JUNIPER NETWORKS | Engineering Simplicity

Configuration Essentials: QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250

Published
2026-07-27

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

Configuration Essentials: QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250

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 this Guide

This guide provides an overview of the core features, initial system configuration procedures, interface setup, diverse use cases, and licensing requirements of the Juniper Networks® QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches. Use this guide to understand the key features of the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches and configure basic system features and common use cases. From this guide, you can easily navigate to other Junos OS Evolved documents that contain more in-depth information about these QFX Series Switches.

At the end of this guide, you should be able to identify the most suitable switch for your network requirements.

NOTE: This guide provides focused, deployment-ready information for the initial setup and configuration of the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches. The guide is designed to complement, and not replace any QFX Series Switch hardware guide, associated software configuration guides, and the product release notes.

What Do You Want To Do?

Table 1: Top Tasks

If you want to...	See:
Learn About QFX Series Switches and their positioning and data packet processing *The QFX Series Switches are designed for high-performance, low-latency, and scalable fabric deployments. These switches use a modular Junos OS architecture that allows the control plane and data plane to run in parallel, maximizing data transition efficiency and overall switch performance. In this topic, you'll also learn how data transits in these switches before it exits the egress pipeline.*	"Introducing QFX Series Switches" on page 2

Table 1: Top Tasks *(Continued)*

If you want to...	See:
Manage QFX Series Switches *Managing the QFX Series Switches involves hardware setup, software configuration, and ongoing maintenance using Junos OS Evolved and management interfaces. In this topic, you'll learn to manage these switches using Junos OS Evolved CLI and Juniper Apstra.*	"QFX Series Switches Management" on page 6
Perform Initial System Configuration *Before getting started with any task on a QFX Series Switch, you need to log in to that switch and perform initial configuration using the Junos OS Evolved CLI. In this topic, you'll learn to log in to your device, create user accounts, authenticate those user accounts, and configure syslog, SNMP, and telemetry services.*	Device Login and Initial Configuration
	No Link Title
	User Account and Authentication
	Logging Syslog, SNMP, and Telemetry
Learn About Interface Types and Configuration *In this topic, you'll learn about the built-in interface types, breakout interface options, some basic interface configurations, and transceivers types.*	Interface Types and Configurations
	"Connectivity and Transceivers Types" on page 42
Configure Key Features *Learn more about key features such as traffic load balancing, flexible match firewall filter, class of service, Junos telemetry, ISSU, and so on.*	"BGP Auto-Discovered Neighbor (BGP Unnumbered Peering)" on page 46
	"Class of Service" on page 49
	"EBGP for the Overlay" on page 54
	"ESI-LAG" on page 56

Table 1: Top Tasks *(Continued)*

If you want to...	See:
	"Flexible Firewall Match Filter" on page 58
	"IBGP using IPv4 Overlay" on page 60
	"IRB Anycast" on page 63
	"In-Service Software Upgrade" on page 66
	"EVPN-VXLAN MAC-VRF Routing-Instance Type" on page 69
	"RDMA over Converged Ethernet version 2" on page 72
	"Route Type 5 IP Routing Instance" on page 75
	"Secure Boot" on page 77
	"Symmetric IRB using RT2 (MAC-IP)" on page 78
	"Traffic Load Balancing" on page 80
Learn About Use Cases Applicable to your Switch *Learn about the switch's use cases and the topology and configuration for each use case.*	"Use Case Index" on page 84
Know About the Licensing and Upgrade Information *Read this topic to find solutions to common licensing and upgrade scenarios that you can manage yourself.*	Junos Licensing and Upgrade Info

Table 1: Top Tasks *(Continued)*

If you want to...	See:
Know More About the Documentation Tools and Resources *Use this topic to learn more about QFX Series Switches through different tools such as Hardware Compatibility Tool, CLI Explorer, Port Checker, and so on.*	Related Documentation Applications and Resources

Documentation Conventions

For information about the notice icons and text and syntax conventions used in the documentation, see Documentation Conventions.

CHAPTER 1

Introduction

IN THIS CHAPTER

Overview

SUMMARY

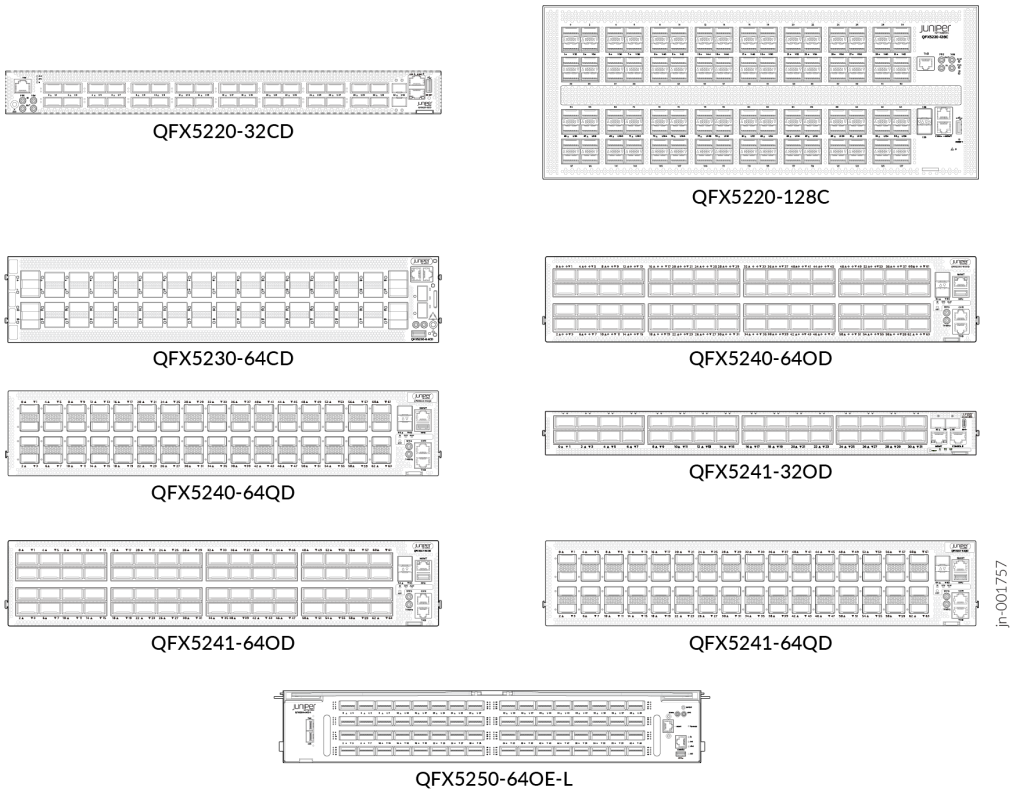
This section provides a collective overview, device positioning, and data packet processing architecture of the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches. In this section, you can also navigate to the device-specific documentation pages by clicking the device name link provided in the introduction.

IN THIS SECTION

Juniper Networks® QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches deliver high performance for data center deployments. They support leaf, spine, and super-spine IP fabrics; advanced Layer 2, Layer 3, and MPLS features; low latency (750 ns), and bidirectional throughput of up to 102.4 Tbps. These switches run Junos OS Evolved and provide adaptable port configurations suitable for cloud environments. The switches support capabilities such as EVPN-VXLAN, automation, and energy-efficient power usage per Gbps.

These switches support a range of Ethernet speeds, enabling data centers to integrate updated servers and storage technologies seamlessly, without the need for disruptive redesigns. For more information about the QFX Series Switches, see the Switching section in QFX Series Switches Documentation.

Figure 1: QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches



Positioning of QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches

The key market positioning elements for the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches include:

- **Leaf-spine architecture:** Optimized for leaf-spine architectures that allow horizontal scaling. This means you can add more leaf or spine switches rather than redesigning the network.
- **Suitable for modern data centers:** Intended for data center environments and IP fabric architectures.
- **Advanced network virtualization and automation:** Supports EVPN-VXLAN, Juniper's recommended overlay technology for large data center fabric that provides massive scalability, fast convergence, and efficient multitenant isolation.

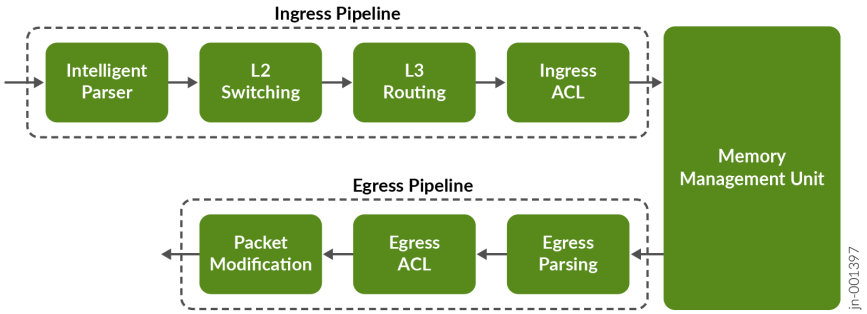
> **NOTE**: This feature is not supported on the QFX5250.

- **Unified Junos OS experience:** Provides a consistent operating model across the fabric that reduces configuration drift and speeds up deployment and troubleshooting.
- **High availability and reliability:** Minimizes application response time and avoids latency spikes under load.

Data Packet Processing Architecture

The data packet transit process for all the QFX Series Switches is almost the same. When a data packet enters any QFX Series Switch, it passes through different blocks or levels as shown in Figure 2 on page 4. At a high level, the data packet transits through three stages: ingress pipeline, memory management unit (MMU), and egress pipeline.

Figure 2: Transit Data Packet Processing



This figure shows that a data packet is processed through a switch's ingress pipeline, memory management unit, and egress pipeline. The role of each functional block is as follows:

- The ingress pipeline consists of four blocks: Intelligent Parser, L2 Switching, L3 Routing, and Ingress ACL.
  - The Intelligent Parser analyzes incoming packets, identifies L2 and L3 headers, determines whether the ingress port is VLAN-enabled or routing-enabled, and converts parsed information into metadata for downstream processing.
  - The L2 Switching block performs VLAN membership and tagging checks, learns source MAC addresses, and looks up destination MAC addresses. Known unicasts are forwarded to the appropriate egress port, while unknown unicasts are flooded within the VLAN. If the ingress port is not VLAN-enabled, then the packets bypass this block.
  - The L3 Routing block processes packets received on routing-enabled ports. It verifies whether the packet is destined for the device and performs routing table lookups. Packets are either forwarded to the destination IP or dropped; flooding is not performed.

  - The Ingress ACL block applies policy actions such as packet filtering, field rewriting, rate limiting, counting, redirection, and security enforcement based on fields parsed earlier.

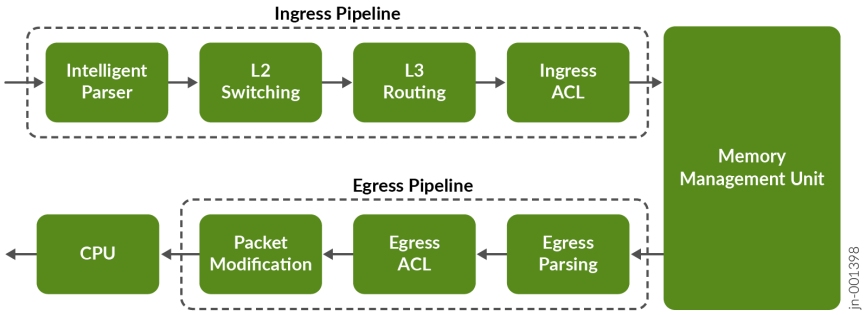
- The **MMU** connects the ingress and egress pipelines. Packets written by the ingress pipeline are stored in memory and scheduled—based on priority—by an internal scheduler before being forwarded to the egress pipeline.
- The egress pipeline consists of the Egress Parser, Egress ACL, and Packet Modification blocks.
  - The Egress Parser block re-parses packet headers and generates metadata for egress processing.
  - The Egress ACL block applies egress-side policies using packet header and port information.
  - The Packet Modification block updates fields such as destination MAC addresses and IP header information before transmitting the packet out of the front-panel port.

Non-Transit Data Packet Processing

If a transit packet fails, non-transit packets traverse the three stages in the same way as transit packets, except for one difference (see Figure 3 on page 5).

When a data packet exits the egress pipeline, it is delivered to the CPU. Before that, during the ingress pipeline, the Intelligent Parser examines the packet to extract the information the CPU needs. If the packet does not contain the required information, the packet is dropped; otherwise, it continues through the egress pipeline to reach the CPU.

Figure 3: Non-Transit Data Packet Processing



Management Options

SUMMARY

This section describes different management options for the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

IN THIS SECTION

You can manage QFX Series Switches using the command-line interface (CLI) and Juniper Apstra.

Junos OS Evolved CLI

The QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 switches support the Junos OS Evolved CLI. You can use Junos OS Evolved for the initial configuration of the switches, such as root authentication and management interface, and for advanced configurations, such as ROCEv2, ESI-LAG, and so on.

Juniper Apstra Data Center Director

For remote configuration and monitoring purposes, you can use Juniper Apstra Data Center Director, which is a data center automation and management platform. Using Apstra, you can automate the network design, deployment, and operation of data center fabrics. Apstra uses agent profiles to manage the QFX Series Switches. For more information about how to onboard data center switches, see Onboarding Data Center Switches with Apstra and Onboarding Devices into Apstra.

QFX Series Switches Licensing and Junos Upgrade

SUMMARY

This section provides information about licensing and Junos OS Evolved upgrade for QFX Series Switches.

IN THIS SECTION

Licensing Information

When you buy any QFX Series Switch, the sales executive guides you with the licensing options available and suggests the most suitable option based on your business requirements. Once you have a license key, see Juniper Licensing User Guide for instructions to activate the license key on your platform.

If your QFX Series Switch encounters an issue and requires a replacement, you need not buy a new license. Instead you can use the same license key on the replaced platform. This information is available in the Juniper Licensing User Guide.

Your license will be valid for a certain number of years and you may need to renew the license at the end of this period. For license renewal information, see How to renew platform device license?

