



Getting Started with NFX250 Network Services Platform



Modified: 2018-11-13

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

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

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

Getting Started with NFX250 Network Services Platform
Copyright © 2019 Juniper Networks, Inc. All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

	About the Documentation	ix
	Documentation and Release Notes	ix
	Using the Examples in This Manual	ix
	Merging a Full Example	x
	Merging a Snippet	x
	Documentation Conventions	xi
	Documentation Feedback	xiii
	Requesting Technical Support	xiii
	Self-Help Online Tools and Resources	xiv
	Creating a Service Request with JTAC	xiv
Part 1	Understanding NFX250 Network Services Platform	
Chapter 1	Overview	3
	NFX250 Overview	3
	Benefits of NFX250 Network Services Platform	3
	NFX250 Models	4
	NFX250 Components	5
Part 2	Configuring the NFX250 Network Services Platform	
Chapter 2	Configuration Tasks	9
	Performing Initial Software Configuration on an NFX250 Device	9
	Configuring the In-band Management Network Connection for NFX250	12
	Pre-allocating Hugepages in the System	13
	Creating the vSRX VNF on the NFX250 Platform	14
	Configuring the vMX Virtual Router as a VNF on NFX250	16
	Configuring the Internal Management IP Address of vSRX VNF	18
	ADSL2 and ADSL2+ Interfaces on NFX250 Devices	20
	ADSL Interface Overview	20
	ADSL2 and ADSL2+	20
	Example: Configuring ADSL SFP Interface on NFX250 Devices	21
	VDSL2 Interfaces on NFX250 Devices	24
	VDSL Interface Overview	25
	VDSL2 Vectoring Overview	25
	VDSL2 Network Deployment Topology	25
	VDSL2 Interface Support on NFX Series Devices	27
	VDSL2 Interface Compatibility with ADSL Interfaces	27
	VDSL2 Interfaces Supported Profiles	27
	Example: Configuring VDSL SFP Interface on NFX250 Devices	28

List of Figures

Part 2	Configuring the NFX250 Network Services Platform	
Chapter 2	Configuration Tasks	9
	Figure 1: Network Service Orchestrator module	11
	Figure 2: NFX250 Front Panel Components	12
	Figure 3: Typical VDSL2 End-to-End Connectivity and Topology Diagram	26
	Figure 4: Backward-Compatible ADSL Topology (ATM DSLAM)	26

List of Tables

	About the Documentation	ix
	Table 1: Notice Icons	xi
	Table 2: Text and Syntax Conventions	xii
Part 2	Configuring the NFX250 Network Services Platform	
Chapter 2	Configuration Tasks	9
	Table 3: Standard Bandwidths of DSL Operating Modes	20
	Table 4: VDSL2 Annex A and Annex B Features	27
	Table 5: Supported Profiles on the VDSL2 Interfaces	28

About the Documentation

- Documentation and Release Notes on page ix
- Using the Examples in This Manual on page ix
- Documentation Conventions on page xi
- Documentation Feedback on page xiii
- Requesting Technical Support on page xiii

Documentation and Release Notes

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

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

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

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see [CLI Explorer](#).

Documentation Conventions

Table 1 on page xi defines notice icons used in this guide.

Table 1: Notice Icons







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

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

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: user@host> configure
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> show chassis alarms No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> Introduces or emphasizes important new terms. Identifies guide names. Identifies RFC and Internet draft titles. 	<ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS CLI User Guide</i> RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. The console port is labeled CONSOLE.
< > (angle brackets)	Encloses optional keywords or variables.	stub <default-metric <i>metric</i> >;
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast multicast (<i>string1</i> <i>string2</i> <i>string3</i>)
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[] (square brackets)	Encloses a variable for which you can substitute one or more values.	community name members [community-ids]
Indentation and braces ({ })	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	

GUI Conventions

Table 2: Text and Syntax Conventions (continued)

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

Documentation Feedback

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

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



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

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

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

Self-Help Online Tools and Resources

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

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

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

Creating a Service Request with JTAC

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

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

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

PART 1

Understanding NFX250 Network Services Platform

- [Overview on page 3](#)

CHAPTER 1

Overview

- [NFX250 Overview on page 3](#)

NFX250 Overview

The Juniper Networks NFX250 Network Services Platform is a secure, automated, software-driven customer premises equipment (CPE) platform that delivers virtualized network and security services on demand. An integral part of Juniper's fully automated Cloud CPE solution suite for NFV, this high-performance virtualized services platform helps service providers improve overall operational efficiency and service agility while empowering enterprise customers with immediate access to custom-designed managed services. Simultaneously supporting multiple Juniper and third-party VNFs on a single device and providing built-in, dynamic, policy-based routing, the NFX250 addresses the needs of small to midsize businesses as well as large multinational or distributed enterprises with a single, highly scalable solution.

NFX250 Network Services Platform are Juniper Network's secure, automated, software-driven customer premises equipment (CPE) devices that deliver virtualized network and security services on demand. Leveraging Network Functions Virtualization (NFV) and built on the Juniper Cloud CPE solution, NFX250 enables service providers to deploy and service chain multiple, secure, high-performance virtualized network functions (VNFs) as a single device. This automated, software-driven solution dynamically provisions new services on demand.

- [Benefits of NFX250 Network Services Platform on page 3](#)
- [NFX250 Models on page 4](#)
- [NFX250 Components on page 5](#)

Benefits of NFX250 Network Services Platform

The NFX250 is an integrated branch router and switch with a multicore CPU that enables it to run multiple Virtual Network Functions (VNFS). The NFX250 Network Services Platform provides these benefits:

- Simultaneously supports multiple Juniper and third-party VNFs on a single device, providing built-in, dynamic, and policy-based routing
- Provides an open framework that supports industry standards, protocols, and seamless API integration

- Supports a variety of flexible deployments. A distributed services deployment model ensures high availability, performance, and compliance
- Incorporates many advanced security features. Secure Boot feature safeguards device credentials, automatically authenticates system integrity, verifies system configuration, and enhances overall platform security
- Modular software architecture provides high performance and scalability for routing, switching, and security enhanced by carrierclass reliability
- Automated configuration eliminates complex device setup and delivers a plug-and-play experience
- High performance simplifies network topologies and operations

NFX250 Models

The NFX250 device is available in three models.

