R3	17.3R3	17.3R3	18.1R3			17.3R3	18.1R3	
QFX5120-48Y-8C	5.1	5.1	5.1	5.1			5.1	5.1	
	18.4R2	18.4R2	18.4R2	18.4R2			18.4R2	18.4R2	
QFX5120-32C	5.1	5.1	5.1	5.1			5.1	5.1	
	18.1R3	18.1R3	18.1R3	18.1R3			18.1R3	18.1R3	
QFX5200-32C	5.0.2	5.0.2	5.0.2				17.3R3		
	17.3R3	17.3R3	17.3R3						
QFX5210-64C	5.0.2	5.0.2	5.0.2				17.3R3		
	17.3R3	17.3R3	17.3R3						
QFX10002-36Q	5.0.2	5.0.2	5.0.2	5.1	5.1	5.1	17.3R3	17.3R3	17.3R3
	17.3R3	17.3R3	17.3R3	18.1R3	18.1R3	18.1R3			
QFX10002-72Q	5.0.2	5.0.2	5.0.2	5.1	5.1	5.1	17.3R3	17.3R3	17.3R3
	17.3R3	17.3R3	17.3R3	18.3R3	18.3R3	18.3R3			
QFX10002-60C	5.1	5.1	5.0.2						
	18.4R2	18.4R2	18.4R2						

Table 19: Supported QFX Series Switches (continued)

QFX10008	5.0.2	5.0.2	5.0.2	5.1	5.1	5.1	17.3R3	17.3R3	17.3R3
	17.3R3	17.3R3	17.3R3	18.3R3	18.3R3	18.3R3			
QFX10016	5.0.2	5.0.2	5.0.2	5.1	5.1	5.1	17.3R3	17.3R3	17.3R3
	17.3R3	17.3R3	17.3R3	18.3R3	18.3R3	18.3R3			
QFX5100-48S-6Q	5.0.2	5.0.2	17.3R3				5.0.2		
	17.3R3	17.3R3					17.3R3		
QFX5100-48T-6Q	5.0.2	5.0.2	5.0.2				5.0.2		
	17.3R3	17.3R3	17.3R3				17.3R3		
QFX5100-24Q-2P	5.0.2	5.0.2	5.0.2				5.0.2		
	17.3R3	17.3R3	17.3R3				17.3R3		
QFX5100-96S-8Q	5.0.2	5.0.2	5.0.2				5.0.2		
	17.3R3	17.3R3	17.3R3				17.3R3		
QFX5100-24Q-AA	5.0.2	5.0.2	5.0.2				5.0.2		
	17.3R3	17.3R3	17.3R3				17.3R3		

Table 20: Supported MX Series Routers

MX Device	Supported from Contrail Release Supported from Junos OS Releases				
	Physical Roles		Special Role	Gateway Roles	
	Leaf	Spine	Route Reflector	DC Gateway	DCI Gateway
vMX	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX80	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3

Table 20: Supported MX Series Routers (continued)

MX240	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX480	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX960	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX2008	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX2010	5.0.2	5.0.2	5.0.2	5.0.2	5.1
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX2020	5.0.2	5.0.2	5.0.2	5.0.2	5.0.2
	17.3R3	17.3R3	17.3R3	17.3R3	18.1R3
MX10003	5.1	5.1	5.0.2	5.1	5.1
	18.1R3	18.1R3	17.3R3	18.1R3	18.1R3
MX204	5.1	5.1	5.0.2	5.1	5.1
	18.1R3	18.1R3	17.3R3	18.1R3	18.1R3
MX10008	5.1	5.1	5.0.2	5.1	5.1
	18.1R3	18.1R3	17.3R3	18.1R3	18.1R3
MX10016	5.1	5.1	5.0.2	5.1	5.1
	18.1R3	18.1R3	17.3R3	18.1R3	18.1R3