Upgrade Junos OS Evolved

By default, all the Juniper platforms have built-in Junos OS Evolved software. To upgrade to the latest Junos OS Evolved release that supports your platform device, see Upgrading Junos OS and Upgrading Junos OS Evolved.

2

CHAPTER

System Configuration

IN THIS CHAPTER

Device Login and Initial Configuration

SUMMARY

This section describes how to perform initial configuration on your QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches by using the Junos OS Evolved CLI. Initial configuration includes root login to the device, setting the root authentication password, assigning the device name, enabling remote access, configuring the IP address for management interface, setting the date and time, and configuring DNS details.

IN THIS SECTION

Overview

When you buy any network device, it comes with the factory-default configuration (basic device configuration). This factory-default configuration loads automatically the first time you install the device and power it on.

In this section, you'll learn to customize the factory-default configuration on a QFX Series Switch by using CLI commands. You'll learn how to connect to your switch using the console port and perform an initial configuration to get your device up and running. You can always revert to the factory-default configuration whenever you want. For more information about restoring the default factory configuration, see Reverting to the Default Factory Configuration.

Before You Begin

Gather the following items or information before configuring the switch:

- Laptop or a desktop PC
- RJ-45, USB-A, and USB-C cable
- Name of the switch that you will use on the network
- Password for the root user
- Domain name the switch will use
- For Out Of Band management, IP address and prefix-length information for the Ethernet interface
- IP address of the default gateway

- IP address of the DNS server

NOTE: For information about how to connect your QFX Series Switch to the console port, see the Switching section in the QFX Series documentation.

Perform Initial Device Configuration

To perform initial software configuration on the QFX Series Switch:

1. Connect the console port of your switch to a laptop or a desktop PC using the RJ-45 cable. By default, console port access is enabled on your device.

NOTE: If your laptop or desktop PC does not have a serial port, use a serial-to-USB adapter.

2. Power on the switch and wait for it to boot. The software boots automatically. When the boot process is complete, the **login:** prompt on the console appears.

```
login:
```

3. At the login prompt, type **root** to log in.

NOTE: Initially, you do not need a password for the root user account. The device prompt `root@%` indicates that you are the root user.

```
login: root
```

4. Type **cli** to start the Junos OS Evolved CLI.

```
root@% cli
root@<hostname>#
```

5. Type **configure** to access the CLI configuration mode.

```
root@<hostname># configure
[edit]
root@<hostname>#
```

6. To prevent automatic software update, stop the chassis auto-upgrade process.

> **NOTE**: Until you assign a root password and perform an initial commit, you might see zero-touch provisioning (ZTP)-related messages on the console. You can safely ignore these messages while you configure the root password.

```
delete chassis auto-image-upgrade
```

7. Stop ZTP.

> **NOTE**: ZTP is enabled on the QFX Series Switches by default. You must stop ZTP before you configure any settings.

```
delete system commit factory-settings
```

8. Set the root authentication password and commit.

```
set system root-authentication plain-text-password
New password: password
Retype new password: password
```

9. Configure the name of the switch. If the name includes spaces, enclose the name in quotation marks (“ ”).

```
set system host-name host-name
```

10. Commit the configuration, and wait for the ZTP process to stop. Learn how to protect network security by configuring the root password.

```
commit
```

11. Enable remote access. Remote management access to the device and all management access protocols—such as Telnet, FTP, and SSH—is disabled by default.

    - To enable SSH service:

      ```
      set system services ssh root-login allow
      ```

      This command allows root login through SSH. You must configure other users' profiles separately.

    - To enable Telnet service, if required:

      ```
      set system services telnet
      ```

      NOTE: When Telnet is enabled, you cannot log in to the QFX Series Switch using root credentials. Root login is allowed only for SSH access.

12. Configure the IP address and prefix length for the management interface on the switch.

    NOTE: For more information about how to connect to the management port, see the Switch Management Cable Specifications and Pinouts section in the hardware guide.

    ```
    set interfaces re0:mgmt-0 unit 0 family inet address ip-address/prefix-length
    ```

13. Set the date and time on a device manually or use an NTP server to synchronize the date and time.

    - From the operational mode, manually set the date and time.

      ```
      run set date YYYYMMDDhhmm.ss
      ```

- In case of NTP server, the date and time syncs automatically. Run the following command to configure the NTP server.

```
set system ntp server ntp-server
run show system uptime | match Current
run show system status
```

14. Configure the domain name and DNS servers on your switch at the **system** hierarchy level. This allows hostname resolution and full identity through DNS.

```
set system domain-name domain
set system name-server ip-address
```

15. Commit the configuration. This step completes the minimum Day-0 configuration.

```
commit
```

16. Save the configuration and exit. At this point, the switch is reachable through the management IP address (SSH) and ready for further Day-1 configuration.

```
exit
```

How to Recover the Root Password for Junos OS Evolved

IN THIS SECTION

This procedure resets the root password without resetting the device configuration to the factory default configuration. Only the root password is reset. This procedure does not affect the device state or any other function.

Video: How to Recover the Root Password in Junos OS Evolved

How to Connect to the Serial Port

The first task in the password reset operation is to connect to the serial port of the device.

To connect to the serial port:

1. Power off the router by pressing the power button on the front panel.
2. Turn off the power to the management device (usually a computer) that you will use to access the CLI.
3. Plug one end of the Ethernet rollover cable supplied with the router into the RJ-45 to DB-9 serial port adapter supplied with the router.
4. Plug the RJ-45 to DB-9 serial port adapter into the serial port on the management device.
5. Connect the other end of the Ethernet rollover cable to the console port on the router.
6. Turn on the power to the management device.
7. On the management device, start your asynchronous terminal emulation application (such as Microsoft Windows Hyperterminal), and select the appropriate COM port to use (for example, COM1).
8. Configure the port settings as follows:
   - Bits per second: 9600
   - Data bits: 8
   - Parity: None
   - Stop bits: 1
   - Flow control: None
9. Power on the router by pressing the power button on the front panel.

   Verify that the POWER LED on the front panel turns green. The terminal emulation screen on your management device displays the router's boot sequence.

How to Recover the Root Password

The password reset operation is triggered early in the boot process. You reset the password in the shell.

To recover the root password:

1. Do a hard reboot of the Routing Engine (that is, shut down the OS, power off the device, and then power on the device).

   On the terminal, you will see the following screen:

```
GNU GRUB  version 2.02~beta3

+----------------------------------------------------------------------+
|*Primary ptx-fixed-19.1-16                                            |
| Primary [Recover password]                                           |
| Primary-Rollback ptx-fixed-19.1-15                                   |
| Primary-Rollback [Recover password]                                  |
|                                                                      |
|                                                                      |
|                                                                      |
|                                                                      |
|                                                                      |
|                                                                      |
|                                                                      |
|                                                                      |
+----------------------------------------------------------------------+

     Use the ^ and v keys to select which entry is highlighted.
     Press enter to boot the selected OS, `e' to edit the commands
     before booting or `c' for a command-line.
```

2. Use the arrow keys to scroll down to the `Primary [Recover password]` option, and press Enter.

```
Disk boot ...
IMA is 0
Loading kernel ...ok.
Loading initrd ...ok.
Booting ...
[    0.791088] Failed to find cpu0 device node
Processing /dev/sda2 for mount on /soft ...[checking]..ok [mounting]..done
Processing /dev/sda5 for mount on /data ...[checking]..ok [mounting]..done
```

```
Processing /dev/sda6 for mount on /data/config ...[checking]..ok [mounting]..done
Processing /dev/sda7 for mount on /data/var ...[checking]..ok [mounting]..done
mkswap: /dev/sda3: warning: wiping old swap signature.
Setting up swapspace version 1, size = 4 GiB (4294963200 bytes)
no label, UUID=7d905478-773b-41e5-8f1d-8166ec03f93a
Processing /dev/sda1 for mount on /boot ...[checking]..ok [mounting]..done
Done with local filesystems setup.
Mounting version junos-linux-install-ptx-x86-64-16.2I20180502181332_evo-builder...done.
System is running with minimal count of software versions
modprobe: FATAL: Module jnx_cbd_fpga_ptx21k not found in directory /lib/modules/4.8.24-
WR2.2.1_standard
Installing kexec kernel...Cannot get kernel page_offset_base symbol address
done
Password recovery in progress. Please enter new password.
```

3. Enter the new password, and then retype the new password and press Enter.

```
New password:
Retype new password:
passwd: password updated successfully
Password recovery done

Welcome to Linux!
```

The reboot proceeds until the login prompt appears.

```
[  OK  ] Started Serial Getty on ttyS0.
[  OK  ] Reached target Login Prompts.
[  OK  ] Started Vsftpd ftp daemon.
[  OK  ] Started Network Time Service.
[  OK  ] Started strongSwan IPsec IKEv1/IKEv2 daemon using swanctl.
[  OK  ] Started Arp filtering arptables.
[  OK  ] Started Management Ethernet Interface Manager Service.
[  OK  ] Started OFP on RE.
         Starting MGD initialization of schema and database on RE...

Juniper Linux Distribution 2.2.1 re0 ttyS0

re0 login:
```

4. At the prompt, log in as root using the new password.

You will see a shell prompt.

```
Last login: Mon May  7 13:09:08 PDT 2018 from xxxx
--- JUNOS builder Linux re0 4.8.24-WR2.2.1_standard #1 SMP PREEMPT Mon Apr 9 13:21:32 PDT
2018 x86_64 x86_64 x86_64 GNU/Linux
root@RE0:~#
```

5. To start the CLI, enter `cli`.

User Account and Authentication

SUMMARY

This section describes how to create user accounts and define authentication methods for the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches. This configuration helps define who can log in to the device, how to verify their authenticity, and what actions these users are authorized to perform on the device.

IN THIS SECTION

A user account is an identity that allows a user to access a particular device and includes a username, password (or authentication method), and login class (permission level). For more information, see User Accounts.

Authentication is the process of verifying the user's identity before granting access. For more information, see User Authentication Overview.

QFX Series Switches include predefined login classes such as operator, read-only, superuser, and unauthorized. For more information, see Login Classes.

Local User Account

Initially, when you start any QFX Series Switch, it defaults to a root user with no password option. However, you are not allowed to do any configuration changes at the root level.

To make any configuration changes, you need an account with a password. This is the simplest authentication method, where the username and password are stored locally on the switch. You can

create a user and define permissions and system access using login classes. For more information, see User Accounts and Login Classes.

Example: To create an admin user:

```
set system login user admin class super-user
set system login user admin authentication plain-text-password
New password:
Retype new password:commit
```

RADIUS Authentication

This type of authentication uses a centralized server to authenticate users that attempt to access a network device. This method is commonly used in telecom networks. For more information, see RADIUS Authentication.

To configure a RADIUS server:

```
set system authentication-order radius
set system authentication-order password
set system radius-server server-address secret "MyRadiusSecret"
set system radius-server server-address source-address ip-address
```

In the above code snippet, configure authentication order, then add `system radius-server` with a secret (and optionally `source-address`, `interface`, `timeout`, `retries`, etc.). Try the RADIUS method of authentication first, and if that fails, use the local user authentication method.

TACACS+ Authentication

This type of authentication is an alternate method of authenticating users that attempt to access a network device. TACACS+ provides authentication and command authorization. This authentication method is often used in enterprise networks. For more information, see TACACS+ Authentication.

To configure a TACACS+ server:

```
set system authentication-order tacplus
set system authentication-order password
set system tacplus-server server-address secret "password"
set system tacplus-server server-address source-address source-address
```

Of all the authentication methods, the system will first try the TACACS+ authentication. If it fails, the system tries the RADIUS server authentication method. If both authentication methods fails, the system tries the local user authentication method.

Logging, SNMP, and Telemetry

SUMMARY

Learn to enable system logging, SNMP, and telemetry services on the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches in your network.

IN THIS SECTION

System Logging (Syslog)

You can configure system logging (syslog) to maintain network stability, security, and performance. The syslog configuration enables network administrators to monitor, troubleshoot, and audit the device activities.

To configure syslog locally on a switch:

```
set system syslog file filename any notice
set system syslog file filename authorization info
```

To configure remote syslog server (sending logs to an external syslog server):

```
set system syslog host syslog-server-ip-address any notice
set system syslog host syslog-server-ip-address structured-data
```

To configure the source IP address used for syslog traffic:

```
set system syslog host ip-address source-address source-ip-address
```

SNMP

SNMP monitor network devices such as switches, routers, and other IP-based devices from a single management host. By default, SNMP is not enabled on a QFX Series Switch. However, the operating system running on these switches, Junos OS Evolved, supports SNMPv1, SNMPv2c, and SNMPv3.

To enable SNMP, you need to add the configuration statements at the `[edit]` hierarchy level. The minimum configuration you can enable for SNMP is SNMP polling.

To define an SNMP community and set its permissions:

```
set snmp community community-name authorization authorization-value
```

For example:

```
set snmp community C1 authorization read-only
```

```
set snmp community C1 authorization read-write
```

To configure basic SNMP identity:

```
set snmp contact contact
set snmp location location
```

To limit SNMP queries to trusted management ports:

```
set snmp community community-name authorization read-only
set snmp community community-string clients mgmt-host-ip-address/prefix
```

For example:

```
set snmp community "noc_ro" authorization read-only
set snmp community "noc_ro" clients 192.0.2.10/32
```

To configure SNMP traps:

```
set snmp community community-name authorization authorization-value
set snmp trap-group trap-group-name version version-number
```

```
set snmp trap-group trap-group-name targets ip-address community community-name
set snmp trap-group trap-group-name categories link
```

For example:

```
set snmp community "noc_ro" authorization read-only
set snmp trap-group "nms" version v2
set snmp trap-group "nms" targets 192.0.2.10 community "noc_ro"
set snmp trap-group "nms" categories link
```

To verify whether SNMP is running after configuration, run the following command in operational mode:

```
run show snmp statistics
```

Telemetry

Junos telemetry is a telemetry solution developed to stream telemetry data from a Junos device. The QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 switches support Junos telemetry to stream real-time network data, such as traffic patterns, device status, error rates, and other metrics that provide insights into the network's health and behavior. For more information about Junos telemetry, see Understanding Junos Telemetry. For information about how to configure gRPC and verify telemetry streaming, see Understanding Authentication and Authorization for gRPC-Based Services.