Product Number	Specifications	Features
NFX250-S1	<p>2.0 GHz 6-core Intel CPU</p> <p>16 GB of memory and 100 GB of solid-state drive (SSD) storage</p> <p>Eight 1-GbE network ports, two 1-GbE RJ-45 ports which can be used as either access ports or as uplinks, two SFP ports, two SFP+ ports, one Management port, and two Console ports</p>	Basic Layer 2/Layer 3
NFX250-S2	<p>2.0 GHz 6-core Intel CPU</p> <p>32 GB of memory and 400 GB of SSD storage</p> <p>Eight 1-GbE network ports, two 1-GbE RJ-45 ports which can be used as either access ports or as uplinks, two SFP ports, two SFP+ ports, one Management port, and two Console ports</p>	Basic Layer 2/Layer 3
NFX250-LS1	<p>1.6 GHz 4-core Intel CPU</p> <p>16 GB of memory and 100 GB of solid-state drive (SSD) storage</p> <p>Eight 1-GbE network ports, two 1-GbE RJ-45 ports which can be used as either access ports or as uplinks, two SFP ports, two SFP+ ports, one Management port, and two Console ports</p>	<p>Supports up to 100 MBPS throughput Secure Router functionality for the following features:</p> <ul style="list-style-type: none"> • IPSec VPN • NAT • Stateful Firewall • Routing services – BGP, OSPF, DHCP, IPv4 and IPv6

Product Number	Specifications	Features
NFX250-S1E	<p>2.0 GHz 6-core Intel CPU</p> <p>16 GB of memory and 200 GB of solid-state drive (SSD) storage</p> <p>Eight 1-GbE network ports, two 1-GbE RJ-45 ports which can be used as either access ports or as uplinks, two SFP ports, two SFP+ ports, one Management port, and two Console ports</p>	<p>Supports up to 100 MBPS throughput Secure Router functionality for the following features:</p> <ul style="list-style-type: none"> • IPSec VPN • NAT • Stateful Firewall • Routing services – BGP, OSPF, DHCP, IPv4 and IPv6

NFX250 Components

The NFX250 consists of the following key components:

- *Juniper Device Manager*: The Juniper Device Manager (JDM) is a low-footprint Linux container that provides these functions:
 - Virtual Machine (VM) lifecycle management
 - Device management and isolation of host OS from user installations
 - NIC, single-root I/O virtualization (SR-IOV), and virtual input/output (VirtIO) interface provisioning
 - Support for the Network Service Orchestrator module to connect to Network Activator
 - Inventory and resource management
 - Internal network and image management
 - Service chaining—provides building blocks such as virtual interfaces and bridges for users to implement service chaining policies
 - Virtual console access to VNFs including vSRX and vjunos
- *Network Service Orchestrator Module*: Network Service Orchestrator module is a client included in the base software of the NFX250, which connects to the Network Activator deployed in a data center that serves your location. The Network Activator intelligently automates service life cycle management on the NFX250 across managed VPN networks, in-region secured Internet connections, and out-of-region IPsec connections. This application enables the boot-up and configuration of the NFX250 device when it first powered on. For details, see the Network Activator documentation at <https://www.juniper.net/documentation/>.
- *vSRX*: vSRX provides the same capabilities as Juniper Networks SRX Series Services Gateways in a virtual form factor, providing perimeter security, IPsec connectivity, and filtering for malicious traffic without sacrificing reliability, visibility, or policy control. This virtual security and routing appliance ensures reliability and high availability for each application. For details, see the vSRX documentation at <https://www.juniper.net/documentation/>.
- *Junos Control Plane*: Junos Control Plane (JCP) is the Junos VM running on the hypervisor. You can use JCP to configure the network ports of the NFX250 device, and JCP runs

by default as **vjunos0** on NFX250. You can log on to JCP from JDM using the SSH service and the command-line interface (CLI) is the same as Junos.

**Related
Documentation**

- [Performing Initial Software Configuration on an NFX250 Device on page 9](#)

PART 2

Configuring the NFX250 Network Services Platform

- [Configuration Tasks on page 9](#)

CHAPTER 2

Configuration Tasks

- [Performing Initial Software Configuration on an NFX250 Device on page 9](#)
- [Configuring the In-band Management Network Connection for NFX250 on page 12](#)
- [Pre-allocating Hugepages in the System on page 13](#)
- [Creating the vSRX VNF on the NFX250 Platform on page 14](#)
- [Configuring the vMX Virtual Router as a VNF on NFX250 on page 16](#)
- [Configuring the Internal Management IP Address of vSRX VNF on page 18](#)
- [ADSL2 and ADSL2+ Interfaces on NFX250 Devices on page 20](#)
- [VDSL2 Interfaces on NFX250 Devices on page 24](#)

Performing Initial Software Configuration on an NFX250 Device

You must perform the initial configuration of the NFX250 device through the console port using the Juniper Device Manager (JDM) command-line interface (CLI).



NOTE: Note that there are changes in the CLI commands. The CLI configuration commands for Release 15.1X53-D40 may not be applicable for this release.

Before you begin connecting and configuring an NFX250 device, set the following parameter values on the console server or PC:

- Baud Rate—9600
- Flow Control—None
- Data—8
- Parity—None
- Stop Bits—1
- DCD State—Disregard

To connect and configure the device from the console:

1. Connect the console port to a laptop or PC using the supplied RJ-45 cable and RJ-45 to DB-9 adapter. The console (**CON**) port is located on the management panel of the device.



NOTE: See the procedure after Step 11 for details on the Network Service Orchestrator process.

2. The Juniper Device Manager (JDM) command-line interface (CLI) displays; log in as **root**. There is no password. If the software booted before you connected to the console port, you might need to press the Enter key for the prompt to appear.

```
login: root
```

3. Start the CLI.

```
root@jdm% cli
```

4. Enter configuration mode.

```
root@jdm> configure
```

5. Add a password to the root administration user account.

```
[edit]
root@jdm# set system root-authentication plain-text-password
New password: password
Retype new password: password
```

6. (Optional) Configure the name of the device. If the name includes spaces, enclose the name in quotation marks (" ").

```
[edit]
root@jdm# set system host-name host-name
```

7. Configure the IP address and prefix length for the device management interface.

```
[edit]
root@jdm# # set interfaces jmgmt0 unit 0 family inet address address/prefix length
jmgmt0 is the out-of-band management network interface in JDM.
```

To configure an IPV6 address, run the **root@jdm# set interfaces jmgmt0 unit 0 family inet6 address *v6_address***.