Table 21: Supported SRX Series Services Gateways

SRX Device	Supported from Contrail Release
	Physical Role
	PNF



Table 21: Supported SRX Series Services Gateways (*continued*)

SRX4600	5.1
SRX4200	5.1
SRX4100	5.1
SRX5800	5.1
SRX5600	5.1
SRX5400	5.1
SRX1500	5.1
vSRX	5.1

# 4

CHAPTER

## Extending Contrail to Bare Metal Servers

---

Bare Metal Server Management | **115**

How Bare Metal Server Management Works | **119**

LAG and Multihoming Support | **123**

Adding Bare Metal Server to Inventory | **125**

Launching a Bare Metal Server | **127**

Onboarding and Discovery of Bare Metal Servers | **128**

Launching and Deleting a Greenfield Bare Metal Server | **130**

Troubleshooting Bare Metal Servers | **131**

---

# Bare Metal Server Management

## IN THIS SECTION

- [Understanding Bare Metal Server Management | 115](#)
- [Features of the Bare Metal Server Management Framework | 117](#)

A bare metal server or a bare metal machine is a physical server that is dedicated to a specific customer, unlike a virtual machine. You can deploy bare metal machines in the same way as you deploy virtual machines by using Contrail UI.

## Understanding Bare Metal Server Management

Starting with Contrail Release 5.0.1, you can manage the life cycle of bare metal servers (BMS) by using a backend framework, which acts as a bare metal server (BMS) manager. The BMS management framework in Contrail uses the functionality provided by the following OpenStack services: ironic, Nova, and Glance. The BMS Management framework or the BMS framework manages the bare metal workload within a fabric. It includes BMS server life cycle management, onboarding of bare metal servers, bare metal image management, flavor management, inventory management, IP address management, security management, monitoring and reporting of life cycle management events, and discovery of bare metal servers.

An Administrative user can configure the BMS framework and a Tenant user can avail the services provided by the BMS framework. [Figure 45 on page 116](#) shows an architectural view of the BMS Management framework.



information. All the nodes onboarded or registered with BMS manager are in **Available** state. After the Tenant user has completed using the bare metal server and remove it, the server is then unprovisioned by the BMS framework and moved to the list of available nodes. Alternatively, the Tenant user can remove the BMS instance from the Tenant's network. For example, if you want to rent a BMS from a service provider, the service provider deploys a BMS instance and gives you an IP address of the BMS instance, which you can use to access the BMS. Once you have completed using the BMS, you can delete the instance and the service provider reclaims the BMS. After reclaiming the BMS the service provider cleans it and rents it to the next client. The BMS framework in Contrail Networking manages all these tasks. If the service provider wants to remove the BMS instance from the service, they can delete it from the available servers and the next tenant will get a new BMS instance from a server.

The BMS framework can install Tenant user-specific software images on BMS and attach them to the Tenant user network in a multi-tenant cloud. It provides a single-click solution for the Tenant users to manage the bare metal servers in their network.

## Features of the Bare Metal Server Management Framework

The BMS management framework provides the following features:

- **BMS Image management**—Provides a list of available bootable images available to the Tenant users to boot their server instances or BMS. The BMS framework uses Glance, which is an OpenStack service used for Image Management.
- **BMS Flavor management**—Provides a list of available flavors of the BMS available in the inventory. The flavors represent the capacity or class of the BMS, such as disk size, memory size, number of cores or the manufacturer of BMS. The BMS framework creates pools of BMS based on their capability, class, or both, and then makes them available to the Tenant users. The BMS framework uses Nova, which is an OpenStack service used to provision computing instances or virtual servers. Nova can be used to create virtual machines and bare metal servers using Ironic. Flavors are used in OpenStack to define the compute, memory, and storage capacity of the Nova computing instances.
- **BMS Life Cycle Management**—Includes the following:
  - **Bringing powered off servers online in a secure manner**—As soon as a BMS is powered off, it is disconnected from the Tenant user network and connected to a cleaning network for clean up of the server. A server is connected to a cleaning network for cleaning operations when it is not being used. If the server is deployed, it is connected to the provisioning network.
  - **Reclaiming the provisioned servers and instances after they are decommissioned by the Tenant users**—After cleaning up, the BMS is added to the pool of available server ready to be deployed as a new BMS. The boot up process is performed on a secure network to prevent the possibility of snooping in a multi-tenant cloud. The cleaning process ensures that the BMS is ready to be deployed with the same or different image when needed.

