




Dell Networking Open Automation Guide

July 2015

Regulatory Model: Open Automation



Notes, cautions, and warnings

-  **NOTE:** A NOTE indicates important information that helps you make better use of your computer.
-  **CAUTION:** A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.
-  **WARNING:** A WARNING indicates a potential for property damage, personal injury, or death.

Copyright © 2015 Dell Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. Dell™ and the Dell logo are trademarks of Dell Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

Contents

| | |
|---|-----------|
| 1 About this Guide..... | 9 |
| Audience..... | 9 |
| Supported Platforms and Required Dell Networking OS Versions..... | 9 |
| Conventions..... | 10 |
| Related Documents..... | 10 |
| 2 Open Automation Framework..... | 12 |
| Bare Metal Provisioning..... | 13 |
| Smart Scripting..... | 13 |
| REST API..... | 14 |
| Web Server with HTTP Support..... | 14 |
| 3 Bare Metal Provisioning..... | 15 |
| Introduction..... | 15 |
| How it Works..... | 15 |
| Prerequisites..... | 15 |
| Industrial Standard Upgrades with BMP..... | 16 |
| BMP Process Overview..... | 16 |
| BMP Operations..... | 17 |
| Configure BMP..... | 17 |
| BMP Mode..... | 18 |
| Normal Context..... | 19 |
| Normal Mode..... | 19 |
| BMP Commands and Examples..... | 19 |
| System Boot and Set-Up Behavior in BMP Mode..... | 19 |
| BMP Mode: Boot and Set-UP Behavior..... | 22 |
| Reload Without a DHCP Server Offer..... | 22 |
| Reload with a DHCP Server Offer Without a Dell Networking OS Offer..... | 22 |
| Reload with a DHCP Server Offer and no Configuration File..... | 23 |
| Reload with a DHCP Server Offer Without a DNS Server..... | 24 |
| DHCP Offer Vendor-Specific Option for BMP..... | 25 |
| DHCP Offer Relay Option 82..... | 25 |
| Software Upgrade Using BMP..... | 26 |
| Apply Configurations Using BMP Scripts..... | 26 |
| Preconfiguration Scripts..... | 26 |
| Auto-Execution Scripts..... | 27 |
| Post-Configuration Scripts..... | 27 |
| Using Preconfiguration Scripts..... | 27 |

| | |
|---|-----------|
| Using the Post-Configuration Script..... | 28 |
| Using Auto-Execution Script (Normal Mode Only)..... | 28 |
| Timers in Pre-configuration and Post-configuration Scripts..... | 29 |
| Script Examples..... | 29 |
| BMP Operations on Servers Overview..... | 34 |
| Configuring the DHCP Server Settings..... | 34 |
| DHCP Server IP Blacklist..... | 35 |
| MAC-Based Configuration..... | 35 |
| MAC-Based IP Address Assignment..... | 36 |
| Class-Based Configuration..... | 37 |
| File Server Settings..... | 37 |
| Domain Name Server Settings..... | 38 |
| BMP MIB Support..... | 38 |
| 4 Bare Metal Provisioning CLI..... | 39 |
| auto-save..... | 39 |
| bmp logging..... | 40 |
| boot-type..... | 41 |
| config-scr-download..... | 42 |
| dhcp-timeout..... | 43 |
| relay..... | 43 |
| reload conditional nvram-cfg-change..... | 44 |
| reload-type..... | 45 |
| retry-count..... | 46 |
| script post-config..... | 47 |
| show boot bmp..... | 48 |
| show config..... | 49 |
| show reload-type..... | 50 |
| stop bmp..... | 51 |
| Vendor class identifier..... | 52 |
| verify..... | 53 |
| 5 Open Management Infrastructure..... | 55 |
| Overview..... | 55 |
| Installing the OMI Agent Package..... | 55 |
| Prerequisites..... | 56 |
| Configuring the OMI..... | 56 |
| Other Scripts..... | 56 |
| Getting Started with OMI..... | 57 |
| Examples..... | 58 |
| 6 Puppet..... | 65 |