NOTE: **jmgmt0** is located on front panel port of the NFX250 device.

8. Configure the default gateway.


```
[edit]
root@jdm# set routing-options static route default next-hop address
```

9. Commit the configuration to activate it on the device.

```
[edit]
root@jdm# commit
```

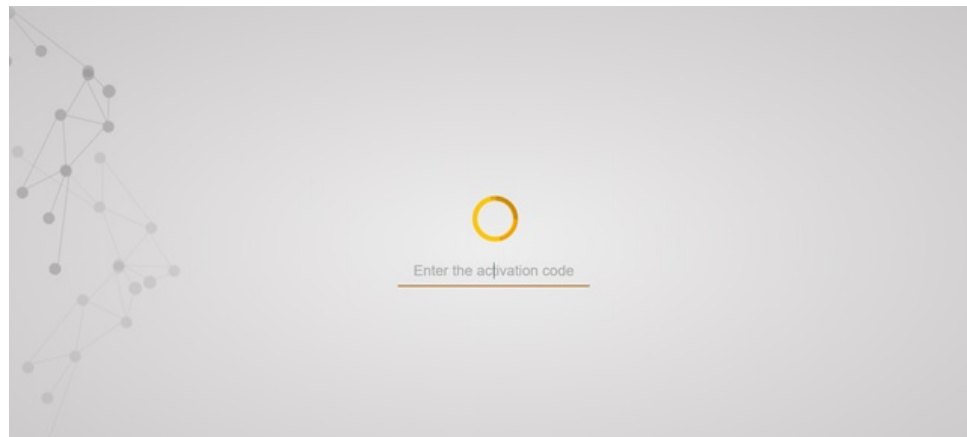
If Network Service Orchestrator module is configured, this client connects to the Network Activator as soon as the device is switched on, and provisions the initial configuration and the latest software image and, if the image on the NFX250 device is not the latest.

Network Activator is responsible for the bare-minimum bootstrapping of the NFX250. After successful configuration and software upgrade, the device reboots and the Network Activator configuration is removed.

To complete the configuration of the Network Service Orchestrator module process:

1. Connect to any front panel WAN port (see [Figure 2 on page 12](#)).
2. Open web browser and enter the IP address 10.10.10.1.
3. Enter the authentication code in the Web page that is displayed.

Figure 1: Network Service Orchestrator module



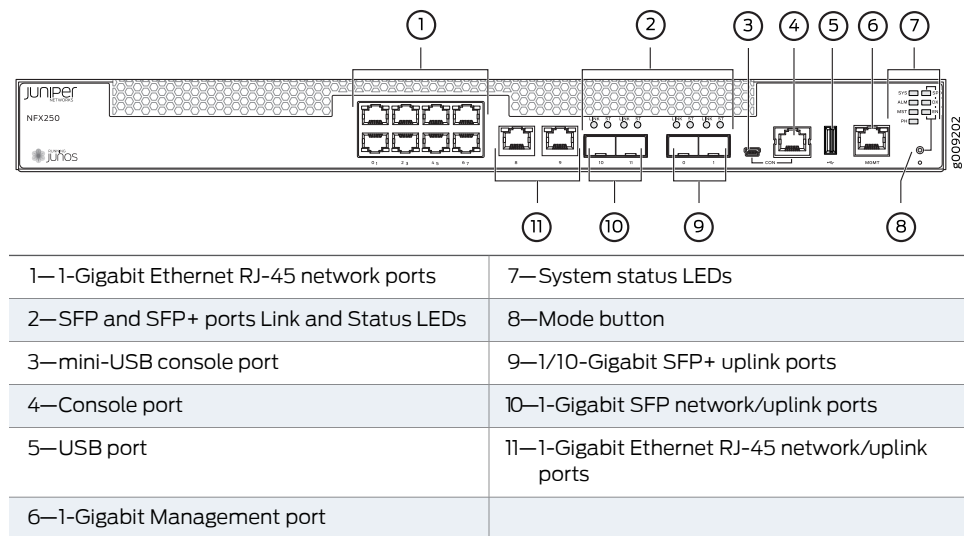
Once the process is complete, a confirmation message is displayed. Click Logs to display details of the bootstrapping process. Refer to *Captive Portal Log Messages* for the list of log messages that are displayed.



NOTE: You can also use the CLI to provide the authentication code:

```
root@jdm> test phone-home server-authentication-code code
```

Figure 2: NFX250 Front Panel Components



- Related Documentation**
- [NFX250 Overview on page 3](#)
 - [Benefits of NFX250 Network Services Platform](#)

Configuring the In-band Management Network Connection for NFX250

Juniper Device Manager (JDM), an in-band management network interface—**jsxe0**. This internal interface is not directly connected to a physical interface. You must link jsxe0 to a physical interface through VLAN bridging—that is, you must configure both the physical interface and jsxe0 to be in the same management VLAN. JCP manages the physical network interfaces and the service interfaces, not JDM; therefore, you must first configure the sxe-0/0/0 and sxe-0/0/1 internal interfaces using the JCP CLI before you can manage the jsxe0 interface using the JDM CLI.



NOTE: Choose the management VLAN ID to ensure that only the management traffic is directed to JDM.

