Juniper Platform Automation and NetDevOps

COURSE OVERVIEW

This five-day, intermediate-level course expands foundational automation skills into large-scale, policy-driven, and life cycle–aware automation for Junos-based networks. Students learn how to design and operate automation workflows that enhance speed, consistency, and reliability while enabling organizations to grow network capacity without increasing operational overhead. Through a combination of guided instruction, demonstrations, and extensive hands-on labs, participants gain the capabilities needed to automate complex, multi-device environments with strong validation, observability, and change-control processes.

Ansible and Jinja2 help learners deploy and adjust OSPF, BGP, and MPLS settings on many Junos devices, keeping configurations correct. The course teaches how to automate Junos software upgrades using pre-checks, post-checks, and scripted rollback workflows to reduce outage risks and maintenance time. JSNAPy-based validation is applied to detect configuration drift and automatically revert failed or noncompliant changes, strengthening operational governance and network stability.

Students implement event-driven automation using Junos commit, op, event, and SNMP scripts to enforce policies in real time. They execute RPCs and advanced operational scripts to automate verification tasks, permissions management, and platform-level controls. Model-driven automation is introduced using YANG and OpenConfig to support vendor-neutral programmability across Junos and non-Junos devices, enabling standardized, future-ready automation architectures.

The course also covers streaming telemetry with gNMI and visualization of network health using TIG stack components for proactive monitoring and capacity planning. Students apply Infrastructure as Code (IaC) principles using Terraform and the Juniper JTAF provider to manage network configurations declaratively and support repeatable, version-controlled operations.

By the end of the course, learners can design and implement end-to-end automation workflows that reduce meantime to detect and resolve issues, enforce consistency across distributed network environments, and support scalable network growth through robust automation practices.

This course uses Junos OS Release 24.4R1, Ansible 2.18, SLAX version 1.2, and JSNAPy 1.3.8.

COURSE LEVEL

Juniper Platform Automation and NetDevOps is an intermediate-level course.

AUDIENCE

Individuals responsible for configuring and monitoring devices running the Junos OS

PREREQUISITES

- Intermediate-level networking knowledge
- Understanding of the OSI model and the TCP/IP protocol suite
- Completion of the *Introduction to Juniper Platform Automation and NetDevOps* course or equivalent experience

RELATED JUNIPER PRODUCTS

SRX Series Firewalls, QFX Series Switches, EX Series Switches, PTX Series Routers, ACX Series Routers, MX Series Universal Routing Platforms

RECOMMENDED NEXT COURSE

Advanced Junos Platform Automation and DevOps

OBJECTIVES

- Describe Junos automation architecture and tools.
- Describe how to use Docker for NetDevOps.
- Describe Junos Evolved automation.
- Describe Jinja2 templates for Junos automation.
- Perform network testing with JSNAPy.

© 2026 Hewlett Packard Enterprise Development LP. Course content subject to change.

- Describe Ansible automation.
- Use Ansible to configure Junos devices.
- Configure Ansible with Jinja2 templates and JSNAPy.
- Describe how to develop Junos op scripts using Python.
- Describe how to develop Junos commit scripts using Python.
- Describe Junos event policies.
- Describe how to develop Junos event scripts using Python.
- Describe the basics of the SLAX language.
- Describe extension functions in Junos OS automation scripts.
- Describe Junos op scripts using SLAX.
- Describe Junos commit scripts and event scripts using SLAX.
- Describe the syntax and semantics of the YANG language.
- Use OpenConfig with Junos OS.
- Create custom Junos YANG modules.
- Describe how Terraform can be used to automate Junos.
- Describe the syntax and semantics of protocol buffers and gRPC.
- Describe the syntax and semantics of the gNMI service.
- Configure Junos gNMI (dial-in) telemetry using TIG stack.
- Perform basic Junos OS Evolved automation tasks.
- Develop Junos SNMP scripts.
- Describe JET functionality and how to use JET APIs to automate Junos.

COURSE CONTENTS

DAY 1

Module 01: Junos Automation Fundamentals

- List the benefits of Junos automation and NetDevOps
- Explain the Junos automation stack and Junos APIs
- Describe the difference between on-box automation and off-box automation
- Describe the purpose of op, commit, event, and SNMP scripts
- Describe the various tools, languages, and libraries used to automate Junos devices

Module 02: Using Docker for NetDevOps

- Explain the basic concepts of Docker
- Explain how to create and manage containers
- Configure networks for Docker containers

Lab 01: Using Docker for NetDevOps

Module 03: Jinja2 Fundamentals

- Explain how to use Jinja2 templates for Junos automation
- Explain how to create Jinja2 templates

Lab 02: Creating Jinja2 Templates

Module 04: Automating Junos Using JSNAPy

- Describe how JSNAPy can automate Junos OS
- Explain how to install and configure JSNAPy
- Create snapshots and perform tests using JSNAPy
- Explain the integration of JSNAPy with other automation tools

© 2026 Hewlett Packard Enterprise Development LP. Course content subject to change.