| | |
|--|-----------|
| Overview..... | 65 |
| Installing Puppet..... | 65 |
| Prerequisites..... | 65 |
| Configuring the Puppet Agent..... | 66 |
| Puppet Agent Configuration Scripts..... | 67 |
| puppet-config..... | 67 |
| puppet-show..... | 67 |
| puppet-agent..... | 67 |
| puppet-apply..... | 68 |
| Building the Manifest File..... | 68 |
| Hostname Configuration..... | 68 |
| Layer 2 Interface Configuration..... | 68 |
| LAG Configuration..... | 69 |
| Physical Interface Configuration..... | 69 |
| VLAN Configuration..... | 69 |
| Using the Puppet Factor..... | 70 |
| Setting up the Puppet Master..... | 70 |
| 7 Smart Scripting..... | 71 |
| Overview..... | 71 |
| Download the Smart Scripting Package..... | 72 |
| Installing Smart Scripting..... | 73 |
| Displaying Installed Packages..... | 74 |
| Uninstalling Smart Scripting..... | 74 |
| Dell Proprietary Utilities..... | 74 |
| clido..... | 74 |
| dellsh..... | 75 |
| dellsyslog..... | 75 |
| Dell SmartScript Utilities..... | 76 |
| SQLite..... | 77 |
| NET SNMP Client..... | 78 |
| Limits on System Usage..... | 78 |
| Supported UNIX Utilities..... | 79 |
| Creating a User Name and Password for Smart Scripting..... | 81 |
| Logging in to a NetBSD UNIX Shell..... | 81 |
| Downloading Scripts to a Switch..... | 81 |
| Setting a Search Path for Scripts..... | 82 |
| Scheduling and Executing Scripts..... | 82 |
| Script Scheduler from EXEC mode..... | 82 |
| Managing Executed Scripts..... | 84 |
| Viewing Script Information..... | 85 |
| Running a Script from the UNIX Shell..... | 86 |

| | |
|--|------------|
| Running Scripts with User Privileges..... | 87 |
| 8 Smart Scripting CLI..... | 88 |
| disable..... | 88 |
| execute..... | 88 |
| mount nfs..... | 89 |
| package install..... | 90 |
| package uninstall..... | 91 |
| script (run)..... | 92 |
| script (stop/resume/clear/kill/unschedule)..... | 94 |
| script event-handler..... | 95 |
| script execute (EXEC mode)..... | 95 |
| script execute (CONFIGURATION mode)..... | 97 |
| script get..... | 99 |
| script path..... | 100 |
| script remove..... | 101 |
| script trigger-event..... | 101 |
| show packages..... | 104 |
| show script..... | 106 |
| start shell..... | 107 |
| triggered-by..... | 108 |
| switch script limit..... | 109 |
| username (event handler)..... | 110 |
| username..... | 111 |
| 9 REST API..... | 112 |
| HTTP and HTTPS..... | 112 |
| XML..... | 112 |
| Important Points to Remember..... | 112 |
| REST Authentication..... | 113 |
| Request Query Parameters..... | 113 |
| Sample BGP Configurations..... | 113 |
| HTTP Status Error Codes..... | 118 |
| REST API – Protocol Data Unit (PDU) Structure..... | 119 |
| Configurations..... | 120 |
| BGP..... | 120 |
| Global Configuration..... | 126 |
| Interface..... | 127 |
| IPv6 Global Configurations..... | 130 |
| Management-Route..... | 131 |
| Policy Based Routing (PBR)..... | 131 |
| Port Mirroring..... | 133 |

| | |
|---|------------|
| Route Maps..... | 133 |
| Static Route..... | 134 |
| Weighted ECMP..... | 134 |
| Operational..... | 134 |
| BGP..... | 134 |
| BGPv6..... | 136 |
| Interface..... | 142 |
| IP Route..... | 143 |
| IP Interface..... | 144 |
| Management – Route..... | 145 |
| MAC Address Table..... | 145 |
| Policy Based Routing (PBR)..... | 146 |
| Port Mirroring..... | 147 |
| Static Access-list..... | 148 |
| System..... | 149 |
| System Alarm..... | 150 |
| System Inventory..... | 150 |
| System Version..... | 151 |
| VLAN..... | 151 |
| Operational Data for IPv6..... | 152 |
| Interface..... | 152 |
| Management-Route..... | 153 |
| Prefix-list..... | 154 |
| Route..... | 154 |
| Management Information Base (MIB)..... | 155 |
| BGP V2..... | 155 |
| Forwarding Information Base (FIB)..... | 159 |
| Forwarding Plane Statistics..... | 160 |
| IF MIB Operations..... | 161 |
| IETF Interface Operations..... | 163 |
| REST API Framework to Execute the CLIs..... | 163 |
| Samples of the config-command..... | 164 |
| Samples of the EXEC command..... | 164 |
| Samples of the show-command..... | 165 |
| 10 REST API CLI..... | 166 |
| rest-server..... | 166 |
| crypto cert generate..... | 167 |
| crypto cert install rest-server..... | 168 |
| show crypto cert..... | 169 |
| show ip rest-server..... | 169 |

| | |
|---|------------|
| 11 Web Server with HTTP Support..... | 171 |
| Starting the Web Server..... | 171 |

About this Guide


This document describes the components and uses of the Open Automation Framework designed to run on the Dell Networking operating system (OS).