The BMS framework uses Ironic, which is an OpenStack service used to launch bare metal machines. Ironic integrates with the bare metal driver in Nova to maintain BMS lifecycle management efficiently.

- **BMS Inventory Management**— Maintains an inventory of all the servers under the BMS framework. The inventory includes the deployed instances and servers as well as those available for deployment.
- **BMS IPAM management**— Ensures that the IP address management is consistent between the virtual and physical instances. IPAM is managed by the Contrail controller.
- **BMS Network Security management**— The boot cycle and/or cleaning of bare metal servers are extensive and lengthy processes, which makes provisioning and cleaning phases susceptible for snooping by hackers in multi-tenant cloud environments. Hence, the BMS framework uses private networks for the provisioning and cleaning phases of the servers. Once the servers are ready for deployment, the BMS framework deploys the servers in the Tenant user network.
- **Tenant Network management**— Manages connectivity between the bare metal servers and Tenant user networks or provisioning and cleaning networks depending on the deployment state of the server.
- **BMS discovery and onboarding**— The BMS framework supports both the discovery of new servers as well as onboarding of the brownfield servers.

**NOTE:** A deployed server must be unprovisioned and made available before it can be deleted from BMS node list.

## RELATED DOCUMENTATION

[How Bare Metal Server Management Works | 119](#)

[LAG and Multihoming Support | 123](#)

[Adding Bare Metal Server to Inventory | 125](#)

[Launching a Bare Metal Server | 127](#)

[Onboarding and Discovery of Bare Metal Servers | 128](#)

[Launching and Deleting a Greenfield Bare Metal Server | 130](#)

[Troubleshooting Bare Metal Servers | 131](#)

# How Bare Metal Server Management Works

## IN THIS SECTION

- [Administrative Workflow | 119](#)
- [Tenant Workflow | 122](#)

The BMS management framework is configured by the Administrative user. The Administrative user follows a specific workflow to configure critical data objects, which are then made available to the Tenant users.

## Administrative Workflow

The Administrative user must perform the following workflow to configure the BMS framework:

- Create two private networks from the Contrail Command user interface (UI), which is visible only to the Administrative user. One network is used for provisioning the servers during deployment phase and the other network is used for cleaning up bare metal servers when they are decommissioned. These private networks provide security to these servers from hackers when they are being provisioned or being cleaned up after removing from the tenant network. From the Contrail networking point of view, the private networks are normal virtual networks, except that they are accessible only to the Administrative user.

Follow these steps to create a Virtual Network from the Contrail Command user interface (UI).

1. Click **Overlay>Virtual Networks**.

The All Networks page is displayed.

2. Click **Create** to create a network.

The Create Virtual Network page is displayed.

Figure 46: Create Virtual Network Page

OVERLAY > Virtual Networks > Create Virtual Network

Network Tags Permissions

Name\* ⓘ  
Enter Name

VN Fabric Type ⓘ  
Routed Switched

Network Policies ⓘ  
Select Network Policies ▾

Allocation Mode ⓘ  
User defined subnet only ▾

VxLAN Network Identifier ⓘ  
1 - 16777215

Subnets  
+ Add



3. Enter a name for the network in the **Name** field.
4. Select network policies from the **Network Policies** list. You can select more than one network policy.
5. Select any one of the following preferred allocation mode.
  - Flat subnet only
  - Flat subnet preferred
  - (Default) User defined subnet only
  - User defined subnet preferred
- An allocation mode indicates how you choose a subnet. You select **Flat subnet only** or **Flat subnet preferred** allocation mode when the subnet is shared by multiple virtual networks. However, you select **(Default) User defined subnet only** or **User defined subnet preferred** allocation mode when you want to define a subnet range.
6. The VXLAN ID is populated by default and is displayed in the **VxLAN Network Identifier** field.
7. Enter valid IPv4 subnet or mask in the **CIDR** field.
8. Enter valid IPv4 address in the **Gateway** field.
9. Click **Create**.