Read configure mutual (bidirectional) authentication for gRPC services to learn how to verify that your QFX Series switches are running Junos OS Evolved and that a network device or a virtual machine running Linux can reach one of the traffic ports on the QFX Series switch. When mutual authentication is configured:

- The server provides its public key certificate after establishing the channel.
- The client uses the server's root CA certificate to authenticate the server.
- The client also provides its certificate when it connects to the server, and the server validates the certificate. If the certificate validation is successful, the client is allowed to make calls.

Perform the following steps to configure and verify telemetry streaming on the QFX Series switches using gNMI.

1. To obtain the X.509 Certificates (server root certificate authority and server key pairs):

```
openssl genrsa -out server.key 4096
openssl req -new -x509 -sha256 -key serverRootCA.key -out serverRootCA.crt -days 3650 -subj
```

```
"/C=US/ST=CA/L=Sunnyvale/O=Juniper/CN=serverRootCA"
openssl genrsa -out server.key 4096
openssl req -new -key server.key -out server.csr -sha256 -subj "/C=US/ST=CA/L=Sunnyvale/
O=Juniper/OU=serverRootCAOrg/CN=vJunosEvolved"
openssl x509 -req -in server.csr -CA serverRootCA.crt -CAkey serverRootCA.key -
CAcreateserial -out server.crt -days 365 -extfile server_ssl_cert_ext.cnf
```

NOTE: The gRPC server certificate must define either the server hostname in the Common Name (CN) field (the sample command above, vJunosEvolved is the server hostname), or the server IP address in the subjectAltName IP address field in the `server_ssl_cert_ext.cnf` file (shown below). The client application must use the same value to establish the connection to the server. If the certificate defines the subjectAltName IP address field, the CN field is ignored during authentication.

The `server_ssl_cert_ext.cnf` file content is as follows:

```
cat server_ssl_cert_ext.cnf
#### include -extfile server_ssl_cert_ext.cnf in command to include extensions file
#### openssl.cnf
extensions = v3_sign
[v3_sign]
subjectAltName=IP:172.25.11.11
```

To create a private certificate authority, and then generate and sign a client certificate for TLS use a Juniper-related services like NETCONF/gRPC:

```
openssl genrsa -out clientRootCA.key 4096
openssl req -new -x509 -sha256 -key clientRootCA.key -out clientRootCA.crt -days 3650 -subj
"/C=US/ST=CA/L=Sunnyvale/O=Juniper/CN=clientRootCA"
openssl genrsa -out client.key 4096
openssl req -new -key client.key -out client.csr -sha256 -subj "/C=US/ST=CA/L=Sunnyvale/
O=Juniper/OU=serverRootCAOrg/CN= client-desktop"
openssl x509 -req -in client.csr -CA clientRootCA.crt -CAkey clientRootCA.key -
CAcreateserial -out client.crt -days 365 -extfile server_ssl_cert_ext.cnf
```

2. To enable gRPC service:

```
set system services http servers server grpc-server port 32767
set system services http servers server grpc-server grpc gnoi
```

```
set system services http servers server grpc-server grpc gnmi
set system services http servers server grpc-server tls local-certificate grpc-server
```

To configure mutual authentication instead of server-only authentication:

```
set system services extension-service request-response grpc ssl port 32767
set system services extension-service request-response grpc ssl local-certificate grpc-
server
```

To configure authentication for the gRPC client directly (in the network device configuration):

```
set system services extension-service request-response grpc ssl mutual-authentication
certificate-authority grpc-client-CA
set system services extension-service request-response grpc ssl mutual-authentication
client-certificate-request require-certificate-and-verify
set system services extension-service request-response grpc ssl hot-reloading
set system services extension-service request-response grpc ssl use-pki
set system services extension-service traceoptions file jsd
set system services extension-service traceoptions flag all
set security pki ca-profile grpc-client-CA ca-identity clientRootCA
set security pki traceoptions file size 10m
set security pki traceoptions file files 3
```

3. To copy the certificates that you have generated on the client to the gRPC server:

```
scp -O server.crt lab@172.25.11.11:/var/home/lab
scp -O server.key lab@172.25.11.11:/var/home/lab
scp -O clientRootCA.crt lab@172.25.11.11:/var/home/lab
```

4. To load the certificate on the server:

```
request security pki local-certificate load certificate-id grpc-server filename /var/
home/lab/server.crt key /var/home/lab/server.key
request security pki ca-certificate load ca-profile grpc-client-CA filename /var/home/lab/
client.crt
```

5. To configure the user account for gRPC services:

```
set system login user gnoi-user class super-user
set system login user gnoi-user authentication plain-text-password
set system login user gnoi-user full-name "gNOI client"
```

6. Identify the information that you want to receive from the QFX Series switch. Information you want to stream through Junos telemetry is specified using a telemetry sensor path. A telemetry sensor path is the hierarchical path (defined using YANG) to the operational data or metrics to be monitored.

   For example, the following sensor path streams administrative and operational status information for interfaces on the device:

```
/junos/system/linecard/interface
```

   For more information about telemetry sensor paths, see Explore Sensor Paths. To view all the supported sensor paths, their corresponding leaf nodes, and the device platforms, see Junos YANG Data Model Explorer.

7. Download, install, and configure the gNMI client to test the telemetry streaming on your QFX Series switch. Junos OS Evolved-based devices support various subscription types. See Subscription Types and subscription mode to identify the subscription type and mode for your network.

8. To test the telemetry streaming on your Linux machine and subscribe to telemetry data using the gNMI client:

```
gnmic sub -a <device_ip>:<gnmi_port> -u <username> -p <password> --format json subscribe --
path <sensor path> --mode stream --stream-mode sample --sample-interval <seconds>
```

   where,

   - `format`: Specifies the output format used by the gNMI client to display telemetry data. The `json` format presents telemetry data in JSON. Available options include JSON, BYTES, PROTO, ASCII, and JSON_IETF.
   - `sub`: Invokes the gNMI subscribe RPC to establish a telemetry subscription with the device.
   - `path`: Is the YANG sensor path to stream telemetry data.
   - `mode`: Defines the subscription mode. The stream mode establishes a continuous subscription that sends telemetry updates over time.

- `stream-mode`: Specifies the streaming behavior for a stream subscription. The sample mode sends updates at regular intervals.
- `sample-interval`: Sets the sampling interval for stream subscriptions when `stream-mode` sample is configured.

In the following example, "stream" is selected as subscription type and "sample" as subscription mode.

```
gnmic sub -a 172.25.11.11:32767 -u gnoi-user -p gnoi123 --tls-ca serverRootCA.crt --tls-
cert client.crt --tls-key client.key --format json subscribe --path /junos/system/linecard/
interface --mode stream --stream-mode sample --sample-interval 10s
```

9. Verify that the telemetry data is received on the collector. A successful output confirms that the gNMI connection is established and the telemetry data is streaming from the QFX Series switch.

   Sample gNMI Telemetry Output (JSON format)

```
{

  "source": "172.25.11.11:32767",

  "subscription-name": "default-1782391912",

  "timestamp": 1782391912042548677,

  "time": "2026-06-25T05:51:52.042548677-07:00",

  "prefix": "interfaces/interface[name=et-0/0/0]",

  "updates": [

    {

      "Path": "name",

      "values": {

        "name": "et-0/0/0"

      }
```

```
    },
    {
      "Path": "state/hardware-port",
      "values": {
        "state/hardware-port": "FPC0:PIC0:PORT0"
      }
    },
    {
      "Path": "state/transceiver",
      "values": {
        "state/transceiver": "FPC0:PIC0:PORT0:Xcvr0"
      }
    },
    {
      "Path": "state/physical-channel",
      "values": {
        "state/physical-channel": {
          "element": [
            {
              "Value": {
                "JsonVal": "MA=="
              }
```

```
          },
          {
            "Value": {
              "JsonVal": "MQ=="
            }
          },
          {
            "Value": {
              "JsonVal": "Mg=="
            }
          },
          {
            "Value": {
              "JsonVal": "Mw=="
            }
          }
        ]
      }
    }
  },
  {
```

```
      "Path": "name",
      "values": {
        "name": "et-0/0/1"
      }
    },
    {
      "Path": "state/hardware-port",
      "values": {
        "state/hardware-port": "FPC0:PIC0:PORT1"
      }
    },
    {
      "Path": "state/transceiver",
      "values": {
        "state/transceiver": "FPC0:PIC0:PORT1:Xcvr0"
      }
    },
    {
      "Path": "state/physical-channel",
      "values": {
        "state/physical-channel": {
          "element": [
```

```
    {
      "Value": {
        "JsonVal": "MA=="
      }
    },
    {
      "Value": {
        "JsonVal": "MQ=="
      }
    },
    {
      "Value": {
        "JsonVal": "Mg=="
      }
    },
    {
      "Value": {
        "JsonVal": "Mw=="
      }
    }
  ]
```

```
        }

      }

    },



… truncated
```

A continuous stream of telemetry data indicates that gNMI telemetry streaming is operating correctly on the QFX Series switches.

10. After verifying your telemetry setup, install one of the many available open-source or third-party collectors.

NOTE: Setup and configuration of these collectors is beyond the scope of this guide.

CHAPTER 3

Interface Configuration

IN THIS CHAPTER

Interface Types and Configuration

SUMMARY

This section describes built-in interface types, breakout interface options, and various commands to configure interfaces on your QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

IN THIS SECTION

Overview

The QFX Series Switches are designed to accommodate a comprehensive array of high-density, high-speed Ethernet interfaces, including 10GbE, 25GbE, 40GbE, 50GbE, 100GbE, 200GbE, 400GbE, 800GbE, and up to 1600GbE interfaces, subject to the specific model. These switches feature QSFP28/QSFP-DD ports that enable channelization and support breakout cables (such as 2x50GbE, 2x100GbE, 2x200GbE, 2x400GbE, 8x50GbE, 8x100GbE, and 1x1600GbE), offering versatile connectivity solutions suitable for data center environments.

The "built-in" on page 32 and "breakout" on page 34 interface options offer significant flexibility in how bandwidth is provisioned, enabling you to optimize port utilization, scale capacity incrementally, and support a mix of access, aggregation, and uplink requirements across diverse deployment scenarios.

Built-In Interface Options

Table 1 on page 33 provides information about the built-in interface speeds on the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

Table 2: Built-In Interface Speeds

Switch	Port Density (Without Breakout)	Management and Control Interfaces
QFX5220	**QFX5220-32CD:** 32 400GbE QSFP56-DD ports **QFX5220-128C:** 128 100GbE QSFP28 ports	• Two SFP+ ports that support 10-Gbps or 1-Gbps speed • One RJ-45 Ethernet management port (100/1000/10000 Mbps) • One RJ-45 console port • One USB port (local storage or image handling)
QFX5230	64 400GbE QSFP56-DD ports	• Two SFP+ ports supports 10-Gbps or 1-Gbps speed • One RJ-45 Ethernet management port (100/1000/10000 Mbps) Similar to QFX5220, two SFP+ ports support 10-Gbps or 1-Gbps speed • One RJ-45 console port • One USB port
QFX5240	**QFX5240-64OD:** 64 800GbE OSFP ports **QFX5240-64QD:** 64 800GbE QSFP56-DD ports	• One RJ-45 Ethernet management port (100/1000/10000 Mbps) Similar to QFX5220, two SFP+ ports support 10-Gbps or 1-Gbps speed. • One RJ-45 console port • One USB port

Table 2: Built-In Interface Speeds *(Continued)*

Switch	Port Density (Without Breakout)	Management and Control Interfaces
QFX5241	**QFX5241-32OD:** 32 800GbE OSFP ports **QFX5241-64OD:** 64 800GbE OSFP ports **QFX5241-64QD:** 64 800GbE QSF56-DD ports	• One RJ-45 Ethernet management port (100/1000/10000 Mbps) Similar to QFX5220, two SFP+ ports support 10-Gbps or 1-Gbps speed. • One RJ-45 console port • One USB port
QFX5250	**QFX5250-64OE-L:** 64 1600GbE OSFP ports	• One RJ-45 Ethernet management port (100/1000/10000 Mbps) • One RJ-45 console port • One Type A, USB 3.0 port (SS) • Two SFP+ ports support 10-Gbps or 25-Gbps speed

Breakout Interface Options

The QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches support breakout capability. With this capability, you can split a high-capacity optical link into several lower-capacity links that help utilize the bandwidth at its best and physical infrastructure in different networking situations.

Table 3: Breakout Interface Options

Switch	Breakout Options
QFX5220-32CD	• 2 × 200GbE • 4 × 100GbE • 2 × 100GbE • 8 × 50GbE • 2 × 50GbE • 4 × 25GbE • 4 × 10GbE
QFX5220-128C	• 4 × 10GbE • 4 × 25GbE • 2 × 50GbE
QFX5230	• 2 × 200GbE • 4 × 100GbE • 3 × 100GbE • 2 × 100GbE • 2 × 50GbE • 4 × 25GbE • 4 × 10GbE

Table 3: Breakout Interface Options *(Continued)*

Switch	Breakout Options
QFX5240-64OD	• 4 × 100GbE • 8 × 100GbE • 2 × 200GbE • 4 × 200GbE
QFX5240-64QD	• 8 × 50GbE • 2 × 100GbE • 4 × 100GbE • 8 × 100GbE • 2 × 200GbE • 4 × 200GbE • 2 × 400GbE • 2 × 50GbE
QFX5241-32OD	• 8 × 50GbE • 8 × 100GbE • 2 × 200GbE • 2 × 400GbE • 4 × 100GbE • 4 × 200GbE

Table 3: Breakout Interface Options *(Continued)*

Switch	Breakout Options
QFX5241-64OD	• 4 × 100GbE • 8 × 100GbE • 2 × 200GbE • 4 × 200GbE • 2 × 400GbE • 4 × 200GbE
QFX5241-64QD	• 8 × 50GbE • 2 × 100GbE • 4 × 100GbE • 8 × 100GbE • 2 × 200GbE • 4 × 200GbE • 2 × 400GbE • 2 × 50GbE

Table 3: Breakout Interface Options *(Continued)*

Switch	Breakout Options
QFX5250-64OE-L	• 1 x 1.6TbE • 2 x 800GbE • 4 x 400GbE • 8 x 200GbE • 1 x 800GbE • 2 x 400GbE • 4 x 200GbE • 8 x 100GbE

To validate the supported port speeds and breakout options for your QFX Series Switches, see Port Checker Tool.