To configure the in-band management connection:

1. Log on to JCP from the JDM command-line interface (CLI).

```
[edit]
root@jdm> ssh vjunos0
The JCP CLI displays, which is same as the Junos CLI.
```

2. Configure the physical network port as a trunk port.

```
user@switch# set interfaces interface-name unit 0 family ethernet-switching interface-mode trunk
```

3. Configure a JCP service port as a trunk port.

```
user@switch# set interfaces service-interface-name unit 0 family ethernet-switching
interface-mode trunk
```

4. Configure the management VLAN and add the physical network interface and the service interface as members of the VLAN.

```
user@switch# set interfaces service-interface-name unit 0 family ethernet-switching
interface-mode trunk
user@switch# set interfaces service-interface-name unit 0 family ethernet-switching vlan
members mgmt-vlan
user@switch# set interfaces interface-name unit 0 family ethernet-switching vlan members
mgmt-vlan
```

5. Exit JCP and return to the JDM CLI.

```
[edit]user@switch# exit
Exiting configuration mode
root> exit
root% exit
Logout
Connection to vjunos0 closed.
root@jdm>
```

6. Configure the jsxe0 interface as a trunk interface with membership in the management VLAN, and configure the management IP address on the interface:

```
[edit]
root@jdm# set interfaces jsxe0 vlan-tagging
root@jdm# set interfaces jsxe0 unit logical-unit-number vlan-id mgmt-vlan-id family inet
address mgmt-ip-address/prefix-length
```

Related Documentation • [NFX250 Overview on page 3](#)

Pre-allocating Hugepages in the System

You can reserve the required number of hugepages based on the memory required by the VNF. To do so:

```
user@jdm# set system memory hugepages page-size page-size page-count page-count
```

Page-size and page-count values depends on the size and total number of hugepages required by all the VNFs that will be launched in the system. Since system memory may be already fragmented, it is recommended that you reboot the system to pre-allocate hugepages during bootstrap.

Creating the vSRX VNF on the NFX250 Platform

vSRX is a virtual security appliance that provides security and networking services in virtualized private or public cloud environments. It can be run as a virtual network function (VNF) on the NFX250 platform. For more details on vSRX, see the product documentation page on the Juniper Networks website at <https://www.juniper.net/>.

To activate the vSRX VNF from the Juniper Device Manager (JDM) command-line interface:

1. Define VLANs required for vSRX VNF interfaces. For example:

```
set host-os vlans v1 vlan-id 2614
set host-os vlans v2 vlan-id 2615
set host-os vlans v3 vlan-id 2714
set host-os vlans v4 vlan-id 2715
```

2. Define any glue VLANs required for the vSRX VNF interfaces. For example:

```
set host-os vlans gluebr0 vlan-id 2814
set host-os vlans gluebr1 vlan-id 2815
```

3. Define vSRX VNF with vSRX image. For example:

```
set virtual-network-functions vsrx image /var/third-party/images/vsrx.qcow2
```

4. (Optional) Create the vSRX VNF with groups that contain custom configuration. For example:

```
set virtual-network-functions vsrx apply-groups junos-vsrx
```

5. Map the vSRX VNF interfaces to VLANs or glue-VLANs. For example:

```
set virtual-network-functions vsrx interfaces eth2 mapping vlan members v1
set virtual-network-functions vsrx interfaces eth3 mapping vlan members v2
set virtual-network-functions vsrx interfaces eth4 mapping vlan members v3
set virtual-network-functions vsrx interfaces eth5 mapping vlan members v4
set virtual-network-functions vsrx interfaces eth6 mapping vlan members gluebr0
set virtual-network-functions vsrx interfaces eth7 mapping vlan members gluebr1
```

6. Specify a mode for the vSRX VNF interfaces. The interface mode can be either access or trunk mode. For example:

```
set virtual-network-functions vsrx interfaces eth2 mapping vlan mode trunk
```

7. Specify the maximum transmission unit (MTU) size for the media in bytes for vSRX VNF interfaces. MTU size can be either 1500 bytes or 2048 bytes. For example:

```
set virtual-network-functions vsrx interfaces eth2 mtu 1500
```

8. Specify the target PCI address for the VNF interface. For example:

```
set virtual-network-functions vsrx interfaces eth2 pci-address pci-address
```

9. At the CLI prompt, enter the **commit** command to activate the vSRX VNF.

```
[edit]
root# commit
```

10. Attach the ISO to vSRX as a CD-ROM device and start vSRX.

```
[edit]
root@jdm# set virtual-network-functions vsrx storage hdb type cdrom source file
/var/third-party/iso/testcd/bootstrapconf.iso
```



NOTE: If a vSRX instance is running, you must restart the instance so that the new configuration is applied from the CD-ROM.

11. (Optional) To create the vSRX VNF with a custom bootstrap configuration, create an ISO image with the configuration file **juniper.conf**.

```
[edit]
root@jdm> request genisoimage /var/third-party/iso/testcd/juniper.conf
/var/third-party/iso/testcd/bootstrapconf.iso
ISO image "/var/third-party/iso/testcd/bootstrapconf.iso" successfully
generated from "/var/third-party/iso/testcd/juniper.conf".
```



NOTE: Ensure that the configuration file is named **juniper.conf**.

12. Verify if the vSRX VNF initiated correctly. You can use JDM cli or Linux virsh commands to verify.

```
[edit]
root@jdm# run show virtual-network-functions
```

ID	Name	State	Liveliness
2	vjunos0	running	alive
12	vsrx	running	alive
7433	jdm running	alive	

Using the Linux virsh command

```
[edit]
root@jdm# virsh list
```

ID	Name	State
2	vjunos0	running
3	vsrx	running

You can see that the vSRX VNF is active.

13. SSH connection to vSRX works only if the liveliness in the show output shows the status **alive**, that is if bootstrap iso config was used to enable DHCP on fxp0 interface of vSRX to get the internal management IP address). If the liveliness status for vSRX VNF is **down**, refer to [“Configuring the Internal Management IP Address of vSRX VNF” on page 18](#).

To log on to the vSRX VNF, enter the command **run ssh vsrx**.

14. (Optional) Verify the vSRX VNF details.

```
root@jdm> show virtual-network-functions vsrx
Virtual Machine Information
-----
Name:                vsrx
IP Address:          192.168.1.4
Status:              Running
Liveliness:          Up
VCPUs:               2
Maximum Memory:      4000768
Used Memory:         4000768
Virtual Machine      Block Devices
-----
Target    Source
-----
hda       /var/third-party/images/vsrx.qcow2
hdf       /var/third-party/iso/testcd/bootstrapconf.iso
```

Related Documentation • [NFX250 Overview on page 3](#)

Configuring the vMX Virtual Router as a VNF on NFX250

The vMX router is a virtual version of the Juniper MX Series 5G Universal Routing Platform. To quickly migrate physical infrastructure and services, you can configure vMX as a virtual network function (VNF) on the NFX250 platform. For more details on the configuration and management of vMX, see [vMX Overview](#).