The All Networks page is displayed. The virtual networks that you created are displayed in this page.

**NOTE:** Though it is recommended that you create two networks for provisioning and cleaning, alternatively, you can use the same network for both provisioning and cleaning.

- Create the BMS images that are available to the tenants through a catalogue—You use the **diskimage-builder**, a special utility in the OpenStack Ironic service to create BMS images. For more information, see <https://docs.openstack.org/diskimage-builder/latest/>.
- Register the BMS images with Glance service—After the images are registered, these images become available to the Tenant users for deployment. For more information, see <https://docs.openstack.org/ironic/latest/install/configure-glance-images.html>.
- Create bare metal flavors and register with Nova service based on the classes or bare metal servers to be offered or managed—You can create multiple bare metal flavors. For example, **baremetal-huge**, **baremetal-large**, **baremetal-small**, and so on. These flavors are then mapped to the inventory of the

available bare metal servers at the time of deployment. The Tenant users can view the flavors in the Contrail Command UI and use the flavors according to their requirement.

- Create Ironi nodes—A BMS server is represented as an Ironi node. The collection of the nodes form the BMS inventory.

To add a bare metal server to Inventory from the Contrail Command UI, the Administrative user must follow the procedure in [“Adding Bare Metal Server to Inventory” on page 125](#).

- Create Ironi ports—These ports represent the NICs in the bare metal servers. This includes the MAC address and the physical connectivity information.
- Set up PXE boot interface—You set up Preboot Execution Environment (PXE) as part of BMS onboarding (or registering) of bare metal servers.

## Tenant Workflow

After the BMS service is instantiated, the Tenant users are offered a catalog of available services. They select the type of server they want to instantiate and the image they want to run. The Tenant users need to follow the given workflow to avail the services provided by bare metal servers:

- Create Tenant user network—BMS connects to this network when it is ready for use.
- Select the BMS flavor and BMS Image that you want to instantiate and issue a boot command. The Tenant user selects a BMS that is available for deployment using the flavor. They use the flavors that are created by the Administrative user. If no BMS meets the criteria specified by flavor, the launch command is rejected with the error message **No Valid Host found**.

**NOTE:** Booting a bare metal server is very much similar to instantiation of a virtual machine; the only difference is that the Tenant user can select the appropriate flavor for BMS depending on the requirement.

- View availability zone information— An availability zone typically applies to virtual machines and can also be applied to BMS. You can view virtual machine availability zone information and BMS availability zone information in two different zones on the user interface.
- Launch a BMS—A bare metal server is launched in the same way as you launch a virtual machine.

To launch a new bare metal server from the Contrail Command UI, follow the procedure in [“Launching a Bare Metal Server” on page 127](#).

## RELATED DOCUMENTATION

[Bare Metal Server Management | 115](#)[LAG and Multihoming Support | 123](#)[Adding Bare Metal Server to Inventory | 125](#)[Launching a Bare Metal Server | 127](#)[Onboarding and Discovery of Bare Metal Servers | 128](#)[Launching and Deleting a Greenfield Bare Metal Server | 130](#)[Troubleshooting Bare Metal Servers | 131](#)