Configure Interfaces

After the initial configuration of your device, you can configure physical interfaces and Layer2 and Layer3 interfaces, channelize interfaces, and aggregate Ethernet interfaces on your QFX Series Switch:

- To configure the interface description:

```
set interfaces interface-name description "description"
```

- To configure interface speed without channelization:

```
set interfaces interface-name ether-options speed speed
```

- (Optional) To configure Layer 2 interfaces at the enterprise level (access or trunk port, assigning interface to VLAN):

```
set interfaces interface-name1 unit 0 family ethernet-switching interface-mode access
set interfaces interface-name1 unit 0 family ethernet-switching vlan members vlan-name]
set interfaces interface-name2 unit 0 family ethernet-switching interface-mode trunk
```

```
set interfaces interface-name2 unit 0 family ethernet-switching vlan members [vlan-name1vlan-name2]
```

- (Optional) To configure an IPv4 address on a logical interface for Layer 3 routing:

```
set interfaces interface-name unit 0 family inet address ip-address/prefix-length
set interfaces interface-name unit 0 family inet6 address ipv6-address/prefix-length
```

- To configure channelization (breaking down high-speed ports into multiple lower-speed interfaces):
  - QFX5220:

    ```
    set chassis fpc fpc pic pic port port-number number-of-sub-ports n speed speed
    ```

  - QFX5230, QFX5240, QFX5241, and QFX5250:

    ```
    set interfaces interface-name number-of-sub-ports n speed speed
    ```

    Or,

    ```
    set interfaces interface-name speed speed number-of-sub-ports n
    ```

    In the commands shown above, *n* can have a value of 2, 4, or 8 and *speed* can have a value of 10g, 25g, 50g, 100g, 200g, or 400g.

- To configure link aggregation by bundling multiple physical Ethernet interfaces into a single logical aggregated Ethernet (ae-) interface and enabling Link Aggregation Control Protocol (LACP) for dynamic negotiation:

```
set interfaces physical-interface-1 ether-options 802.3ad ae-interface
set interfaces physical-interface-2 ether-options 802.3ad ae-interface
set interfaces ae-interface aggregated-ether-options lacp active
set interfaces ae-interface unit unit family inet/inet6 address ip-address/prefix-length
```

In this configuration, `lacp active` mode allows an Ethernet link to actively transmit LACP PDUs to its partner. Another LACP mode, `lacp passive`, sends LACP PDUs only when the Ethernet link receives them from the remote end. For more information about aggregated Ethernet LACP for switches, see Configure LACP Link Protection of Aggregated Ethernet Interfaces for Switches.

To specify the interval and speed at which the interfaces send LACP packets:

```
set interfaces ae-interface aggregated-ether-options lacp periodic interval
```

- To configure a VLAN:

```
set interfaces ae-interface unit unit-number family ethernet-switching interface-mode mode
set interfaces ae-interface unit unit-number family ethernet-switching vlan members vlan-name
```

For example:

```
set interfaces et-0/0/1 unit 0 family ethernet-switching interface-mode access
set interfaces et-0/0/1 unit 0 family ethernet-switching vlan members VLAN10

set interfaces et-0/0/48 unit 0 family ethernet-switching interface-mode trunk
set interfaces et-0/0/48 unit 0 family ethernet-switching vlan members [ VLAN10 VLAN20 ]
```

- To configure L3 routing for the associated VLANs (leaf devices in ERB design, spine devices in CRB design):

```
set vlans VLAN-NAME-1 vlan-id ID-1
set vlans VLAN-NAME-1 l3-interface irb.IRB-UNIT-1

set vlans VLAN-NAME-2 vlan-id ID-2
set vlans VLAN-NAME-2 l3-interface irb.IRB-UNIT-2

set interfaces irb unit IRB-UNIT-1 family inet address IP-address/PREFIX-1
set interfaces irb unit IRB-UNIT-2 family inet address IP-address/PREFIX-2
```

For example:

```
set vlans VLAN10 vlan-id 10
set vlans VLAN10 l3-interface irb.10

set vlans VLAN20 vlan-id 20
set vlans VLAN20 l3-interface irb.20
```

```
set interfaces irb unit 10 family inet address 192.0.2.1/24
set interfaces irb unit 20 family inet address 198.51.100.1/24
```

- To configure a VLAN with a specified name and assign it a unique VLAN identifier (ID number):

```
set vlans vlan-name vlan-id vlan-id-number
```

- To associate a Layer 3 integrated routing and bridging (IRB) interface with the specified VLAN, enabling routing between that VLAN and other networks:

```
set vlans vlan-name l3-interface irb.logical-interface-number
```

- For more information about configuring IRB interfaces on switches, see Configuring IRB Interfaces on Switches.

Verify Interfaces

To verify interface configuration on your device:

```
show chassis hardware
show interfaces terse-
show interfaces interface-name
show interfaces interface-name extensive
show ethernet-switching interface interface-name
```

To verify that LACP has been set up correctly and enabled as active on one end:

```
show lacp interfaces interface-name
```

To verify that the LACP packets are being exchanged between interfaces:

```
show lacp statistics interfaces aggregated-interface
```

To verify the active and standby links:

```
run show interfaces redundancy
```

To validate the optical module connected to an interface by reviewing its diagnostics such as transmit and receive power.

```
show interfaces diagnostics optics interface-name
```

For more information about configuring, monitoring, and troubleshooting Ethernet interfaces, see Interfaces User Guide for Switches.

Connectivity and Transceivers Types

SUMMARY

This section provides information about the connectivity types (transceivers) supported by QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

Before working on any switch, you must know the connectivity types a switch can offer and the associated use cases.

Table 4: QFX Series Switches: Connectivity Types and Use Cases

Platform Type	Transceiver Type	Speed	Reachability	Fiber Type	Use Case
QFX Series Switch	DAC	800 G	1 m – 2 m	Twinax Copper	Intra-rack, server to TOR
		400 G	1 m, 2 m, 2.5 m		
		100 G	1 m, 3 m, 5 m, 2 m (breakout)		
		40 G	50 cm, 1 m, 3 m, 5 m		

Table 4: QFX Series Switches: Connectivity Types and Use Cases *(Continued)*

Platform Type	Transceiver Type	Speed	Reachability	Fiber Type	Use Case
		10 G	1 m, 2 m, 3 m, 5 m, 7 m, 10 m		
	AOC	800 G, 400 G, 100 G, 40 G, 10 G	1 m, 3 m, 5 m, 7 m, 10 m, 15 m, 20 m, 30 m	Fiber	TOR to leaf switch, rack to rack
	VR	800 G, 400 G	< 50 m	Multimode	Intra-rack, server to TOR
	VR8	1.6 T			
	SR	100 G, 2 X 100 G, 40 G, 25 G, 10 G	< 100 m	Multimode	TOR to leaf switch, short intra-DC connections
	DR	800 G, 2 X 400 G, 400 G, 100 G	500 m	Single-mode	Spine-and-leaf in large pods, AI/ML clusters
	DR8	1.6 T	< 2 km		
		800 G			
	FR	1.6 T, 800 G, 2 X 400 G, 400 G, 4 X 100 G, 8 X 100 G, 100 G	< 2 km	Single-mode	Leaf to spine or spine to spine across buildings
	LR	2 x 400 G, 8 X 100 G, 400 G, 4 X 100, 2 X 100 G, 100 G, 40 G, 4 X 10 G, 10 G, 25 G	10 km	Single-mode	Interbuilding within campus or metro access

Table 4: QFX Series Switches: Connectivity Types and Use Cases *(Continued)*

Platform Type	Transceiver Type	Speed	Reachability	Fiber Type	Use Case
	ZR	400 G, 100 G	Up to 80 km (unamplified) and 140 km (amplified)	Single-mode	Distributed data center interconnect, Metro, regional, and core networks, Edge and DCI deployments, packet-optical convergence

NOTE: For more information about transceivers, see What is meant by DR, FR, LR, and VR in an optic?

For more information about transceivers and cables supported by the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches, see the respective sections in the Hardware Compatibility Tool (HCT), as listed here:

- QFX5220-32CD: Hardware Components Supported on QFX5220-32CD.
- QFX5220-128C: see Hardware Components Supported on QFX5220-128C.
- QFX5230-64CD: Hardware Components Supported on QFX5230-64CD.
- QFX5240-64OD: see Hardware Components Supported on QFX5240-64OD.
- QFX5240-64QD: see Hardware Components Supported on QFX5240-62QD.
- QFX5241-64OD: see Hardware Components Supported on QFX5241-64OD.
- QFX5241-64QD: see Hardware Components on QFX5241-64QD.
- QFX5250-64OE-L: see Hardware Components on QFX5250-64OE-L.

4

CHAPTER

Key Features and Configuration

IN THIS CHAPTER

BGP Auto-Discovered Neighbor (BGP Unnumbered Peering)

SUMMARY

This section describes BGP auto-discovered neighbor (also known as BGP unnumbered peering) feature and its configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

The QFX Series Switches in EVPN-VXLAN data center fabrics use BGP auto-discovered neighbors to simplify IPv6 underlay configuration. The switches automatically discover and establish peering sessions with directly connected neighbors using link-local IPv6 addresses. You can use this feature to configure BGP peering by interface rather than by specifying remote or local neighbor IP addresses. Using the IPv6 Neighbor Discovery Protocol to automatically discover link-local addresses of directly connected neighbors, you can simplify deployment and reduce misconfiguration risk.

QFX Series Switches in the EVPN-VXLAN data center fabric support both Internal Border Gateway Protocol (IBGP) for the overlay routing and External Border Gateway Protocol (EBGP) for overlay and underlay routing within spine-and-leaf architecture fabrics. For more information about BGP, see Understanding BGP.

ATTENTION: IPv6 Neighbor Discovery must be enabled and operational on the fabric-facing interfaces for BGP to discover peer link-local addresses. Auto-discovered must be configured on both ends of each link (example, on leaf and spine nodes).

Configure BGP Auto-Discovered Neighbor

Using this feature, you can configure BGP neighbors by interface or interface range, eliminating the need to manually define per-peer IP addresses for each BGP session. It expands your fabric without reconfiguring individual peer addresses, reducing operational complexity, and provisioning errors.

To enable BGP auto-discovered neighbor using IPv6 Neighbor Discovery on a device:

1. Configure the fabric-facing interfaces with both `inet` and `inet6` address families:

```
set interfaces interface-range to-spine member interface-name-1
set interfaces interface-range to-spine member interface-name-2
set interfaces interface-range to-spine unit 0 family inet address IPv4-address
set interfaces interface-range to-spine unit 0 family inet6 IPv6-address
```

2. Configure the local autonomous number (AS) number:

```
set routing-options autonomous-system AS-number
```

3. Define an AS list to restrict which remote AS numbers are accepted:

```
set policy-options as-list a-list members start-end
```

   It allows you to define an allowed AS number list so that only neighbors with matching AS numbers can establish BGP sessions, maintaining security without static neighbor definitions.

4. Enable IPv6 router advertisements (RAs) on the fabric-facing interfaces:

```
set protocols router-advertisement interface interface-name1 max-advertisement-interval
maximum-interval-number
set protocols router-advertisement interface interface-name1 min-advertisement-interval
minimum-interval-number
```

5. Configure the BGP group with dynamic neighbor autodiscovery:

```
set protocols bgp group autodisc type external
set protocols bgp group autodisc family inet unicast extended-nexthop
set protocols bgp group autodisc family inet6 unicast

set protocols bgp group autodisc dynamic-neighbor ndp peer-auto-discovery family inet6 ipv6-nd
set protocols bgp group autodisc dynamic-neighbor ndp peer-auto-discovery interface interface-
name1
set protocols bgp group autodisc peer-as-list a-list
```

NOTE: To dynamically discover IPv6 neighbors, QFX Series Switches use various options like BGP unnumbered peering, IPv6 neighbor discovery protocol (NDP), and so on. The use of dynamic discovery reduces the burden of manually configuring an IPv6 underlay in an EVPN-VXLAN data center fabric. This feature is called as *BGP autodiscovery* or *BGP autopeering*.

Verify BGP Auto-Discovered Neighbor Configuration

- To check whether the fabric interfaces are up and running:

```
show interfaces interface-name
show interfaces terse
```

- To verify that all fabric interfaces are sending and receiving IPv6 router advertisements:

```
show ipv6 router-advertisement
```

- To verify that the fabric devices have learned the MAC-to-link-local address bindings of all directly attached IPv6 neighbors using IPv6 neighbor discovery:

```
show ipv6 neighbors
```

- To verify that all fabric devices have established BGP peering sessions to the directly connected neighbors:

```
show bgp summary auto-discovered
show bgp neighbor auto-discovered
```

- To verify that all nodes are advertising their loopback address while learning the loopback address of other nodes:

```
show route protocol bgp
```

Class of Service

SUMMARY

This section briefly describes the class-of-service (CoS) feature and its configuration on the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

IN THIS SECTION

Overview

The class-of-service (CoS) feature enables Juniper devices to divide the network traffic into classes and set various levels of throughput and packet loss when congestion occurs. Using this feature, you can control packet loss as you can define rules according to your network requirements.