Before you configure the VNF, check the system inventory and confirm that the required resources are available. vMX as VNF must be designed and configured so that its resource requirements do not exceed the available capacity of the system. Ensure that a minimum of 20 GB space is available on NFX250.

To configure vMX as VNF on NFX250 using the Juniper Device Manager (JDM) command-line interface (CLI):

1. Download the nested image available at **vmx-nested-< release>.qcow2**.
2. Create a new disk image for vMX.

```
%qemu-img create -f qcow2 junos_vmx_hdd.qcow2 20G
```

3. Define VLANs required for the vMX VNF interfaces. For example:

```
user@host# set host-os vlans v1 vlan-id 2614
user@host# set host-os vlans v2 vlan-id 2615
```

4. Define the glue VLANs required for the vMX VNF interfaces. For example:

```
user@host# set host-os vlans gluebr0 vlan-id 2614
user@host# set host-os vlans gluebr1 vlan-id 2615
```

5. Define vMX for VNF with the vMX image. For example:

```
user@host# set virtual-network-functions vmx image
/var/third-party/images/vmx-nested-<release>.qcow2
```

6. Specify the maximum primary memory that the VNF can use. For optimal performance, it is recommended to configure with at least 5 GB memory.

```
user@host# set virtual-network-functions vmx memory size < n >
```

7. Specify the number of cores per CPU in a virtual machine. For vMX VNF, you need a minimum of 4 virtual CPU cores.

```
user@host# set virtual-network-functions vmx virtual-cpu count < n > features
hardware-virtualization
```

8. Add an additional data drive that stores the configuration parameters.

```
user@host# set virtual-network-functions vmx storage vdc type disk file-type
vmx-nested-< release>.qcow2
```

9. Map the vMX VNF interfaces to VLANs or glue-VLANs.

```
user@host# set virtual-network-functions vmx interfaces eth2 description wan0
```

```
user@host# set virtual-network-functions vmx interfaces eth2 mapping vlan members
< vlan >
```

```
user@host# set virtual-network-functions vmx interfaces eth3 description wan1
```

```
user@host# set virtual-network-functions vmx interfaces eth3 mapping vlan members
< vlan >
```

10. At the CLI prompt, enter the **commit** command to activate vMX VNF.

```
user@host# commit
```

11. Verify if the vMX VNF has been configured correctly on NFX250.

```
root@jdm# run show virtual-network-functions
```

ID	Name	State	Liveliness
3	vjunos0	running	alive

10	vmx		running	running	alive
11341	jdm	running		alive	

If you use virsh, enter

```
root@jdm# virsh list
ID      Name                                State
-----
2       vjunos0                            running
3       vmx                                running
```

This shows that the vMX VNF is active.

12. Verify if the vMX VNF has been configured correctly on NFX250.

```
root@jdm# run show virtual-network-functions
```

To upgrade the vMX VNF, deactivate the VNF configuration and select the new image copied to the `/var/third-party/images/vmx-nested-< release>.qcow2` location. Reactivate the VNF configuration again.

13. For in-band management network connections, the assigned management port is fxp0. For out-of-band management, ge-0/0/0 is used, and ge-0/0/1 is used for WAN interfaces.

- Related Documentation**
- [NFX250 Overview on page 3](#)
 - [JDM User Guide for NFX250 Network Services Platform](#)

Configuring the Internal Management IP Address of vSRX VNF

VNF internal management IP addresses (192.168.x.x) on the default network (virbr0 bridge) are assigned through DHCP, and /etc/hosts updates handled automatically.



NOTE: You can log on to the vSRX VNF through SSH only after configuring DHCP on the fxp0 interface so that the internal management IP address is set on the fxp0 interface.

To configure the internal management IP address on the fxp0 interface of vSRX VNF.:

1. Verify if the vSRX VNF is successfully created.

```
root@jdm# /var/third-party# virsh list
```

Id	Name	State
2	vjunos0	running
3	vsrx	running

You can see that the vSRX VM is running.

2. Log on to the vSRX VNF console.

```
root@jdm# ~# virsh -e$ console vsrx
```

login:



NOTE:

- Ensure that the escape character is not as same as the character used for any other purpose in the vSRX console.
- You can use this virsh console to log on to the console of any VM for troubleshooting, if other login methods fail.

3. Log in as **root**.
4. Enter the configuration mode.

```
root@user-vsrx>configure
```

5. Change the default vSRX root password after you log in to the console:

```
[edit]
root@user-vsrx# set system root-authentication password
```

6. Enable DHCP on fxp0 interface and commit the configuration.

```
[edit]
root@user-vsrx# set interfaces fxp0 unit 0 family inet dhcp
```

```
[edit]
root@user-vsrx# commit
```

7. Use the character **#** to return to the JDM CLI.
8. Verify if fxp0 interface is up.

```
root@jdm>show virtual-network-functions
```

ID	Name	State	Liveliness
2	vjunos0	running	alive
3	vsrx	running	alive
9192	jdm	running	alive

The status **alive** indicates that the IP address has been assigned successfully.

You can now establish an SSH connection to the vSRX VNF by running the **ssh vsrx** command.

Related Documentation

- [NFX250 Overview on page 3](#)

ADSL2 and ADSL2+ Interfaces on NFX250 Devices

- [ADSL Interface Overview on page 20](#)
- [Example: Configuring ADSL SFP Interface on NFX250 Devices on page 21](#)

ADSL Interface Overview

Asymmetric digital subscriber line (ADSL) technology is part of the xDSL family of modem technologies that use existing twisted-pair telephone lines to transport high-bandwidth data. ADSL lines connect service provider networks and customer sites over the "last mile" of the network—the loop between the service provider and the customer site.

ADSL transmission is asymmetric because the downstream bandwidth is typically greater than the upstream bandwidth. The typical bandwidths of ADSL2 and ADSL2+ circuits are defined in [Table 3 on page 20](#).

Table 3: Standard Bandwidths of DSL Operating Modes

Operating Modes	Upstream	Downstream
ADSL2	1—1.5 Mbps	12—14 Mbps
ADSL2+	1—1.5 Mbps	24—25 Mbps

ADSL2 and ADSL2+ support the following standards:

- LLC SNAP bridged 802.1q
- VC MUX bridged

Supported security devices with xDSL SFP can use PPP over Ethernet (PPPoE) to connect through ADSL lines only.