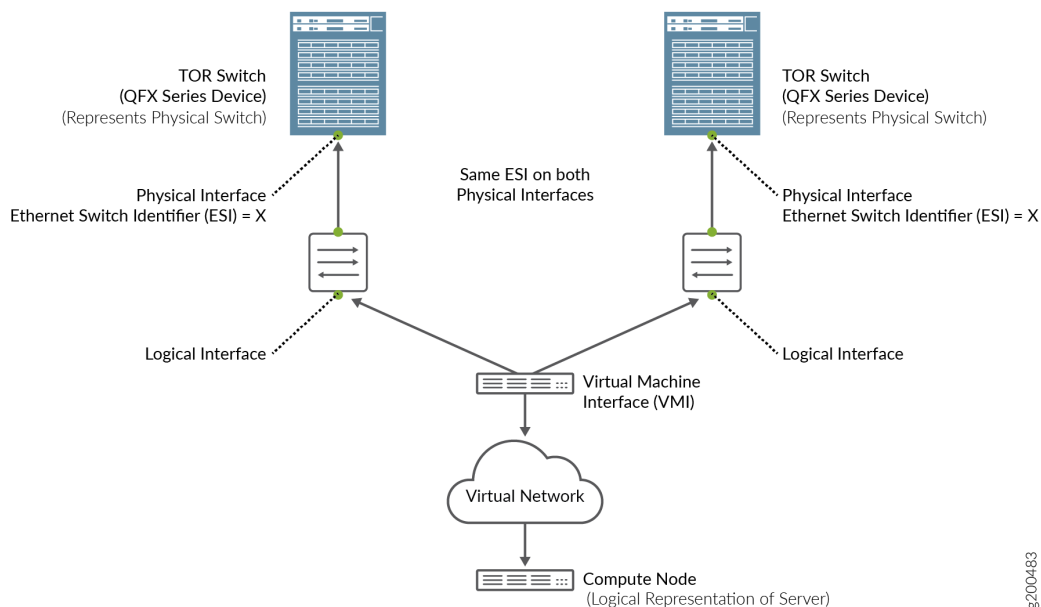
## LAG and Multihoming Support

Bare metal servers connect to multiple TORs to establish redundancy (MLAG/multihomed configurations). Also, depending on the port bandwidth on the TOR and the NICs on the bare metal servers, multiple ports can be utilized to connect a bare metal server to the TOR (LAG configurations). These interfaces are also called as *bond* interfaces. On bonded interfaces, LACP protocol is enabled by default.

In LAG configuration, the two physical interfaces on the TOR switch (a QFX Series device) become members of a link aggregation group (LAG). The LAG connects to the aggregated Ethernet (AE) interface, which is again a physical interface that connects to the logical interface. The logical interface is connected to the virtual machine interface (VMI), which is connected to the virtual network (VN). The VN is connected to the node, which is the logical representation of a bare metal server.

In a multihomed configuration, a single port on a BMS connects to the physical interfaces on two QFX devices. The QFX devices have one physical interface each, both having the same Ethernet switch identifier (ESI). The physical interfaces are assigned the same ESI to enable the QFX device to recognize the interface as a multihomed interface. [Figure 47 on page 124](#) shows how BMS is connected to a TOR switch.

Figure 47: Connectivity in Multihomed Configuration



8200483

## RELATED DOCUMENTATION

[Bare Metal Server Management | 115](#)
[How Bare Metal Server Management Works | 119](#)
[Adding Bare Metal Server to Inventory | 125](#)
[Launching a Bare Metal Server | 127](#)
[Onboarding and Discovery of Bare Metal Servers | 128](#)
[Launching and Deleting a Greenfield Bare Metal Server | 130](#)
[Troubleshooting Bare Metal Servers | 131](#)

# Adding Bare Metal Server to Inventory

The Administrative user must follow these steps to add a bare metal server to Inventory from the Contrail Command user interface (UI):

1. Click **Infrastructure**> **Servers**.

A list of servers is displayed.

2. Click **Create** to create a server.

The Create Server page is displayed.