The QFX Series Switches QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 also support CoS where control plane and data plane processes and functions run in parallel, optimizing the utilization of the high-performance CPU. The CoS hardware specification details for your switch are as follows:

Table 5: CoS Hardware Specifications Supported by QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches

QFX Series Switches	CoS Hardware Specifications						
	Buffer Memory	DSCP Classifiers	Rewrite	Queues	Queuing and Scheduling	WRED	802.1p Classifiers
QFX5220	64 MB	64 profiles	128 profiles	8 unicast and 2 multicast per port	Fixed hierarchical scheduling (FHS) and two-level hierarchy	128 profiles per pipe	64 profiles
QFX5230-64CD	112 MB	64 profiles	128 profiles	8 unicast and 2 multicast per port	Four-level hierarchical scheduling scheme	128 profiles per pipe	64 profiles

Table 5: CoS Hardware Specifications Supported by QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches *(Continued)*

QFX Series Switches	CoS Hardware Specifications						
	Buffer Memory	DSCP Classifiers	Rewrite	Queues	Queuing and Scheduling	WRED	802.1p Classifiers
QFX5240	165.2 MB	64 profiles	128 profiles	8 unicast and 2 multicast per port	Four-level hierarchical scheduling scheme	32 pipes, each with 128 profiles	64 profiles
QFX5241	165.2 MB	64 profiles	128 profiles	8 unicast and 2 multicast per port	Four-level hierarchical scheduling scheme	32 pipes, each with 128 profiles	64 profiles
QFX5250	267 MB	64 profiles	128 profiles	8 unicast and 2 multicast per port	Four-level hierarchical scheduling scheme	32 pipes, each with 128 profiles	64 profiles

CoS Configuration for AI Data Center

Juniper CoS provides a flexible set of components that you can use to control the traffic on your network. Here's a list of some of the components:

- Defining classifiers that classify incoming traffic into forwarding classes to place traffic in groups for transmission.
- Configuring schedulers for each output queue to control the service level (priority, bandwidth characteristics) of each type of traffic.
- Configuring various CoS components individually or in combination to define CoS services.

If you do not configure CoS settings, Junos OS performs some CoS functions to ensure that traffic and protocol packets are forwarded with minimum delay when the network experiences congestion.

Configure Behavior Aggregate Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)

Behavior aggregate (BA) classifiers examine the Differentiated Services Code Point (DSCP or DSCP IPv6) value, the IEEE 802.1p CoS value, or the MPLS EXP value in the packet header to determine the CoS settings applied to the packet. You can use these classifiers to set the forwarding class and loss priority of a packet based on the incoming CoS value.

To configure a BA classifier:

```
set class-of-service classifiers dscp | dscp-ipv6 | ieee-802.1 classifier-name import default
set class-of-service classifiers dscp | dscp-ipv6 | ieee-802.1 classifier-name forwarding-class
forwarding-class-name loss-priority loss-priority-value code-points value
set class-of-service interfaces interface-name unit unit classifiers dscp | dscp-ipv6 |
ieee-802.1 classifier-name
set class-of-service interfaces interface-* unit * classifiers dscp | dscp-ipv6 | ieee-802.1
classifier-name
```

To verify the BA classifier configuration:

```
show configuration class-of-service interfaces interface-name
show configuration class-of-service classifiers
show class-of-service interface interface-name
show class-of-service interface interface-name detail
```

For information about BA classifier configuration, see configuring BA classifiers.

Configure Schedulers

A scheduler is an egress service policy attached to an interface's queue that helps in deciding the priority and bandwidth share of the network traffic. Each interface has multiple output queues (mapped from forwarding-class, 802.1p, or DSCP). When traffic congestion happens on the egress port, the scheduler determines which queue gets sent next and how much.

You can manage and control the traffic congestion on your switch by using the Data Center Quantized Congestion Notification (DCQCN) approach in the "RoCEv2" on page 72 environment. It provides mechanisms to adjust traffic rates in response to congestion events without relying on packet drops, striking a balance between reducing traffic rates and maintaining ongoing traffic flow. To implement flow and congestion control, DCQCN combines priority-based flow control (PFC) and explicit congestion notification (ECN). PFC mitigates data loss by pausing traffic transmission for specific traffic classes, based on IEEE 802.1p priorities or DSCP markings mapped to queues. Whereas, ECN offers a proactive congestion signaling mechanism, reducing transmission rates while allowing traffic to continue flowing during congestion periods.

To classify traffic on the basis of DSCP and implement traffic classification using the fabric-dscp classifier:

```
set class-of-service classifiers dscp classifier-name forwarding-class forwarding-class-name1
loss-priority low code-points dscp-values
set class-of-service classifiers dscp classifier-name forwarding-class forwarding-class-name2
loss-priority low code-points dscp-values
set class-of-service interfaces interface-name unit 0 classifiers dscp classifier-name
set class-of-service interfaces interface-name unit 0 scheduler-maps sm1
```

To verify DSCP classifier configuration:

```
show configuration class-of-service classifiers
show configuration class-of-service interfaces
show class-of-service classifiers dscp classifier-name
```

To configure shared packet buffer memory on your switch by partitioning it between ingress and egress:

```
set class-of-service shared-buffer ingress buffer-partition lossless percent ingress-lossless-
percent
set class-of-service shared-buffer ingress buffer-partition lossless dynamic-threshold ingress-
lossless-dynamic-threshold
set class-of-service shared-buffer ingress buffer-partition lossless-headroom percent ingress-
lossless-headroom-percent
set class-of-service shared-buffer ingress buffer-partition lossy percent ingress-lossy-percent
set class-of-service shared-buffer egress buffer-partition lossless percent egress-lossless-
percent
set class-of-service shared-buffer egress buffer-partition lossy percent egress-lossy-percent
```

To apply a congestion-notification profile to interfaces, enable PFC watchdog under that profile, identify which incoming packets belong to the PFC-protected class (DSCP-based), and configure a switch to map to 802.1p priority and choose the flow-control queue:

```
set class-of-service interfaces et-* congestion-notification-profile pfc
set class-of-service congestion-notification-profile pfc pfc-watchdog
set class-of-service congestion-notification-profile pfc input dscp code-point dscp-values
set class-of-service congestion-notification-profile pfc output ieee-802.1 code-point code-point-
bits flow-control-queue queue
```

To classify inbound traffic with DSCP and assign it to the forwarding class. This configuration maps it to queue, assigns pfc-priority, makes that queue a no-loss queue, and enables PFC for DSCP traffic:

```
set class-of-service classifiers dscp classifier-name forwarding-class forwarding-class-name
loss-priority loss-priority-value code-points value
set class-of-service forwarding-classes class forwarding-class-name queue-num queue-num-value
set class-of-service forwarding-classes class forwarding-class-name pfc-priority pfc-priority-
value
set class-of-service congestion-notification-profile profile-name pfc input dscp code-point value
```

To configure NO-LOSS traffic scheduling:

```
set class-of-service schedulers scheduler-name drop-profile-map loss-priority low|medium-low|
medium-high|high protocol any drop-profile drop-profile-name
set class-of-service drop-profiles drop-profile-name interpolate fill-level fill-level1
set class-of-service drop-profiles drop-profile-name interpolate fill-level fill-level2
set class-of-service drop-profiles drop-profile-name interpolate drop-probability drop-
probability-1
set class-of-service drop-profiles drop-profile-name interpolate drop-probability drop-
probability-2
```

To verify NO-LOSS traffic scheduling configuration:

```
show class-of-service interface interface-name
show configuration class-of-service schedulers
show configuration class-of-service drop-profiles
```

To assign strict-high priority to the queue and reserve 5% of the interface's bandwidth:

```
set class-of-service schedulers scheduler-name priority strict-high
set class-of-service schedulers scheduler-name transmit-rate percent 5
set class-of-service schedulers scheduler-name shaping-rate percent 5
```

To view the scheduler-map, congestion-notification status, profile name, and the classifier applied to the interface:

```
show class-of-service interface interface-name
```

To view the mapping between DSCP values and forwarding classes and to confirm correct assignments:

```
show class-of-service classifier name classifier-name
```

To view the forwarding-classes to queue mapping:

```
show class-of-service forwarding-class
```

This command is used to verify the queue mapping (CNP is greater than equal to queue 3, and NO-LOSS is greater than equal to queue 4), and the no-loss status and PFC priority of the NO-LOSS queue.

To view the scheduler map and the schedulers including their priority, assigned rate, and whether ECN is enabled:

```
show class-of-service scheduler-map scheduler-map-name
```

To view peak queue occupancy for each queue on interface:

```
show class-of-service shared-buffer
```

EBGP for the Overlay

SUMMARY

This section briefly describes EBGP for the overlay feature and its configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

External Border Gateway Protocol (EBGP) for the overlay means using EVPN signaling and peering in the VXLAN overlay networks, specifically in IPv6 fabric designs on the QFX5220, QFX5230, QFX5240,

and QFX5241 Switches. EBGP peers between IPv6 loopback addresses of these QFX Series switches to handle EVPN signaling for VXLAN tunnels. This feature simplifies scaling in EVPN-VXLAN fabrics, enhances reliability with BFD, and supports efficient load balancing in multistage Clos topologies.

Configure EBGP for the Overlay

To enable EBGP and set router ID:

```
set routing-options router-id loopback-ip
```

To configure EBGP group and neighbors for the spine and leaf overlay:

```
set protocols bgp group overlay type external
set protocols bgp group overlay local-address loopback-ip
set protocols bgp group overlay family inet unicast
set protocols bgp group overlay family evpn
set protocols bgp group overlay peer-as remote-as
set protocols bgp group overlay neighbor peer-loopback-ip
```

To add BFD for multihop overlay sessions:

```
set protocols bgp group BGP-group-name multihop
set protocols bgp group BGP-group-name multihop ttl EBGP-multihop-ttl
set protocols bgp group BGP-group-name neighbor peer-loopback-ip bfd-liveness-detection minimum-
interval bfd-min-interval-ms
set protocols bgp group BGP-group-name neighbor peer-loopback-ip bfd-liveness-detection
multiplier bfd-multiplier
```

Verify EBGP for the Overlay Configuration

To verify that EBGP is functional on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches:

```
show bgp summary
```

To view detailed overlay session details such as status, configuration, and statistics for BGP neighbors:

```
show bgp neighbor
```

To view overlay routes in case EVPN is configured:

```
show evpn database
```

For more information about how to configure EBGP for IPv6 overlay peering, see Configure EBGP for IPv6 Overlay Peering.

ESI-LAG

SUMMARY

This section briefly describes the Ethernet Segment Identifier–Link Aggregation Group (ESI-LAG) feature and its configuration on the QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

NOTE: The QFX5220 does not support ESI-LAG.

The provider edge (PE) devices in an EVPN fabric can handle traffic to and from the attached multihomed end devices by grouping the multihoming links into an EVPN Ethernet segment with an identifier. This identifier is known as Ethernet segment identifier (ESI). The multihoming links participating in the Ethernet segment are configured into a link aggregation group (LAG). Thus, the set of multihomed links is called an ESI LAG or EZ-LAG. ESI-LAG provides a simplified configuration statement hierarchy and a built-in commit script that you can use to set up EVPN dual-homing. You can then easily migrate a multichassis link aggregation group (MC-LAG) topology to a standards-based EVPN-VXLAN multihoming model. For more information about ESI-LAG, see Benefits of using Easy EVPN LAG Configuration.

NOTE: ESI-LAG is an optional feature meant for only multihoming. The feature is not applicable to single-homing users.

Configure ESI-LAG on QFX5230, QFX5240, and QFX5241 Switches

To configure ESI-LAG on your QFX Series Switch:

- Run the following sample CLI commands at the `[edit services evpn]` hierarchy level by providing the parameters to set up the prescribed EVPN fabric.

```
set chassis aggregated-devices ethernet device-count N
set interfaces aeID aggregated-ether-options lacp active
set interfaces aeID aggregated-ether-options lacp periodic fast
set interfaces interface-name ether-options 802.3ad aeID
set interfaces aeID esi auto-derive type-1-lacp system-id system-id
set interfaces aeID all-active
set interfaces aeID unit 0 family ethernet-switching interface-mode interface-mode
set interfaces aeID unit 0 family ethernet-switching vlan members [VLAN-List]
```

The following sample configuration requires minimal parameters to configure an EVPN fabric with a single server dual-homed to two PE devices. When you commit the configuration, the built-in commit script generates the full underlying EVPN and ESI-LAG configuration. For more information about ESI-LAG, see Easy EVPN LAG (EZ-LAG) Configuration.

NOTE: This configuration is for a single PE device. For a dual-homed server, you must apply same configuration on the second PE device, using its local interface connected to the same server.

```
set services evpn device-attribute peer-id peer-id1 system-id system-id
set services evpn device-attribute peer-id peer-id1 peer-to-peer peer-subnet inet Peer-Subnet
set services evpn device-attribute peer-id peer-id1 peer-to-peer interface-name interface-name
set services evpn device-attribute loopback peer1-subnet inet IPv4-address
set services evpn evpn-vxlan irb irb-name vlan-id vlan-id subnet-address inet IPv4-Address
set services evpn evpn-vxlan server server-1 esi-lag-id ESI-Id
set services evpn evpn-vxlan server server-1 vlan-id-list [vlan-list]
set services evpn evpn-vxlan server server-1 interface-name interface-name
```

A built-in commit script processes the simplified configuration elements at commit time (before the standard Junos OS Evolved configuration validity checks) and generates a corresponding EVPN fabric configuration.

NOTE: The commit script for this feature, `services_evpn_commit_script.py`, is enabled by default on supported platforms.

```
set system scripts commit file services_evpn_commit_script.py allow-transient
```

Verify ESI-LAG Configuration

To verify your ESI-LAG configuration, run the following commands:

```
show chassis aggregated-devices

show interfaces ae-ID

show interfaces interface-name
```

Flexible Match Firewall Filters

SUMMARY

This topic briefly describes the flexible match firewall filter feature and its configuration on the QFX5230-64CD and QFX5240 Switches.

IN THIS SECTION

Overview

NOTE: This feature is applicable to the QFX5230-64CD and QFX5240 switches only.

You can use firewall filters on QFX Series Switches to control the network traffic on the basis of configured filtering criteria. These criteria include port number, source MAC address, destination MAC address, VLAN ID, EtherType, enabling or disabling any protocol, source IP address (in case of L3-aware mode), and so on. For more information about the firewall filters, see Overview of Firewall Filters for QFX Series Switches.

In addition to default firewall filters, the **QFX5230-64CD** and **QFX5240** switches can filter data packets on the basis of specified packet fields. For example, users can define filters such as from which layer to start filtering, the number of bytes to offset, or identifying a field with a value of 1000. When a data packet matches the defined filters, users can choose to accept or reject it on the basis of its relevance or potential risk to the network.