ADSL2 and ADSL2+

The ADSL2 and ADSL2+ standards were adopted by the ITU in July 2002. ADSL2 improves the data rate and reach performance, diagnostics, standby mode, and interoperability of ADSL modems.

ADSL2+ doubles the possible downstream data bandwidth, enabling rates of 20 Mbps on telephone lines shorter than 5000 feet (1.5 km).

ADSL2 uses seamless rate adaptation (SRA) to change the data rate of a connection during operation with no interruptions or bit errors. The ADSL2 transceiver detects changes

in channel conditions—for example, the failure of another transceiver in a multicarrier link—and sends a message to the transmitter to initiate a data rate change. The message includes data transmission parameters such as the number of bits modulated and the power on each channel. When the transmitter receives the information, it transitions to the new transmission rate.

Example: Configuring ADSL SFP Interface on NFX250 Devices

- [Requirements on page 21](#)
- [Overview on page 21](#)
- [Configuration on page 21](#)
- [Results on page 24](#)

Requirements

This example uses the following hardware and software components:

- NFX250 device running Junos OS Release 15.1X53-D495.

Overview

In this example, you are configuring ADSL SFP interface on an NFX250 device with the following configurations:

- Physical interface - **ge-0/0/11**
- Virtual network function (VNF) - **nfx250-a-vsrx1**
- Memory size - **4194304**
- ADSL SFP options - **vpi3, vci34, and encaps llcsnap-bridged-802dot1q**

To configure ADSL SFP interface on NFX250 devices, you must configure JDM, vSRX, and vJunos0.



NOTE: Ensure that connectivity to the host is not lost during the configuration process.

Configuration

Step-by-Step Procedure

To configure ADSL SFP interfaces on NFX250 devices:

1. Connect to the host.

```
user@host> configure
[edit]
user@host#
```

2. Create VLANs using VLAN IDs:

```
user@host# set host-os vlans xdsl-test vlan-id 50
```

```
user@host# set host-os vlans vlan130 vlan-id 130
user@host# set host-os vlans vlan131 vlan-id 131
user@host# set host-os vlans vlan132 vlan-id 132
```

3. Enable enhanced orchestration to manage VNFs and service chains without requiring the VNF XML descriptor files:

```
user@host# set system services enhanced-orchestration
```

4. Allocate resources for a VNF:

```
user@host# set virtual-network-functions nfx250-a-vsrx1 image
/var/third-party/images/media-vsrx-vmdisk-15.1-2018-04-24.0_DEV_X_151_X49.qcow2
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu 0 physical-cpu 2
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu 1 physical-cpu 6
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu count 2
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu features
hardware-virtualization
user@host# set virtual-network-functions nfx250-a-vsrx1 no-default-interfaces
user@host# set virtual-network-functions nfx250-a-vsrx1 memory size 4194304
user@host# set virtual-network-functions nfx250-a-vsrx1 memory features hugepages
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth0 management
out-of-band
```

5. Map physical interfaces to virtual interfaces:

```
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth1 mapping vlan
mode trunk
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth1 mapping vlan
members vlan130
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
mode trunk
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members vlan130
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members vlan131
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members vlan132
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members xdsl-test
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
native-vlan-id 50
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth3 mapping vlan
mode trunk
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth3 mapping vlan
members xdsl-test
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth3 mapping vlan
native-vlan-id 50
```

6. Configure the Junos Control Plane (JCP) virtual machine (VM):

```
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching interface-mode
trunk
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
xdsl-test
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
vlan130
```

```
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
vlan131
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
vlan132
user@host# set interfaces ge-0/0/11 native-vlan-id 50
user@host# set interfaces ge-0/0/11 dsl-sfp-options adsl-options vpi 3
user@host# set interfaces ge-0/0/11 dsl-sfp-options adsl-options vci 34
user@host# set interfaces ge-0/0/11 dsl-sfp-options adsl-options encaps
llcsnap-bridged-802dot1q
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members
xdsl-test
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members vlan130
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members vlan131
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members vlan132
user@host# set vlans vlan130 vlan-id 130
user@host# set vlans vlan131 vlan-id 131
user@host# set vlans vlan132 vlan-id 132
user@host# set vlans xdsl-test vlan-id 50
```

7. Commit the configuration.

```
user@host# commit and-quit
user@host> exit
```

Results

From configuration mode, verify your configuration by entering the **show interfaces ge-0/0/11** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it:

```
[edit]
user@host# show interfaces ge-0/0/11

Physical interface: ge-0/0/11, Enabled, Physical link is Up
  Interface index: 258, SNMP ifIndex: 533
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, ADSL2P mode,
  Speed: 1000Mbps, BPDU Error: None, Loop Detect PDU Error: None,
  Ethernet-Switching Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online, IEEE 802.3az Energy Efficient Ethernet: Disabled,
  Auto-MDIX: Enabled
  ADSL status:
    Modem status   : Showtime (Adsl2plus)
    DSL mode       :      Auto      Annex A
  Device flags    : Present Running
  Interface flags : SNMP-Traps Internal: 0x4000
  Link flags      : None
  CoS queues      : 12 supported, 12 maximum usable queues
  Current address : 08:b2:58:1f:0d:0d, Hardware address: 08:b2:58:1f:0d:0d
  Last flapped    : 2018-11-02 08:43:20 UTC (5d 23:12 ago)
  Input rate      : 736 bps (1 pps)
  Output rate     : 736 bps (1 pps)
  Active alarms   : None
  Active defects  : None
  PCS statistics
    Bit errors          Seconds
    Errored blocks      0
  Ethernet FEC statistics
    FEC Corrected Errors      0
    FEC Uncorrected Errors    0
    FEC Corrected Errors Rate 0
    FEC Uncorrected Errors Rate 0
  PRBS Statistics : Disabled
  Interface transmit statistics: Disabled

Logical interface ge-0/0/11.0 (Index 336) (SNMP ifIndex 535)
  Flags: Up SNMP-Traps 0x24024000 Encapsulation: Ethernet-Bridge
  Input packets : 0
  Output packets: 0
  Protocol eth-switch, MTU: 1514
  Flags: Trunk-Mode
```

Related
Documentation

VDSL2 Interfaces on NFX250 Devices

- [VDSL Interface Overview on page 25](#)
- [VDSL2 Network Deployment Topology on page 25](#)

- [VDSL2 Interface Support on NFX Series Devices on page 27](#)
- [Example: Configuring VDSL SFP Interface on NFX250 Devices on page 28](#)

VDSL Interface Overview

Very-high-bit-rate digital subscriber line (VDSL) technology is part of the xDSL family of modem technologies that provide faster data transmission over a single flat untwisted or twisted pair of copper wires. The VDSL lines connect service provider networks and customer sites to provide high bandwidth applications (triple-play services) such as high-speed Internet access, telephone services like VoIP, high-definition TV (HDTV), and interactive gaming services over a single connection.