3. Select **Detailed** button from the Choose Mode options.

4. Select **Baremetal** button from the **Select workload type this server will be used for** options.

5. Enter a name for the host in the **Hostname** field.

6. Enter appropriate credentials for host in the **Credentials** field.

7. Select required kernel from the **Deploy Kernel** list.

8. Select required ram size from **Deploy Ramdisk** list.

9. Add the following values in the **Network Interfaces** fields:

Field	Action
Name	Assign a name to the Port.
MAC address	Enter the MAC address of the Port.
Device/ TOR- Interface	Select the Leaf/ TOR- Interface to which the Port is connected.
Enable PXE	Select this checkbox to enable PXE booting for only one Port.

10. Add the following values in the **Port Groups** field:

Field	Action
Name	Assign a name to the Port Group.

Field	Action
Member Interfaces	Add the ports that form the Port Group.

11. Add the following values in the **IPMI Info** fields:

Field	Action
IPMI Driver	Enter valid IPMI Driver name. The driver value for Openstack SKU: queens and ocata is <b>pxe_ipmitool</b> . The driver value for Openstack SKU: rocky is <b>ipmi</b> . For more information, you can refer to Openstack document: <a href="#">Enabling drivers and hardware types</a> .
IPMI Address	Enter the IPMI Address of the BMS Server.
IPMI Port	Enter port number on which IPMI is deployed. The default value is 623, as shown in the Contrail Command UI. You can update this to different IPMI port, according to your requirement.
IPMI Username	Enter IPMI Username.
IPMI Password	Enter IPMI Password.

12. Add the following values in the **Baremetal Properties** field based on the capacity of the server:

Field	Action
Memory mb	Enter RAM size of BMS Server in megabytes (Mb).
CPU's	Enter CPU count of BMS Server.
CPU Arch	Enter CPU Architecture of BMS Server. The default value is x86_64.
Local gb	Enter Disk Size of BMS Server in gigabytes (Gb).
Capabilities	This is the sets the capability of BMS Server. The default value is "boot_option:local".

13. Click **Create**. The **Servers** page is displayed with the list of servers created by the Administrative user.

RELATED DOCUMENTATION

<a href="#">Bare Metal Server Management   115</a>
<a href="#">How Bare Metal Server Management Works   119</a>
<a href="#">LAG and Multihoming Support   123</a>
<a href="#">Launching a Bare Metal Server   127</a>
<a href="#">Onboarding and Discovery of Bare Metal Servers   128</a>
<a href="#">Launching and Deleting a Greenfield Bare Metal Server   130</a>
<a href="#">Troubleshooting Bare Metal Servers   131</a>

# Launching a Bare Metal Server

The Tenant user must follow these steps to launch a new bare metal server (BMS) from the Contrail Command UI:

1. Click **Workloads>Instances**.  
The Instances page is displayed.
2. Click **Create** to create a new instance.  
The Create Instance page is displayed.
3. Select **New Baremetal Server** as the Server Type.
4. Enter the following information in the **Create Instance** page:

Table 22: Add Existing Bare Metal Server Information

Field	Action
Instance Name	Enter a name for the BMS instance.
Select Boot Source	Select a Image or Instance Snapshot from the list.
Select Image	Select the BMS Image you created for the BMS from the list.
Select Flavor	Select the Flavor for the BMS from the list.
Select SSH Key	Select the SSH key for the BMS from the list, to login into SSH without password.

Table 22: Add Existing Bare Metal Server Information (continued)

Field	Action
Availability Zone	Assign Availability Zone as <b>nova-baremetal</b> for BMS lifecycle management.
Count (1-10)	Assign values from 1 to 10, to spin the number of BMS instances.

ERROR: Unresolved graphic fileref="" not found in  
"//cmsxml/default/main/supplemental/STAGING/images/".