Configure Ingress Firewall Filter

To configure an ingress firewall filter for Layer 2 – Ethernet-Switching family:

```
set firewall family ethernet-switching filter filter-name term term-name from match-condition
set firewall family ethernet-switching filter filter-name term term-name then action
set interfaces interface-name unit unit-number family ethernet-switching filter input filter-name
```

Configure Egress Firewall Filter

To configure an egress firewall filter on Layer 3 – INET family:

```
set firewall family inet filter filter-name term term-name from match-condition
set firewall family inet filter filter-name term term-name then action
set interfaces interface-name unit unit-number family inet filter output filter-name
```

Verify Firewall Filter Configuration

To verify the firewall filter configuration:

```
show configuration firewall family ethernet-switching filter filter-name
show configuration interfaces interface-name | display set | match filter
show firewall filter filter-name
```

References

- For more information about firewall filters, see Firewall Filters Overview.
- For more information about how to configure firewall filters, see Configuring Firewall Filters.
- For information about the guidelines to configure firewall filters, see Guidelines for Configuring Firewall Filters.

IBGP Using IPv4 Overlay

SUMMARY

This section briefly describes the IBGP using IPv4 overlay feature and its configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

Internal Border Gateway Protocol (IBGP) is a BGP peering type used for routing IPv4 underlay and overlay networks in data center fabrics. IBGP operates within the same autonomous system (AS) as peer devices and requires full-mesh peering or route reflectors to avoid loops. IBGP uses internal peering and shares the same AS number across peers. IBGP sessions often use loopback interfaces (for example, lo0) as the local address for stability in IPv4 fabrics.

Configure IBGP Using IPv4 Overlay

To configure IBGP for the overlay peering in an IPv4 fabric:

1. Configure an AS number for overlay IBGP. All leaf and spine devices participating in the overlay use the same AS number.

   On spine and leaf devices:

   ```
   set routing-options autonomous-system AS-Number
   ```

2. Configure IBGP using EVPN signaling on each spine device to peer with every leaf device. Also, form the route reflector cluster and configure equal-cost multipath (ECMP) for BGP.

   TIP: By default, BGP selects only one best path when there are multiple, equal-cost BGP paths to a destination. When you enable BGP multipath by including the `multipath` statement at the `[edit protocols bgp group group-name]` hierarchy level, the device installs all of the equal-cost BGP paths into the forwarding table. This feature helps load-balance the traffic across multiple paths.

   Spine 1:

```
set protocols bgp group overlay type internal
set protocols bgp group overlay local-address spine-loopback-ip
set protocols bgp group overlay family evpn
set protocols bgp group overlay cluster route-reflector-cluster-id
set protocols bgp group overlay multipath
set protocols bgp group overlay neighbor leaf1-loopback-ip-address
...
set protocols bgp group overlay neighbor last-leaf-loopback-ip-address
```

3. Configure IBGP on the spine devices to peer with all the other spine devices acting as route reflectors. This step completes the full-mesh peering topology required to form a route reflector cluster.

   Spine 1:

```
set protocols bgp group overlay-full-mesh-spine type internal
set protocols bgp group overlay-full-mesh-spine local-address this-spine-loopback-ip-address
set protocols bgp group overlay-full-mesh-spine  family evpn
set protocols bgp group overlay-full-mesh-spine  neighbor other-spine1-loopback-ip-address
set protocols bgp group overlay-full-mesh-spine  neighbor other-spine2-loopback-ip-address
set protocols bgp group overlay-full-mesh-spine  neighbor other-spine3-loopback-ip-address
```

4. Configure BFD on all BGP groups on the spine devices to enable rapid detection of failures and reconvergence.

   Spine 1:

```
set protocols bgp group overlay bfd-liveness-detection minimum-interval interval-value
set protocols bgp group overlay bfd-liveness-detection multiplier multiplier-value
```

```
set protocols bgp group overlay bfd-liveness-detection session-mode automatic

set protocols bgp group  overlay-full-mesh-spine bfd-liveness-detection minimum-interval
interval-value
set protocols bgp group  overlay-full-mesh-spine bfd-liveness-detection multiplier multiplier-
value
set protocols bgp group  overlay-full-mesh-spine bfd-liveness-detection session-mode automatic
```

5. Configure IBGP with EVPN signaling from each leaf device (route reflector client) to each spine device (route reflector cluster).

   Leaf 1:

```
set protocols bgp group overlay type internal
set protocols bgp group overlay local-address leaf-loopback-ip-address
set protocols bgp group overlay family evpn
set protocols bgp group overlay neighbor leaf1-loopback-ip-address
set protocols bgp group overlay neighbor leaf2-loopback-ip-address
set protocols bgp group overlay neighbor leaf3-loopback-ip-address
set protocols bgp group overlay neighbor leaf4-loopback-ip-address
```

6. Configure BFD on the leaf devices to enable rapid detection of failures and reconvergence.

```
set protocols bgp group OVERLAY bfd-liveness-detection minimum-interval interval-value
set protocols bgp group OVERLAY bfd-liveness-detection multiplier multiplier-value
set protocols bgp group OVERLAY bfd-liveness-detection session-mode automatic
```

Verify IBGP Configuration Using IPv4 Overlay

To verify that IBGP is functional on the spine devices:

```
show bgp summary
```

To verify that BFD is operational on the spine devices:

```
show bfd session
```

To verify that IBGP is operational on the leaf devices:

```
show bgp summary
```

To verify that BFD is operational on the leaf devices:

```
show bfd session
```

IRB Anycast

SUMMARY

This section briefly describes integrated routing and bridging (IRB) feature and its anycast gateway configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

The integrated routing and bridging (IRB) feature enables Layer 2 bridging and Layer 3 IP routing on the same interface, allowing efficient routing between VLANs or bridge domains without a separate router. When specifying an IP or a MAC address for an IRB interface on the multihomed devices, you can use an anycast address. The anycast address enables you to configure the same addresses for the IRB interface on each of the multihomed devices, thereby establishing the devices as anycast gateways.

An IRB anycast gateway uses IRB interfaces on multihomed devices to share the same IP and MAC address as a default gateway, enabling redundancy and load balancing in an EVPN-MPLS, EVPN-VXLAN, or L2VPN setup.

The QFX5220, QFX5230, QFX5240, and QFX5241 switches support IRB on both default routing instances and virtual routing and forwarding (VRF) instances. To segment the traffic on a LAN into separate broadcast domains, separate virtual LANs (VLANs) are created. VLANs limit the amount of traffic flowing across the entire LAN, reducing the possible number of collisions and packet retransmissions within the LAN.

For more information about IRB, see Understanding Integrated Routing and Bridging.

Configure IRB Anycast Gateway

Your IP address subnet scheme helps to determine whether you can use the IRB interface commands to set up your anycast gateway or not. Before you begin the configuration, ensure that:

1. The router interfaces are configured.
2. An underlay protocol such as OSPF, ISIS, or BGP is configured.
3. BGP is configured.
4. MPLS is configured.

NOTE: For information about IRB anycast gateway configuration guidelines, see Anycast Gateway Configuration Guidelines and for limitations, see Anycast Gateway Configuration Limitations.

Run the following commands in the CLI at the `[edit]` hierarchy level.

To configure an IRB interface with an IPv4 or an IPv6 address:

```
set interfaces irb unit logical-unit-number family inet address ipv4-address/prefix-length
set interfaces irb unit logical-unit-number family inet6 address ipv6-address/prefix-length
```

To configure an IRB interface with a media access control (MAC) address:

```
set interface irb mac mac-address
```

NOTE: In the command above, you can also use the MAC address (chassis MAC) that Juniper Networks devices automatically generate.

To configure a virtual gateway address (VGA) with IPv4 and IPv6 address:

```
set interfaces irb unit logical-unit-number family inet address primary-ipv4-address/prefix
virtual-gateway-address gateway-ipv4-address
set interfaces irb unit logical-unit-number family inet6 address primary-ipv6-address/prefix
virtual-gateway-address gateway-ipv6-address
```

To configure a virtual gateway address (VGA) with MAC address:

```
set interfaces irb unit logical-unit-number virtual-gateway-mac mac-address
```

NOTE: In these commands, you can also use the MAC address (chassis MAC) that Juniper Networks devices automatically generate.

Verify IRB Anycast Gateway

Run the following sample commands to verify that the IRB anycast gateway feature is configured.

- To verify that the interface is up and IP is configured properly:

```
show interfaces irb terse
```

- To verify that IRB anycast IP address or MAC address is configured properly:

```
show ethernet-switching virtual-gateway
```

- To verify that advertised MAC address and IP addresses route is present:

```
show evpn database
show evpn route type mac-ip
```

- To verify leaf routing table is reachable:

```
show route table vrf-name.inet.0
```

- To verify end-to-end connectivity:

```
ping IRB-IP
```

In-Service Software Upgrade

SUMMARY

This topic briefly describes the in-service software upgrade (ISSU) feature and its configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches, except the QFX5220-128C and QFX5250-64OE-L switch.

IN THIS SECTION

Overview

The QFX Series Switches (except QFX5210 and QFX5220-128C) support the in-service software upgrade (ISSU) feature. You can use ISSU to upgrade the network devices to a new software version without disrupting the network service. For more information about ISSU on QFX Series Switches, see In-service Software Upgrade.

The upgrade time for an ISSU on the QFX Series Switches depends on the platform and network conditions. The procedure minimizes traffic disruption and downtime. The ISSU process restarts applications instead of performing full system reboot, reducing upgrade time.

Prerequisites to Perform ISSU

You must make sure that the following prerequisites are met before upgrading Junos OS Evolved on your switch:

- You have sufficient disk space for the upgrade and a backup of the system is available.
- Download the software package from the Juniper Networks Support website at https://www.juniper.net/support/ and place the package on your local server.
- If the BGP protocol is configured on the main routing instance or a specific routing instance, then configure BGP graceful restart and set the `restart-time` value to greater than or equal to 300 seconds.

```
set routing-instances vrf-name protocols bgp graceful-restart
set routing-instances vrf-name protocols bgp graceful-restart restart-time 300
```

- If OSPF runs on the main or specific routing instance, set the graceful `restart-time` value. The default is 180s, but you can configure it from 300s to 3600s.

```
set routing-instances vrf-name protocols ospf graceful-restart
set routing-instances vrf-name protocols ospf graceful-restart restart-duration 300 300
```

- If the IS-IS protocol is configured on the main routing instance or a specific routing instance, then configure IS-IS graceful restart and set the `restart-time` value to greater than or equal to 300 seconds. IS-IS `restart-duration` range is 30–300 seconds (default 30 seconds), with a maximum of 300 seconds.

```
set routing-instances instance-name protocols isis graceful-restart
set routing-instances instance-name protocols isis graceful-restart restart-duration 300
```

- To configure BGP graceful restart and the `restart-time` value on the main routing instance:

```
set protocols bgp graceful-restart
set protocols bgp graceful-restart restart-time seconds
```

- To ensure that graceful restart works reliably, set the IS-IS `hold-time` to 41 seconds or greater (default is 27 seconds).

```
set protocols isis interface interface-name hold-time 41
```

- If the PIM protocol is configured on the main routing instance or a specific routing instance, then configure PIM graceful restart and set the `restart-time` value to greater than or equal to 300 seconds.

```
set routing-instances instance-name protocols pim graceful-restart
set routing-instances instance-name protocols pim graceful-restart restart-time 300
```

NOTE: QFX5230 does not support graceful restart for the PIM protocol.

- To verify any protocol (IS-IS, PIM, BGP, OSPF) configuration:

```
show configuration protocols protocol-name | display set | match graceful-restart
 show protocol-name neighbor neighbor-address
 show protocol-name summary
```

- If a spanning tree protocol (STP) is configured, then configure the STP-enabled ports as edge ports and enable bridge protocol data unit (BPDU) protection.

```
set protocols (mstp | rstp | vstp) bpdu-block-on-edge
set protocols (mstp | rstp | vstp) interface (interface-name | all) edge
```

- Configure the value of the Address Resolution Protocol (ARP) aging timer to 240 minutes.

```
set routing-options arp aging-timer 240
```

  If you want to configure ARP under a specific routing-instance, use the following command:

```
set routing-instances VRF routing-options arp aging-timer 240
```

- Validate the existing configuration against the new software image to check whether it supports ISSU.

```
request system software validate-restart package-name
```

Upgrade a QFX Series Switch OS Using ISSU

To upgrade the QFX Series Switch operating system (OS) using ISSU:

1. Run the following command on the switch whose OS you want to upgrade:

```
request system software add package-path restart
```

2. Log in at the login prompt and run the following command to verify the release of the installed software:

```
show system software list
```

3. Verify that the system is running properly and correctly handling traffic following prerequisites listed in "Prerequisites to Perform ISSU" on page 66.

4. If you want to make any changes to the configuration after the upgrade, remember to back up the software and configuration:

```
request system snapshot
```

Effects of ISSU

The packet or data loss that happens during the upgrade process varies from switch to switch.

- The QFX5220-32CD switch has a minimal packet loss of up to 50 ms during upgrade, whereas the QFX5230, QFX5240, and QFX5241 switches have a packet loss of up to 200 ms.
- There is more data loss in the QFX5230, QFX5240, and QFX5241 switches as compared to the QFX5220-32CD switch. This is because the new switches offer more scalability, flexibility, improved architecture, and support for new features such as EVPN-VXLAN.

EVPN-VXLAN MAC-VRF Routing-Instance Type

SUMMARY

This section describes the EVPN-VXLAN MAC-VRF routing-instance type for AI-ML data centers and its configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

Juniper Networks QFX5220, QFX5230, QFX5240, and QFX5241 Switches are designed for high-performance data center fabrics, particularly supporting overlay features such as EVPN-VXLAN, MVPN protocols, software-defined networking (SDN), and VLAN. In this topic, you will learn to configure EVPN-VXLAN for AI-ML data centers.

EVPNs use Layer 2 virtual bridges to connect dispersed customer sites. VXLANs extend Layer 2 over Layer 3 networks, offering scalable segmentation similar to VLANs. EVPN with VXLAN supports large-scale Layer 2 connectivity for cloud providers, replacing STP and enabling robust Layer 3 routing. For more information about EVPN-VXLAN, see Data Center EVPN-VXLAN Fabric Architecture Guide.