Open Automation includes support for:

- Bare metal provisioning (BMP)
- Open management infrastructure (OMI)
- Puppet
- Representational state transfer (REST) application programming interface (API)
- Smart scripting
- Web graphic user interface (GUI) and HTTP server

Audience

This document is intended for data center managers and network administrators responsible for virtualization and system management. You must have basic knowledge about virtualization technology and networking to work with the Open Automation framework.

 **NOTE:** Although this document contains information about protocols, it is not intended to provide complete information about protocol configuration and usage. For this information, see the documents listed in [Related Documents](#).

Supported Platforms and Required Dell Networking OS Versions

Dell Networking switches supports Open Automation on the following OS versions.

| Dell Networking Platform | Minimum Operating Software |
|--------------------------|----------------------------|
| S4810 | 8.3.10.1 or later |
| S4820T | 9.2(0.0) or later |
| S3048-ON | 9.8(0.0P2) or later |
| S5000 | 9.6(0.0) or later |
| S6000 | 9.0.2.0 or later |
| S6000-ON | 9.7(0.0) or later |
| MXL | 9.2(0.0) or later |
| Z9500 | 9.5(0.1) or later |

Dell Networking Platform Minimum Operating Software

(BMP, SmartScripts, and REST API only)

Conventions

To describe command syntax, this document uses the following conventions.

| Convention | Description |
|------------------|--|
| keyword | Keywords are shown in a terminal font and are entered in the CLI as listed. |
| parameter | Parameters are shown in italics and require you enter a number or word in the CLI. |
| {X} | Keywords and parameters within braces require entries in the CLI. |
| [X] | Keywords and parameters within brackets are optional. |
| x y | Keywords and parameters separated by bar require that you to choose one. |

Related Documents

For more information about Dell Networking switches described in this document, refer to the following documents.

| Dell Networking Platform | Related Documents |
|--------------------------|--|
| S4810 | <ul style="list-style-type: none">• <i>Dell Networking OS Command Line Reference guide for the S4810 System</i>• <i>Dell Networking OS Configuration Guide for the S4810 System</i>• <i>Installing the S4810 System</i> |
| S4820T | <ul style="list-style-type: none">• <i>Dell Networking OS Command Line Reference guide for the S4820T System</i>• <i>Dell Networking OS Configuration Guide for the S4820T System</i>• <i>Installing the S4820T System</i> |
| S3048-ON | <ul style="list-style-type: none">• <i>Dell Networking OS Command Line Reference guide for the S3048-ON System</i>• <i>Dell Networking OS Configuration Guide for the S3048-ON System</i>• <i>Installing the S3048-ON System</i> |
| S5000 | <ul style="list-style-type: none">• <i>Dell Networking OS Command Line Reference guide for the S5000 System</i>• <i>Dell Networking OS Configuration Guide for the S5000 System</i>• <i>Installing the S5000 System</i> |
| S6000 | <ul style="list-style-type: none">• <i>Dell Networking OS Command Line Reference guide for the S6000 System</i>• <i>Dell Networking OS Configuration Guide for the S6000 System</i> |

Dell Networking Platform

Related Documents

- *Installing the S6000 System*

S6000-ON

- *Dell Networking OS Command Line Reference guide for the S6000-ON System*
- *Dell Networking OS Configuration Guide for the S6000-ON System*
- *Installing the S6000-ON System*

MXL Switch

- *Dell Networking OS Command Line Reference guide for the MXL 10/40GbE Switch I/O Module*
- *Dell Networking OS Configuration Guide for the MXL 10/40GbE Switch I/O Module*

Z9500

- *Dell Networking OS Command Line Reference guide for the Z9500 System*
- *Dell Networking OS Configuration Guide for the Z9500 System*
- *Installing the Z9500 System*

Dell Networking platform and OS

Dell Networking Release Notes for the platform and version you are using.

Open Automation Framework

The Dell Networking S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL platforms support the Open Automation framework.

The Open Automation framework provides an open, industry standards-based automation technology that simplifies managing dynamic virtual data centers and reduces risk and overhead.

With the Open Automation framework, resources in a virtualized data center are efficiently managed without requiring manual reconfiguration of virtual switches (vSwitches), virtual machines (VMs) on network servers, and VM control software each time there is a network change. Automated provisioning of network resources during VM migration also maintains connectivity and security policies.