5. Click **Create** to launch a new baremetal server.

RELATED DOCUMENTATION

<a href="#">Bare Metal Server Management   115</a>
<a href="#">How Bare Metal Server Management Works   119</a>
<a href="#">LAG and Multihoming Support   123</a>
<a href="#">Adding Bare Metal Server to Inventory   125</a>
<a href="#">Onboarding and Discovery of Bare Metal Servers   128</a>
<a href="#">Launching and Deleting a Greenfield Bare Metal Server   130</a>
<a href="#">Troubleshooting Bare Metal Servers   131</a>

# Onboarding and Discovery of Bare Metal Servers

IN THIS SECTION

- [Onboarding of Bare Metal Servers | 129](#)
- [Discovery of Bare Metal Servers | 129](#)

BMS Manager supports onboarding and discovery of bare metal servers.



## Onboarding of Bare Metal Servers

Contrail Networking Release 5.1 supports two types of bare metal servers deployments—greenfield deployments and brownfield deployments.

Greenfield deployments (LCM) are the bare metal servers that have not been deployed and requires to be managed by the BMS manager. These servers do not have an image installed on them. Greenfield servers do not have an IP address assigned.

Brownfield deployments (non-LCM) are the bare metal servers that are already deployed and are in active use by the Tenant users. These servers needs to be added to the Contrail fabric management enrollment. These servers have IP addresses already assigned to them.

## Discovery of Bare Metal Servers

### IN THIS SECTION

- [Manual Discovery | 129](#)
- [Auto Discovery | 129](#)

The Tenant user needs to onboard all bare metal servers that are already provisioned and configured. These bare metal servers are managed by the BMS management framework. The Administrative users and the Tenant users can onboard the servers by automatically discovering the servers or manually registering the servers.

### Manual Discovery

Manual discovery is performed by registering all bare metal servers, their MAC addresses and their physical connectivity manually. This step is described in the section *Administrative Workflow*.

### Auto Discovery

With Contrail Networking Release 5.1, Auto Discovery of all servers can be achieved by utilizing the Ironic Inspector and the DHCP framework. When a server is powered on and physically connected to the TOR device, the DHCP frames are utilized to discover the MAC address as well as the connectivity information. Ironic Inspector uses the MAC address to match existing inventory. If a match is not found, an implicit registration of the server is performed, which is referred to as auto discovery.

## RELATED DOCUMENTATION

[Bare Metal Server Management | 115](#)[How Bare Metal Server Management Works | 119](#)[LAG and Multihoming Support | 123](#)[Adding Bare Metal Server to Inventory | 125](#)[Launching a Bare Metal Server | 127](#)[Launching and Deleting a Greenfield Bare Metal Server | 130](#)[Troubleshooting Bare Metal Servers | 131](#)

## Launching and Deleting a Greenfield Bare Metal Server

This topic describes how to launch a greenfield bare metal server.

1. In the Contrail UI, select **Workloads > Instances > Create Instance**.
2. From the **Server Type** Field, select **New Bare Metal Server**.
3. Select the boot source, BMS image, and the BMS flavor available for the server type selected.
4. Click **Create**.

Following BMS launch, the BMS PXE boots from the ironic-provision network. The ironic-provision network is not visible to the tenant. BMS then connects to the provisioning network, connects to the TSN node, and gets a temporary IP address from the subnet of the provisioning network. This temporary IP address is not visible to the tenant. BMS downloads the boot image from the TFTP server and saves it locally for subsequent local boots. After the BMS is ready, it reboots. This time, the BMS boots from local image. During the second reboot, the BMS is disconnected from the ironic-provision network and is connected to the tenant network. This process of transferring from the ironic-provisioning network to the tenant network is called *Network Flip*. Then, the TSN node provides the BMS an IP address from the tenant network. Once the BMS boots and is ready for use, it is connected to tenant network.

The tenant can delete a BMS when it is not needed in the network. When a BMS is disconnected from the tenant network, it is connected to the cleaning-network or the ironic-provisioning network. This network flip is done to prevent snooping of hackers when the BMS is being cleaned up. The ironic-provisioning network cleans up the server moves it back to the pool of available servers, to be ready for redeployment as a new BMS.