Configure EVPN-VXLAN for AI-ML Data Centers

To configure EVPN-VXLAN for AI-ML data centers, use MAC-VRF routing instances. MAC-VRF routing instances can support multiple customer-specific EVPN instances (EVIs) using different EVPN service types such as `vlan-based`, `vlan-aware`, and `vlan bundle`. For more information about the MAC-VRF routing instance type, see MAC-VRF Instances Enable Customer-Specific VRF Tables .

NOTE: The choice between `vlan-aware` and `vlan-based` depends on whether you want the instance to support multiple VLANs (`vlan-aware`) or only a single VLAN (`vlan-based`).

To configure VLAN-aware MAC-VRF instances on a leaf node:

NOTE: Replace the values in the sample code with those specific to your deployment. For additional MAC-VRF instances, repeat steps 1–7 with different VLANs, route distinguishers (RDs), VRF targets, and interfaces. For more information about MAC-VRF routing instances, see MAC-VRF Routing Instance Type Overview.

```
set routing-instances MAC-VRF-NAME instance-type mac-vrf
set routing-instances MAC-VRF-NAME service-type vlan-aware

set routing-instances MAC-VRF-NAME route-distinguisher AS:NUM
set routing-instances MAC-VRF-NAME vrf-target target:AS:NUM

set routing-instances MAC-VRF-NAME vtep-source-interface LOOPBACK-INTERFACE

set routing-instances MAC-VRF-NAME protocols evpn encapsulation vxlan
set routing-instances MAC-VRF-NAME protocols evpn default-gateway no-gateway-community
set routing-instances MAC-VRF-NAME protocols evpn extended-vni-list all
```

```
set routing-instances MAC-VRF-NAME interface ACCESS-INTERFACE.UNIT

set routing-instances MAC-VRF-NAME vlans VLAN-NAME-1 vlan-id VLAN-ID-1
set routing-instances MAC-VRF-NAME vlans VLAN-NAME-1 vxlan vni VNI-1
set routing-instances MAC-VRF-NAME vlans VLAN-NAME-1 l3-interface IRB-INTERFACE-1

set routing-instances MAC-VRF-NAME vlans VLAN-NAME-2 vlan-id VLAN-ID-2
set routing-instances MAC-VRF-NAME vlans VLAN-NAME-2 vxlan vni VNI-2
set routing-instances MAC-VRF-NAME vlans VLAN-NAME-2 l3-interface IRB-INTERFACE-2
```

To configure VLAN-based MAC-VRF instances on a leaf node:

```
set routing-instances MAC-VRF-NAME instance-type mac-vrf
set routing-instances MAC-VRF-NAME service-type vlan-based

set routing-instances MAC-VRF-NAME route-distinguisher AS:NUM
set routing-instances MAC-VRF-NAME vrf-target target:AS:NUM

set routing-instances MAC-VRF-NAME vtep-source-interface LOOPBACK-INTERFACE
set routing-instances MAC-VRF-NAME protocols evpn encapsulation vxlan
set routing-instances MAC-VRF-NAME interface ACCESS-INTERFACE.UNIT

set routing-instances MAC-VRF-NAME vlan-id VLAN-ID
set routing-instances MAC-VRF-NAME vxlan vni VNI
set routing-instances MAC-VRF-NAME l3-interface IRB-INTERFACE
```

To configure VLAN-bundle MAC-VRF instances on a leaf node:

```
set routing-instances MAC-VRF-NAME instance-type mac-vrf
set routing-instances MAC-VRF-NAME service-type vlan-bundle
set routing-instances MAC-VRF-NAME route-distinguisher AS:NUM
set routing-instances MAC-VRF-NAME vrf-target target:ASN:TARGET-ID
set routing-instances MAC-VRF-NAME vtep-source-interface LOOPBACK-INTERFACE
set routing-instances MAC-VRF-NAME protocols evpn encapsulation vxlan

set routing-instances MAC-VRF-NAME vlans VLAN-NAME vlan-id VLAN-ID
set routing-instances MAC-VRF-NAME vlans VLAN-NAME vxlan vni VNI

set interfaces interface-name unit unit family ethernet-switching interface-mode access|trunk
set interfaces interface-name unit unit family ethernet-switching vlan members VLAN-NAME
```

Verify EVPN-VXLAN MAC-VRF Routing Instance Configuration

Before verifying the EVPN-VXLAN MAC-VRF routing instance configuration, ensure that the other devices in your network are configured properly and then run the following sample commands to verify the output:

```
show evpn instance
show routing-instances MAC-VRF-name
show route table MAC-VRF-name.evpn.0 active-path
show mac-vrf forwarding vlans
```

For more information about this feature, see EVPN-VXLAN for AI-ML Data Centers.

RDMA over Converged Ethernet version 2

SUMMARY

This topic briefly describes the RDMA over Converged Ethernet version 2 (ROCEv2) feature and its configuration on the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

IN THIS SECTION

Overview

Juniper Networks supports ROCEv2 primarily on its QFX Series Switches including QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 for building high performance lossless Ethernet fabrics for AI/ML workloads and storage. This feature is ideal for boosting data center efficiency, reducing overall complexity, and increasing data delivery performance. ROCEv2 traffic flow can coexist with other types of traffic flows in the network. ROCEv2-based Ethernet fabric uses UDP protocol whereas traditional Ethernet fabric uses TCP. Explicit congestion notification (ECN) and priority-based flow control (PFC) are some of the benefits that ROCEv2 offers over other Ethernet traffic flows. Thus, RDMA traffic is given priority over other traffic flows to read and write data to the server.

In the following section, you will learn to configure and verify ROCEv2 configuration on your Juniper QFX Series Switches. The configuration includes configuring ECN and PFC. For more information about data center quantized congestion notification (DCQCN), see Understanding DCQCN

Configure ROCEv2

To enable DCQCN, configure both ECN and PFC for a traffic flow.

1. Configure classifiers for ROCEv2 traffic and for Congestion Notification Packets (CNP).

```
set class-of-service classifiers dscp mydscp forwarding-class CNP loss-priority FC-DCSP code-points DSCP-CNP
set class-of-service classifiers dscp mydscp forwarding-class NO-LOSS loss-priority FC-NO-LOSS code-points DSCP-NO-LOSS
set interfaces interface-name unit unit classifiers dscp mydscp
```

2. Configure ECN on the egress port for a lossless flow.

```
set class-of-service drop-profiles dp1 interpolate fill-level Min-Fill-Percentage drop-probability Min-Drop-Prob
set class-of-service drop-profiles dp1 interpolate fill-level Max-Fill-Percentage drop-probability Max-Drop-Prob
set class-of-service schedulers s1 drop-profile-map loss-priority loss|high protocol ip|non-ip drop-profile dp1
set class-of-service schedulers s1 explicit-congestion-notification
set class-of-service schedulers s2-cnp transmit-rate percent cnp-rate-percent
set class-of-service schedulers s2-cnp priority priority
set class-of-service scheduler-maps sm1 forwarding-class NO-LOSS scheduler s1
set class-of-service scheduler-maps sm1 forwarding-class CNP scheduler s2-cnp
set class-of-service interfaces interface-name scheduler-map sm1
```

3. Configure PFC on the ingress port for the same lossless flow.

```
set class-of-service classifiers dscp LOSSLESS-DSCP code-point DSCP-Code-Point forwarding-class NO-LOSS loss-priority loss-priority
set class-of-service interfaces interface-name unit unit classifiers dscp LOSSLESS-DSCP
set class-of-service rewrite-rules ieee-802.1 LOSSLESS-PCP forwarding-class NO-LOSS loss-priority loss-priority code-point PCP-Code-Point
set class-of-service interfaces interface-name unit unit rewrite-rules ieee-802.1 PCP-Name
set protocols dcbx interface interface-name
set protocols dcbx interface interface-name priority-flow-control priority pfc-priority
```

4. Configure the shared buffers.

```
set class-of-service shared-buffer ingress buffer-partition lossless percent ingress-lossless-percent
set class-of-service shared-buffer ingress buffer-partition lossless dynamic-threshold ingress-lossless-dynamic-threshold
set class-of-service shared-buffer ingress buffer-partition lossy percent ingress-lossy-percent
set class-of-service shared-buffer ingress buffer-partition lossless-headroom percent ingress-lossless-headroom-percent
set class-of-service shared-buffer egress buffer-partition lossless percent engress-lossless-percent
set class-of-service shared-buffer egress buffer-partition lossy percent engress-lossy-percent
```

NOTE: You must follow these rules to commit the configuration on platforms running Junos OS Evolved:

- You must configure all three of the ingress partitions.
- The sum of the ingress shared buffer configuration for all partitions must be 100 percent.
- For lossy and lossless buffer partitions both the ingress and egress buffer-partition percentages should be equal.

Setting `dynamic-threshold` for the lossless ingress buffer partition is optional. ECN uses this option for the threshold calculation on lossless queues. If you don't configure this option, `dynamic-threshold` uses its default value of 7.

5. Configure forwarding classes and assign queues.

```
set class-of-service forwarding-classes class CNP queue-num queue-num
set class-of-service forwarding-classes class NO-LOSS queue-num queue-num
set class-of-service forwarding-classes class NO-LOSS no-loss
set class-of-service forwarding-classes class NO-LOSS pfc-priority pfc-priority
```

6. Verify your configuration.

```
show class-of-service drop-congestion-notification
show class-of-service drop-congestion-notification interface
```

```
show class-of-service forwarding-classes
show class-of-service interfaces
```

7. Commit your configuration.

```
commit
```

Route Type 5 IP Routing Instance

SUMMARY

This section provides an overview of the Route Type 5 routing instance feature and its configuration on the QFX5220, QFX5230, QFX5240, QFX5241 Switches.

IN THIS SECTION

Overview

You can use EVPN-VXLAN to extend the Layer 2 connectivity between the different data centers using route types. Route types establish a session between the provider edge and the customer edge. There are many route types such as Type 1 route, Type 2 route, Type 3 route, Type 4 route, Type 5 route, Type 6 route, Type 7 route, and Type 8 route. For more information about different route types, see Defining EVPN-VXLAN Route Types.

A Type 5 route (also referred to as the IP prefix route) is used to enable communication between data centers when the Layer 2 connection does not extend across data centers and the IP subnet in a Layer 2 domain is confined within a single data center. In this scenario, the Type 5 route enables connectivity across data centers by advertising the IP prefixes assigned to the VXLANs confined within a single data center. The data packets are sent as Layer 2 Ethernet frames encapsulated in the VXLAN header. Additionally, the gateway device for the data center performs Layer 3 routing and provides an IRB functionality.

For more information about Route Type 5 implementation in an EVPN-VXLAN environment, see Implementing Pure Type 5 Routes in an EVPN-VXLAN Environment.

Configure Route Type 5 IP Routing Instance

To configure a Route 5 Type (RT5) IP routing instance on a your switch, you need to use a supported Layer 2 (L2) instance type (see instance-type) in which you can enable the EVPN protocol with other parameters such as an encapsulation type, a route distinguisher, and a route target. The supported instance-type options for the EVPN protocol includes the `evpn` instance type (default), `virtual-switch` instance type, and `mac-vrf` instance type.

NOTE: Support for different EVPN instance types is platform-specific, so not all platforms support all of these instance-type values. This procedure uses the instance-type `evpn`.

- Configure a VRF routing instance:

```
set routing-instances VRF-NAME instance-type vrf
set routing-instances VRF-NAME route-distinguisher RD-ASN:RD-NUM
set routing-instances VRF-NAME vrf-target target:RT-ASN:RT-NUM
```

- Configure the EVPN protocol with Type 5 (IP prefix routes).

```
set routing-instances VRF-NAME protocols evpn ip-prefix-routes advertise direct-nexthop
set routing-instances VRF-NAME protocols evpn ip-prefix-routes encapsulation vxlan
set routing-instances VRF-NAME protocols evpn ip-prefix-routes vni VNI-ID
set routing-instances VRF-NAME protocols evpn ip-prefix-routes export EXPORT-POLICY-NAME
```

- Configure routing options:

```
set routing-instances VRF-NAME routing-options static route PREFIX/LEN discard
set routing-instances VRF-NAME routing-options multipath
```

- Configure interfaces, the route distinguisher, and the VRF target:

```
set routing-instances VRF-NAME interface INTERFACE-1
set routing-instances VRF-NAME interface INTERFACE-2
set routing-instances VRF-NAME route-distinguisher RD-IP:RD-VALUE
```

```
set routing-instances VRF-NAME vrf-target target:TARGET-ASN-OR-IP:TARGET-ID
set routing-instances VRF-NAME vrf-table-label
```

For more information about RT5 configuration, see Example: Selectively Enable DLB with a Firewall Filter Match Condition.

Verify RT5 IP Routing Instance Configuration

Run the following sample commands to verify RT5 routing instance configuration:

```
show bgp summary
show route table VRF-NAME.inet.0
show route table bgp.evpn.0
show route receive-protocol bgp neighbor
```

References

- For more information about how to configure seamless type-5 to type-5 stitching for EVPN-VXLAN, see Configuring Seamless Type-5 to Type-5 Stitching for EVPN-VXLAN.
- For more information about EVPN type-5 route with VXLAN encapsulation, see EVPN Type 5 Route with VXLAN Encapsulation for EVPN-VXLAN.

Secure Boot

SUMMARY

This topic describes the secure boot feature and its configuration on the QFX5241 and QFX5250 switches.

The QFX5241 (QFX5241-32OD, QFX5241-64OD, and QFX5241-64QD) and QFX5250-64OE-L switches support the secure boot feature. The secure boot implementation is based on the UEFI 2.4 standard. The BIOS has been hardened and serves as a core root of trust. Implementation of secure boot cryptographically protects the BIOS updates, bootloader, and kernel, preventing tampering. Disabling it

risks security and may brick the device if keys are revoked improperly. For more information about secure boot system, see Secure Boot Overview.

Secure Boot feature is enabled by default. It requires no user configuration to implement. No CLI commands are needed to activate this feature. To verify that your device supports secure boot, use the Feature Explorer and search for `Secure Boot`.