To automate monitoring and network device management, the Open Automation framework uses industry standard scripting languages, such as Perl, Python, and Ruby. To adapt to required configuration changes, you can quickly allocate virtual resources. In addition, a network device failure is more quickly detected and resolved; as a result, network uptime increases.

Automated bare metal provisioning (BMP) allows you to reduce operational overhead by automatically configuring Dell Networking switches, accelerating switch installation, and simplifying operating system upgrades.

Support for multiple, industry standard hypervisors, virtual switches, and system management tools ensure that automated solutions work within an established data-center environment in which heterogeneous server, storage, and networking equipment inter-operate. In addition, Open Automation allows you to customize automated solutions for your current multi-vendor virtualization environment.

An onboard web-based graphical user interface (GUI) provides a user-friendly way to monitor and manage a data center network. To provide additional management capability, HTTP and HTTPS daemons run on supported switches, such as the representational state transfer (REST) application programming interface (API).

The Open Automation framework consists of the following network management tools:

- Bare metal provisioning
- Open management interface
- Puppet
- Smart scripting
- Representational state transfer application programming interface
- Web server with HTTP support

To extend and add functionality to your Dell Networking OS without requiring updates to the OS release, use these components together or independently.

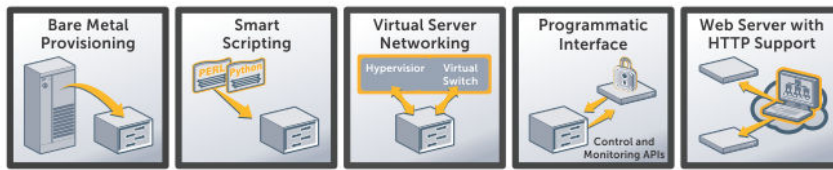



Figure 1. Open Automation Framework

 **NOTE:** The Open Automation framework is referred to as Open Automation in the rest of this document.

Bare Metal Provisioning

Bare metal provisioning (BMP) provides the following features.

- Automatic network switch configuration and automated configuration updates
- Enforced standard configurations
- Reduced installation time
- Simplified operating system upgrades

Automated BMP reduces operational expenses, accelerates switch installation, simplifies upgrades, and increases network availability by automatically configuring Dell Networking switches. BMP eliminates the need for a network administrator to manually configure a switch, resulting in faster installation, elimination of configuration errors, and enforcement of standard configurations.

With BMP, after you install your switch, the switch searches the network for a dynamic host configuration protocol (DHCP) server. The DHCP server provides the switch with an IP address and the location of a file server, such as TFTP. The file server maintains a configuration file and an approved version of the Dell Networking OS. The switch automatically configures itself by loading and installing an embedded Dell Networking OS image with the startup configuration file.

For more information about BMP, refer to the *Dell Networking OS Configuration Guide* for your platform (S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, or Z9500 switches) or to the *Dell Networking OS Configuration Guide for the MXL 10/40GbE Switch IO Module* for MXL switches.

Smart Scripting

Smart Scripting provides the following:

- Support for industry standard languages, such as Perl, Python, and Ruby, avoiding the need to learn a new proprietary scripting language.
- The ability to customize device monitoring and management to suit your network needs, including custom maintenance tasks, discovery programs, and event logging for faster problem resolution.

Smart scripting increases network availability and manageability by allowing network administrators to deploy custom monitoring and management scripts on Dell Networking switches. Using custom scripts, network administrators can implement version control systems, automatically generate alerts, create custom logging tools, and automate management of network devices. Any function that you can perform through the Dell Networking OS command line interface (CLI), you can perform with smart scripting.

The scripting environment (Expect, Perl, Python, Ruby, Tcl, UNIX and ZSH shell scripts) makes it easy for IT administrators to quickly develop scripts.

REST API

Representational state transfer (REST)-application programming interface (API) provides tools that allow you to manage Dell Networking OS switches by in-house or third-party system management tools. The program management environment and set of interfaces communicate directly with the third-party system management tools, thus avoiding the need for a dedicated network management tool. REST API simplifies network management and reduces the number of management tools you need.

- Common third-party management tool sets are supported as plug-ins to Open Automation, including Dell AIM, EMC Smarts Ionix, IBM Systems Director, HP Network Automation (NA), CA Spectrum Infrastructure Manager, and Oracle Enterprise Manager (OEM).
- Industry standard management protocols are supported, such as simple network management protocol (SNMP) (Get and Set) and REST.
- User protocols are supported, such as CLI/CLI-script, XML (Get and Set), and web-based commands.

REST-API greatly improves the network manageability by allowing the Dell Networking switches to be managed by third party system management tools via standard program interfaces.