## RELATED DOCUMENTATION

Bare Metal Server Management	115
How Bare Metal Server Management Works	119
LAG and Multihoming Support	123
Adding Bare Metal Server to Inventory	125
Launching a Bare Metal Server	127
Onboarding and Discovery of Bare Metal Servers	128
Troubleshooting Bare Metal Servers	131

## Troubleshooting Bare Metal Servers

This topic provides the steps to troubleshoot BMS.

- **Follow these steps to troubleshoot some of the common issues:**

- Verify that the following objects are created:
  - When the BMS is in provisioning state (when BMS is booting for the first time), there should be two neutron ports—one on provisioning network and another on the tenant network. Run the **openstack port list/show** command to view the list of ports.

The port connected to the provisioning network should have **local\_link\_information** displaying the name of the QFX or TOR and the port to which the bare metal server connected.

- After network flip, only one port should be present. The port connected to provisioning network should be deleted.
- Verify that the logical Interface(s) are created. Run the **curl http://localhost:8082/logical-interfaces** command to view the logical interfaces. The logical interface should point to the correct physical interface.

- **Follow these steps to troubleshoot LAG interfaces (AE interfaces):**

- Ensure that an aggregated Ethernet physical interface is created. Run the **curl http://localhost:8082/physical-interfaces** command to verify. The AE interface name starts with **ae**.
- Ensure that logical Interface is created. Run the **curl http://localhost:8082/logical-interfaces** command. The logical interface should have parent reference pointing to the **ae** physical interface.
- Ensure that a link aggregation group (LAG) is created. Run the **curl http://localhost:8082/link-aggregation-group** command to verify.

- **Follow these steps to troubleshoot multihomed interfaces:**

- Ensure that two logical Interfaces are created. Run the **curl http://localhost:8082/logical-interfaces** command to verify.

Each logical interface should have a parent reference pointing to the physical interface. The Ethernet segment identifier (ESI) should be set to the same value for both physical Interfaces.

- **Follow these steps if you get the error message No Valid Host Found when you launch a BMS server.**

- Run the **openstack baremetal node list/show** command to verify that the nodes are registered on Ironic and are not in error state.
- Run the **openstack baremetal port list/show** command to verify that ports for the nodes are registered.
- Run the **openstack baremetal portgroup list/show** command to verify that the port groups (in case of LAG/MH deployments).
- Run the **openstack flavor list/show** command to verify the BMS flavors details to ensure that the flavor matches with the node specification.
- Review the **api-server** logs for errors. The log contains errors of there is a duplicate MAC address or the physical interface is not configured.
- Review the **ironic-conductor** logs for errors. For example, **PXE\_ENABLED port is not found**.

- **Follow these steps if the server does not boot or if the server remains in boot state:**

- Verify whether the server is assigned an IP address on the provisioning network.
  - If an IP address is not assigned, verify whether the TSN node is reachable.
  - If an IP address is assigned, check whether the TFTP boot server is reachable.

In either case, you can use the **tcpdump** tool to review the TCP packets to check whether the bare metal server can reach these servers.

- Follow these steps if the server was assigned an IP address and is booted on provisioning network, but remains the same state. That is, network flip does not happen.
  - Verify the **ironic-conductor** logs to see whether Ironic Python Agent (IPA) on the bare metal server is able to communicate with Ironic Conductor.
  - Check whether the image was built correctly with the correct IPA.

## RELATED DOCUMENTATION

[Bare Metal Server Management | 115](#)

[How Bare Metal Server Management Works | 119](#)

[LAG and Multihoming Support | 123](#)

[Adding Bare Metal Server to Inventory | 125](#)

Launching a Bare Metal Server | 127

---

Onboarding and Discovery of Bare Metal Servers | 128

---

Launching and Deleting a Greenfield Bare Metal Server | 130