Symmetric IRB using RT2 (MAC-IP)

SUMMARY

This section briefly describes the symmetric integrated and routing bridging with EVPN over VXLAN with Type 2 routes and its configuration on the QFX5220, QFX5230, QFX5240, and QFX5241 Switches.

IN THIS SECTION

Overview

The provider edge (PE) devices in an EVPN fabric can handle traffic to and from the attached multihomed end devices by grouping the multihoming links into an EVPN Ethernet segment with an identifier (ESI). The multihoming links participating in the Ethernet segment are configured into a link aggregation group (LAG). Thus, the set of multihomed links is called as an 'ESI LAG' or EZ-LAG feature. This feature provides a simplified configuration statement hierarchy and a built-in commit script that you can use to set up an EVPN multihoming. This feature makes it easy to migrate a multichassis link aggregation group (MC-LAG) topology to a standards-based EVPN-VXLAN multihoming model. For more information about ESI-LAG, see Benefits of using Easy EVPN LAG Configuration.

Configure Symmetric IRB Using RT2

Juniper devices use asymmetric model with EVPN Type 2 routes by default, or you can enable the symmetric model with EVPN Type 2 routes. EVPN Type 2 symmetric routing is supported as follows:

- In an EVPN-VXLAN, fabric with an edge-routed bridging (ERB) overlay
- EVPN instances are configured using MAC-VRF routing instances with VLAN-based or VLAN-aware bundle service types (see MAC-VRF Routing Instance Type Overview).

To enable symmetric EVPN Type 2 routing on leaf devices in an EVPN-VXLAN fabric with an ERB overlay.

1. On the leaf devices that serve as VTEPs, configure the base MAC-VRF EVPN instances, tenant L3 VRF instances, associated VLANs, and corresponding IRB interfaces for your EVPN-VXLAN network.

2. Configure EVPN Type 5 routing in the EVPN-VXLAN network to establish Layer 3 connectivity between all leaf devices serving as VTEPs.

   In this step, you configure the L3 VRF instances to use EVPN Type 5 routes with a corresponding VNI to advertise the direct next hops for VXLAN traffic. For example:

   ```
   set routing-instances l3-vrf-name protocols evpn ip-prefix-routes advertise direct-nexthop
   set routing-instances l3-vrf-name protocols evpn ip-prefix-routes encapsulation vxlan
   set routing-instances l3-vrf-name protocols evpn ip-prefix-routes vni l3-vni
   ```

   The device uses the L3 VRF EVPN Type 5 VNI as the L3 transit VNI for symmetric EVPN Type 2 routing.

3. Enable symmetric Type 2 routing in the L3 VRF instances using the `irb-symmetric-routing vni vni` statement at the `[edit routing instances name protocols evpn]` hierarchy level.

   For example:

   ```
   set routing-instances l3-vrf-name protocols evpn irb-symmetric-routing
   set routing-instances l3-vrf-name protocols evpn irb-symmetric-routing vni l3-vni
   ```

   Specify the transit VNI value (*l3-vni*) as the same VNI you configure in Step 2 when you enable EVPN Type 5 routing for each L3 VRF.

Verify Symmetric IRB Configuration Using RT2

To verify the configuration of symmetric IRB using RT2:

```
show route table evpn.0
show route table l3-vrf-name.inet.0 protocol evpn
show evpn vni
show interfaces irb terse
show bgp evpn summary
```

Traffic Load Balancing

SUMMARY

This topic briefly describes the traffic load balancing feature and its configuration on the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches.

IN THIS SECTION

Traffic load balancing distributes network traffic across multiple servers or virtual machines to prevent any single server from experiencing downtime. Load balancing ensures high availability, reliability, and scalability of network resources. The different types of traffic load balancing are discussed in the following sections.

Hash-Based (or Flow-Based) Load Balancing

The QFX Series Switches use hash-based load balancing algorithms to distribute and monitor data traffic across links to avoid any congestion. If a packet arrives out of order, it is dropped and retransmission is requested. You can configure hash-based load balancing through equal-cost multipath (ECMP) for BGP.

To configure hash-based load balancing:

```
set protocols bgp multipath multiple-as
set protocols bgp group group multipath
set routing-options forwarding-table export loadbal
set policy-options policy-statement loadbal then load-balance consistent-hash
```

NOTE: Multiple equal-cost static routes enable ECMP for load balancing.

For more information about how to configure hash-based load balancing through ECMP, see Configure IPv4 Static Routing.

Dynamic Load Balancing

In addition to static load balancing, QFX Series Switches also support dynamic load balancing (DLB). DLB is similar to static load balancing, but it also monitors the local ports to know how much buffer each port has. It then redirects the data traffic to the port that has maximum buffer.

To configure DLB on your QFX Series Switches, run the following commands in your CLI at the `[edit]` hierarchy level:

```
set forwarding-options enhanced-hash-key ecmp-dlb flowlet
set forwarding-options enhanced-hash-key ecmp-dlb flowlet inactivity-interval value
set forwarding-options enhanced-hash-key ecmp-dlb flowlet flowset-table-size value
```

Run this code block on each device. For more information about how to configure DLB, see Dynamic Load Balancing.

To verify DLB configuration on your switch:

```
show forwarding-options enhanced-hash-key
```

Selective Dynamic Load Balancing (Not Supported on QFX5220)

Based on dynamic load balancing, selective when combined with "Flexible Match Firewall Filter" on page 58 enables you to apply filters on some of the data flows selected by you.

To configure selective DLB with flexible match firewall filter, run the following commands in your CLI at the `[edit]` hierarchy level:

```
set firewall family inet filter FILTER-NAME term TERM-NAME from flexible-match-range match-start
MATCH-START
set firewall family inet filter FILTER-NAME term TERM-NAME from flexible-match-range byte-offset
BYTE-OFFSET
set firewall family inet filter FILTER-NAME term TERM-NAME from flexible-match-range bit-length
BIT-LENGTH
set firewall family inet filter FILTER-NAME term TERM-NAME from flexible-match-range range RANGE-
VALUE
set firewall family inet filter FILTER-NAME term TERM-NAME then dynamic-load-balance enable
set firewall family inet filter FILTER-NAME term TERM-NAME then count COUNTER-NAME
set interfaces INTERFACE-NAME unit UNIT-NUMBER family inet filter input FILTER-NAME
```

For more information about selective DLB and its configuration, see Selective Dynamic Load Balancing (DLB).

Global Load Balancing

NOTE: Global load balancing is not supported on QFX5220 and QFX5230 switches.

In addition to dynamic load balancing, the QFX5240, QFX5241, and QFX5250 switches also support global load balancing. In case of global load balancing, the spine platform monitors the local port buffers and also sends a signal to the leaf platform with the port buffer status of the platform. The leaf decides the travel path on the basis of its buffer ports but also considers the buffer ports on the next hop, which is the spine platform. So, the global load balancing feature helps the leaf device decide on an end-to-end data travel path within the network fabric only.

To configure global load balancing on QFX5240, QFX5241, and QFX5250 switches:

- On spine devices, run the following commands in `helper-only` mode.

```
set protocols bgp global-load-balancing helper-only
set forwarding-options enhanced-hash-key ecmp-dlb flowlet | per-packet
```

- On leaf devices, run the following commands in `load-balancer-only` mode.

```
set protocols bgp global-load-balancing load-balancer-only
set forwarding-options enhanced-hash-key ecmp-dlb flowlet | per-packet
```

For more information about global load balancing and its configuration, see Global Load Balancing.

RDMA-Aware Load Balancing

Even though remote direct memory access (RDMA)-aware load balancing is supported on the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 switches, we recommend that you enable this feature on the QFX5240, QFX5241, and QFX5250 switches only. The QFX5240, QFX5241, and QFX5250 switches monitor data traffic from one NIC to another, where the switches assign buffer for each data flow from end to end all the way to the NIC. This feature is known as queue-pair hashing, pinning, or binding. For information about how to configure RDMA-aware load balancing on QFX Series Switches, see Fabric Spine and Leaf Nodes Configuration.

For more information about queue-pair hashing configuration, see Queue-Pair Hashing for RDMA Flows.

5
CHAPTER

Use Case Configuration

IN THIS CHAPTER

Use Case Index

SUMMARY

This section provides use cases and related documentation for the QFX5220, QFX5230, QFX5240, QFX5241, and QFX5250 Switches. These use cases help you understand the role of your device when deployed with other platforms in a data center environment.

IN THIS SECTION

The list of use cases and related documentation resources are as follows:

EVPN-VXLAN in a Data Center

Traditional data centers used to have chassis-based switches that were difficult to manage. With changing organizational requirements, data center fabrics are also changing. Nowadays, data centers are expected to support virtualization, span multiple geographies, incorporate hybrid cloud elements, and provide infrastructure for AI workloads. Your data center can meet these requirements by implementing a 3-stage Ethernet VPN-Virtual Extensible LAN (EVPN-VXLAN) fabric architecture with Juniper Apstra. This architecture uses edge-routed bridging (ERB) that distributes traditional network chassis into a switching fabric.

With the QFX5220, QFX5230, QFX5240, and QFX5241 Switches running Junos OS Evolved and using Juniper Apstra as an orchestration platform, you can now provision, manage, and monitor your data centers. The QFX5220 Switch works as a spine transit node in the Ethernet Routing Bridge (ERB) or Border Overlay (BO) design to provide transit connectivity between leaf devices and to aggregate traffic toward the border or edge node. The QFX5230, QFX5240, and QFX5241 Switches support server leaf nodes. These platforms provide the necessary capabilities for leaf-level VXLAN termination and EVPN control plane participation in this architecture.

For more information about the 3-stage EVPN-VXLAN fabric architecture, see 3-Stage Data Center Design with Juniper Apstra (JVD).

To configure EVPN-VXLAN in your data center environment, see EVPN VXLAN Configuration.

Collapsed Spine Switches

A collapsed-spine architecture in a data center combines the functions of spine and leaf switches into a single layer of devices. This architecture thereby eliminates the leaf layer, collapsing its functions onto the spine switches. The collapsed spine switches perform underlay-routing and EVPN-VXLAN overlay functions.

NOTE: This use case is not applicable to QFX5250 switches.

An enterprise that doesn't require a 3-stage data center network might opt for collapsed spine switches in its data center. Deployment scenarios include small, high-availability data center networks, single-rack pods within a larger data center, remote sites and branch offices, and deployments with low budget, space, or power constraints.

For more information about this use case, see Collapsed Data Center Fabric with Juniper Apstra—Juniper Validated Design (JVD).

To configure this use case in your data center environment, see Collapsed Spine Switches Configuration.

AI Cluster Networking

To meet the bandwidth requirement of artificial intelligence–machine learning (AI-ML) workloads and storage systems, you can use QFX5230 and higher-numbered switches. Most switches offer 51.2-Tbps throughput with 1600GbE, 800GbE, 400GbE, 200GbE, 100GbE, and 50GbE interfaces and high port density, supporting intra-IP fabric and network interface card (NIC) connectivity for AI-ML use cases. These switches support remote direct memory access (RDMA) and Remote Direct Memory Access over Converged Ethernet version 2 (ROCEv2) with congestion management features such as priority-based flow control (PFC), explicit congestion notification (ECN), and data center quantized congestion notification (DCQCN). All these features are useful in processing AI-ML workloads that transfer data at the network layer.

NOTE: If you are new to AI-ML data center networking, see AI-ML Data Center Overview.

For information about this use case, see AI Data Center Network with Juniper Apstra, Nvidia GPUs, ConnectX NIC, and Weka Storage—Juniper Validated Design (JVD).

EVPN-VXLAN in AI Cluster Networking

The QFX5230, QFX5240, and QFX5241 switches can have EVPN-VXLAN as an overlay over a pure IP fabric. This setup supports leaf, spine, and super-spine deployments for AI data centers. This cluster supports either a 3-stage IP fabric architecture or a 3-stage EVPN-VXLAN fabric architecture.

For information about how to build an AI data center, see AI Data Center Network with Juniper Apstra, AMD GPUs, Broadcom Thor2 NIC, AMD Pollara NIC, and Vast Storage—Juniper Validated Design (JVD).

6
CHAPTER

References and Further Reading

IN THIS CHAPTER

Documentation Tools and Resources

SUMMARY

This section provides information about the documentation tools and resources.

IN THIS SECTION

Documentation Tools

Following are the Juniper documentation tools that you can use to learn more about QFX5200 switches.

- Feature Explorer- Use this tool to learn about the software features that run on your hardware platforms in relation to Junos OS and Junos OS Evolved releases.
- Hardware Explorer - Use this tool to simplify and streamline your hardware planning, configuration, and optimization.
  - Hardware Specifications - Use this tool to deep dive into detailed specs for Juniper devices.
  - Hardware Compatibility Tool- Use this tool to find information about the transceivers, line cards, and interface modules that are supported on QFX5200 switches and other Juniper products.
  - Port Checker - Use this tool for a visual representation of various Juniper network devices, and assistance to configure and validate different port combinations.
  - Power Calculator - Use this tool to calculate the maximum power a product can consume by dynamically configuring different components.
- CLI Explorer- Use this tool to explore configuration statements and commands in Junos OS and Junos OS Evolved.

Resources

The following list of resources might be useful to you when working on QFX5200 switches:

- **Certifications:** If your are interested in buying any QFX5200 switch and want that switch to be exported to your country then in that case you are required to obtain product compliance. For product compliance, you may need to undertake bunch of certifications like RoHS, NEBS etc. You can write to juniper-epc@juniper.net for certifications including your sales account team as part of that email (in case of any delay, sales account team can help).

- **Data Sheets**
  - QFX5250-64OE-L
  - QFX5241-64OD and QFX5241-64QD
  - QFX5240 Line of Switches Datasheet
  - QFX5230-64CD
  - QFX5220
  - QFX5210
  - QFX5200
- **Quick Start Guides**
  - Onboarding Data Center Switches with Apstra
  - Fast Track to Rack Installation and Power
  - Perform Initial Software Configuration for QFX5240 Switches
- **Day 1 Guide:** DAY ONE: DEPLOY A DATA CENTER FABRIC WITH APSTRA AND JUNOS