Lab 03: JSNAPy

DAY 2

Module 05: Introduction to Ansible

- Explain the fundamentals of Ansible
- Describe the elements in the Ansible environment for network device management
- Explain how to retrieve information from Junos devices using Ansible

Lab 04: Ansible Operations with Junos

Module 06: Automating Junos Using Ansible

- Configure Ansible vault to securely connect to Junos devices
- Use Ansible to manage Junos device configurations
- Optimize Ansible playbook operations with extended features

Lab 05: Using Ansible for Junos Configuration Management

Module 07: Ansible with Jinja2 Templates and JSNAPy

- Create Ansible playbooks that use variables and templates
- Configure Ansible and Jinja2 templates to manage Junos device configurations
- Configure Ansible to generate JSNAPy snapshots

Lab 06: Using Ansible with Jinja2 Templates and JSNAPy

Module 08: Junos Op Scripts Using Python

- Explain how to create and execute Python op scripts
- Describe how to use arguments with Python op scripts
- Explain how to execute Junos RPCs from Python op scripts
- Describe how to configure Junos devices with Python op scripts

Lab 07: Junos Op Scripts Using Python

Module 09: Junos Commit Scripts Using Python

- Describe how Python commit scripts can perform different actions
- Develop Python commit scripts that perform configuration changes
- Configure and enable Python commit scripts

Lab 08: Junos Commit Scripts Using Python

DAY 3

Module 10: Junos Event Policies

- Identify Junos OS events
- Describe how to create Junos OS event policies

Module 11: Junos Event Scripts Using Python

- Describe how to create Junos event scripts

Lab 09: Junos Event Policies and Event Scripts

Module 12: SLAX Fundamentals

- Describe the relation of SLAX to XML, XPath, and Junos XML API
- Create simple SLAX scripts
- Describe SLAX templates, variables, and flow control statements

© 2026 Hewlett Packard Enterprise Development LP. Course content subject to change.

Module 13: SLAX Extension Functions

- Explain how to use the Junos function library
- Describe the difference between the available SLAX versions

Module 14: Junos Op Scripts Using SLAX

- Describe how to create and execute SLAX op scripts
- Explain how to use arguments with SLAX op scripts
- Describe how to execute Junos RPCs from SLAX op scripts
- Explain how to configure Junos devices with SLAX op scripts

Lab 10: Junos Op Scripts Using SLAX

DAY 4

Module 15: Junos Commit Scripts and Event Scripts Using SLAX

- Describe the basic SLAX commit script operation
- Develop SLAX commit scripts that perform transient and persistent changes
- Configure and enable SLAX commit scripts
- Describe how to create event scripts using SLAX

Lab 11: Junos Commit Scripts Using SLAX

Module 16: YANG Language

- List the main features and the language syntax of YANG
- Identify YANG language use cases with Junos OS

Lab 12: Using pyang with Juniper YANG Models

Module 17: OpenConfig

- Describe the advantages of OpenConfig
- Modify the Junos OS configuration using OpenConfig

Lab 13: Implementing OpenConfig

Module 18: Custom YANG Modules

- Describe custom configuration statements and translation scripts
- Describe custom RPCs and action scripts

Lab 14: Implementing a Custom YANG Module

Module 19: Terraform (Experimental)

- Describe the Infrastructure as Code workflow with Terraform
- Create a Junos Terraform provider using Junos Terraform Automation Framework
- Automate Junos configuration with custom Junos Terraform provider

Lab 15: Using Junos Terraform Provider

DAY 5

Module 20: Protocol Buffers and gRPC

- List the main features and the workflow of protocol buffers
- Describe gRPC and implement a simple service

Lab 16: Protocol Buffers and gRPC

© 2026 Hewlett Packard Enterprise Development LP. Course content subject to change.

Module 21: gNMI

- Describe the RPCs and main messages of the gNMI service
- Use the gNMIc client with the Junos target

Lab 17: Using gNMIc with Junos Targets

Module 22: Junos gNMI (Dial-In) Telemetry

- Describe how to collect telemetry from Junos using gNMI and TIG stack
- Describe how to display telemetry data in Grafana

Lab 18: Junos gNMI Telemetry with TIG Stack

SELF-STUDY MODULES

Module 23: Junos OS Evolved Automation

- Describe Junos OS Evolved
- Explain how to install and manage vJunosEvolved
- Describe how to run the third-party applications in the containers
- Describe how to automate Junos OS Evolved using Junos PyEZ

Module 24: Junos SNMP Scripts

- Describe how Junos OS SNMP scripts are used
- Create and configure Junos OS SNMP scripts

Lab 19: Junos SNMP Scripts (Optional)

Module 25: Juniper Extension Toolkit

- List the main JET components and use cases
- Describe gRPC and JET IDL files
- Develop JET applications using Service API

Lab 20: Using JET Service API (Optional)

© 2026 Hewlett Packard Enterprise Development LP. Course content subject to change.