VDSL2 is an enhancement to G.993.1 (VDSL) and permits the transmission of asymmetric (half-duplex) and symmetric (full-duplex) aggregate data rates up to 100 Mbps on short copper loops using a bandwidth up to 17 MHz. The VDSL2 technology is based on the ITU-T G.993.2 (VDSL2) standard, which is the International Telecommunication Union standard describing a data transmission method for VDSL2 transceivers.

The VDSL2 uses discrete multitone (DMT) modulation. DMT is a method of separating a digital subscriber line signal so that the usable frequency range is separated into 256 frequency bands (or channels) of 4.3125 KHz each. The DMT uses the Fast Fourier Transform (FFT) algorithm for demodulation or modulation for increased speed.

VDSL2 interface supports Packet Transfer Mode (PTM). The PTM mode transports packets (IP, PPP, Ethernet, MPLS, and so on) over DSL links as an alternative to using Asynchronous Transfer Mode (ATM). PTM is based on the Ethernet in the First Mile (EFM) IEEE802.3ah standard.

VDSL2 provides backward compatibility with ADSL2 and ADSL2+ because this technology is based on both the VDSL1-DMT and ADSL2/ADSL2+ recommendations.

- [VDSL2 Vectoring Overview on page 25](#)

VDSL2 Vectoring Overview

Vectoring is a transmission method that employs the coordination of line signals that reduce crosstalk levels and improve performance. It is based on the concept of noise cancellation, like noise-cancelling headphones. The ITU-T G.993.5 standard, "Self-FEXT Cancellation (Vectoring) for Use with VDSL2 Transceivers," also known as G.vector, describes vectoring for VDSL2.

The scope of Recommendation ITU-T G.993.5 is specifically limited to the self-FEXT (far-end crosstalk) cancellation in the downstream and upstream directions. The FEXT generated by a group of near-end transceivers and interfering with the far-end transceivers of that same group is canceled. This cancellation takes place between VDSL2 transceivers, not necessarily of the same profile.

VDSL2 Network Deployment Topology

In standard telephone cables of copper wires, voice signals use only a fraction of the available bandwidth. Like any other DSL technology, the VDSL2 technology utilizes the

remaining capacity to carry the data and multimedia on the wire without interrupting the line's ability to carry voice signals.

This example depicts the typical VDSL2 network topology deployed using NFX device.

A VDSL2 link between network devices is set up as follows:

1. Connect an end-user device such as a LAN, hub, or PC through an Ethernet interface to the customer premises equipment (CPE) (for example, an NFX device).
2. Connect the CPE to a DSLAM.
3. The VDSL2 interface uses either Gigabit Ethernet or fiber as second mile to connect to the Broadband Remote Access Server (B-RAS) as shown in [Figure 3 on page 26](#).
4. The ADSL interface uses either Gigabit Ethernet (in case of IP DSLAM) as the “second mile” to connect to the B-RAS or OC3/DS3 ATM as the second mile to connect the B-RAS as shown in [Figure 4 on page 26](#).

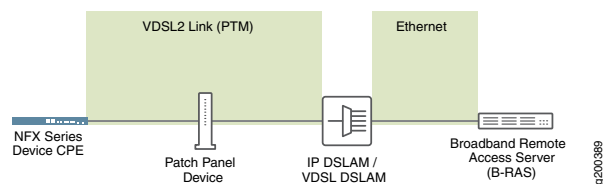


NOTE: The VDSL2 technology is backward compatible with ADSL2 and ADSL2+. VDSL2 provides an ADSL2 and ADSL2+ interface in an ATM DSLAM topology and provides a VDSL2 interface in an IP or VDSL DSLAM topology.

The DSLAM accepts connections from many customers and aggregates them to a single, high-capacity connection to the Internet.

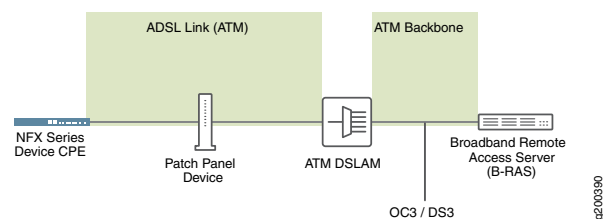
[Figure 3 on page 26](#) shows a typical VDSL2 network topology.

Figure 3: Typical VDSL2 End-to-End Connectivity and Topology Diagram



[Figure 4 on page 26](#) shows a backward-compatible ADSL topology using ATM DSLAM.

Figure 4: Backward-Compatible ADSL Topology (ATM DSLAM)



VDSL2 Interface Support on NFX Series Devices

The VDSL2 interface is supported on the NFX Series devices listed in [Table 4 on page 27](#). (Platform support depends on the Junos OS release in your installation.)

Table 4: VDSL2 Annex A and Annex B Features

Features	POTS
Devices	CPE-SFP-VDSL2
Supported annex operating modes	Annex A and Annex B*
Supported Bandplans	Annex A 998 Annex B 997 and 998
Supported standards	ITU-T G.993.2 and ITU-T G.993.5 (VDSL2)
Used in	North American network implementations
ADSL backward compatibility	G 992.3 (ADSL2) G 992.5 (ADSL2+)



NOTE: Only one CPE-SFP-VDSL2 device is supported at a time.

VDSL2 Interface Compatibility with ADSL Interfaces

VDSL2 interfaces on NFX Series devices are backward compatible with most ADSL2 and ADSL2+ interface standards. The VDSL2 interface uses Ethernet in the First Mile (EFM) mode or Packet Transfer Mode (PTM) and uses the named interface ge-0/0/10 and ge-0/0/11.



NOTE:

- The VDSL2 interface has backward compatibility with ADSL2 and ADSL2+.
- It requires around 60 seconds to switch from VDSL2 to ADSL2 and ADSL2+ or from ADSL2 and ADSL2+ to VDSL2 operating modes.

VDSL2 Interfaces Supported Profiles

A profile is a table that contains a list of pre-configured VDSL2 settings. [Table 5 on page 28](#) lists the different profiles supported on the VDSL2 interfaces and their properties.

Table 5: Supported Profiles on the VDSL2 Interfaces

Profiles	Data Rate
8a	50
8b	50
8c	50
8d	50
12a	68
12b	68
17a	100
Auto	Negotiated (based on operating mode)

Example: Configuring VDSL SFP Interface on NFX250 Devices

- [Requirements on page 28](#)
- [Overview on page 28](#)
- [Configuration on page 29](#)
- [Results on page 31](#)

Requirements

This example uses the following hardware and software components:

- NFX250 device running Junos OS Release 15.1X53-D495.

Overview

In this example, you are configuring VDSL SFP interface on an NFX250 device with the following configurations:

- Physical interface - **ge-0/0/11**
- Virtual network function (VNF) - **nfx250-a-vsrx1**
- Memory size - **4194304**
- VDSL SFP options - **profile auto and carrier auto**

To configure VDSL SFP interface on NFX250 devices, you must configure JDM, vSRX, and vJunos0.



NOTE: Ensure that connectivity to the host is not lost during the configuration process.

Configuration

Step-by-Step Procedure

To configure VDSL SFP interfaces on NFX250 devices:

1. Connect to the host.

```
user@host> configure
[edit]
user@host#
```

2. Create VLANs using VLAN IDs:

```
user@host# set host-os vlans xdsl-test vlan-id 50
user@host# set host-os vlans vlan130 vlan-id 130
user@host# set host-os vlans vlan131 vlan-id 131
user@host# set host-os vlans vlan132 vlan-id 132
```

3. Enable enhanced orchestration to manage VNFs and service chains without requiring the VNF XML descriptor files:

```
user@host# set system services enhanced-orchestration
```

4. Allocate resources for a VNF:

```
user@host# set virtual-network-functions nfx250-a-vsrx1 image
/var/third-party/images/media-vsrx-vm disk-15.1-2018-04-24.0_DEV_X_151_X49.qcow2
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu 0 physical-cpu 2
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu 1 physical-cpu 6
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu count 2
user@host# set virtual-network-functions nfx250-a-vsrx1 virtual-cpu features
hardware-virtualization
user@host# set virtual-network-functions nfx250-a-vsrx1 no-default-interfaces
user@host# set virtual-network-functions nfx250-a-vsrx1 memory size 4194304
user@host# set virtual-network-functions nfx250-a-vsrx1 memory features hugepages
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth0 management
out-of-band
```

5. Map physical interfaces to virtual interfaces:

```
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth1 mapping vlan
mode trunk
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth1 mapping vlan
members vlan130
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
mode trunk
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members vlan130
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members vlan131
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members vlan132
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
members xdsl-test
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth2 mapping vlan
native-vlan-id 50
```

```
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth3 mapping vlan
mode trunk
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth3 mapping vlan
members xdsl-test
user@host# set virtual-network-functions nfx250-a-vsrx1 interfaces eth3 mapping vlan
native-vlan-id 50
```

6. Configure the Junos Control Plane (JCP) virtual machine (VM):

```
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching interface-mode
trunk
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
xdsl-test
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
vlan130
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
vlan131
user@host# set interfaces sxe-0/0/0 unit 0 family ethernet-switching vlan members
vlan132
user@host# set interfaces ge-0/0/11 native-vlan-id 50
user@host# set interfaces ge-0/0/11 dsl-sfp-options vdsl-options profile auto
user@host# set interfaces ge-0/0/11 dsl-sfp-options vdsl-options carrier auto
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members
xdsl-test
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members vlan130
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members vlan131
user@host# set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members vlan132
user@host# set vlans vlan130 vlan-id 130
user@host# set vlans vlan131 vlan-id 131
user@host# set vlans vlan132 vlan-id 132
user@host# set vlans xdsl-test vlan-id 50
```

7. Commit the configuration.

```
user@host# commit and-quit
user@host> exit
```

Results

From configuration mode, verify your configuration by entering the **show interfaces ge-0/0/11** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it:

```
[edit]
user@host# show interfaces ge-0/0/11

Physical interface: ge-0/0/11, Enabled, Physical link is Up
  Interface index: 258, SNMP ifIndex: 533
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, VDSL2 mode, Speed: 1000mbps,
  BPDU Error: None, Loop Detect PDU Error: None,
  Ethernet-Switching Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled,
  Auto-negotiation: Enabled, Remote fault: Online, IEEE 802.3az Energy Efficient
  Ethernet: Disabled, Auto-MDIX: Enabled
  VDSL status:
    Modem status : Showtime (Profile-12a)
    VDSL profile  :      Auto      Annex B
    Device flags  : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 12 supported, 12 maximum usable queues
  Current address: 08:b2:58:1f:0d:0d, Hardware address: 08:b2:58:1f:0d:0d
  Last flapped   : 2018-11-02 08:43:20 UTC (6d 00:29 ago)
  Input rate     : 888 bps (1 pps)
  Output rate    : 888 bps (1 pps)
  Active alarms  : None
  Active defects : None
  PCS statistics
    Bit errors           Seconds 0
    Errored blocks       Seconds 0
  Ethernet FEC statistics
    FEC Corrected Errors      Errors 0
    FEC Uncorrected Errors    Errors 0
    FEC Corrected Errors Rate Errors 0
    FEC Uncorrected Errors Rate Errors 0
    PRBS Statistics : Disabled
  Interface transmit statistics: Disabled

Logical interface ge-0/0/11.0 (Index 336) (SNMP ifIndex 535)
  Flags: Up SNMP-Traps 0x24024000 Encapsulation: Ethernet-Bridge
  Input packets : 0
  Output packets: 0
  Protocol eth-switch, MTU: 1514
  Flags: Trunk-Mode
```

[Related
Documentation](#)