The programmatic management environment and set of interfaces communicate directly with the third-party system management tools, avoiding the need for a dedicated network management tool. As a result, the network management is simplified and the number of management tools are minimized.

Web Server with HTTP Support

Open Automation supports web connectivity through its web server with HTTP support. The web server consists of both HTTP and HTTPS daemons running on a switch.

Bare Metal Provisioning

Bare metal provisioning (BMP) is included as part of the Dell Networking OS image and is supported on the S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL platforms.

Introduction

BMP improves operational efficiency to your system by automatically loading pre-defined configurations and Dell Networking OS images using standard protocols such as dynamic host configuration protocol (DHCP) and common file transfer mechanisms.

Bare metal provisioning:

- Reduces the time to install and configure the network device.
- Helps eliminate configuration errors and ensure consistent configurations.
- Functions on a single system or on multiple systems.
- Includes simple network management protocol (SNMP) support.
- Includes support for pre- and post-configuration scripts.

How it Works

With BMP, the system retrieves a configuration file or a preconfiguration script indicated in the DHCP offer.

Using the preconfiguration script, you can:

- Verify the integrity of the boot image the DHCP offer downloads.
- Decide what type of configurations you want to apply based on your network reachability, port status, and neighbor discovery.
- Monitor your CPU, memory utilization, port traffic status, or perform link and topology checks with the link layer discovery protocol (LLDP).
- Retrieve and apply the configuration from a central repository.

If you disable BMP, Normal mode provides Autoexec support. Using the Autoexec feature, you can apply script-based configurations at system start-up.

Prerequisites

Before you use BMP to auto-configure your Dell Networking switch, configure the following:

- External DHCP server (required) — a network device offering configuration parameters.
- File server (required) — a network device for storing and servicing files.
- Domain name server (DNS) (optional) — a server that associates domain names in the network with IP addresses.

- Relay agent (optional) — an intermediary network device that passes messages between the DHCP clients and the DHCP server when the server is not on the same subnet. It can also provide IP addresses for multiple subnets.

For more information, refer to [Domain Name Server Settings](#) and [File Server Settings](#).

Industrial Standard Upgrades with BMP


Dell Networking OS follows the Industrial Standards with BMP. Following are the steps to perform the upgrade process:


1. Chassis Loads in BMP mode.
2. The BMP_BOOT is a special variable in Non-volatile Random Access Memory (NVRAM) for Industrial Standard Upgrades during BMP BOOT process.
3. The BMP_BOOT variable in NVRAM validates the presence of BOOT configuration.
4. If the BMP_BOOT variable is not configured, the system initiates the DHCP process and receives the offer. However, if the BMP_BOOT variable is configured, the system applies and commits the BOOT configurations present in the NORMAL_BOOT variable. Then, the system starts the DHCP process and receives the offer.
5. The offer validates the presence of image file.
6. If the image file is present in the DHCP offer, the system validates the current active boot partition with the DHCP offer image header.
7. If the image header matches, the system skips the image file download process and validates the presence of configuration file in the offer.
8. If the image header mismatches, the system initiates the image upgrade process.
9. The system validates the BMP_BOOT variable in the NVRAM as part of the upgrade process. If the BMP_BOOT variable is not configured, the system assigns the inactive boot partition to BMP_BOOT variable. Then, it stores the copy of the current BOOT parameters in NORMAL_BOOT variable of NVRAM. It also upgrades the image to the inactive boot partition.
10. The BMP process exits and the system cleans up the BMP_BOOT and NORMAL_BOOT variables, if the configuration or script file is not present in the DHCP offer. However, if the configuration or script file is present, download and apply the configuration or script file.
11. The system validates the `auto-save` BMP option. If the `auto-save` option is enabled, the system commits the configuration changes and cleans up the BMP_BOOT variables and exits the BMP process. If the `auto-save` option is disabled, the changes to the configurations do not take effect. The system cleans up the BMP_BOOT and NORMAL_BOOT variables and exits the BMP process.

BMP Process Overview

When your system boots up in default BMP mode, the following items are requested.

1. The current (new) Dell Networking OS build image.
2. The configuration file or preconfiguration script (EXPECT, TCL, or Z-shell [ZSH] script).
3. A list of checksums for all these components.

 **NOTE:** If you already installed the SmartScripts package, the preconfiguration scripts can be in Perl, Python or Ruby.

 **NOTE:** The configuration file maintains normal BMP functionality when a preconfiguration script is not sent.

BMP Operations

BMP is supported on the user ports and management ports of a switch.

If you enabled virtual link trunking (VLT) on aggregator switches and you are configuring the top-of-rack (ToR) switch to load BMP, configure the aggregator switches using the `lACP ungroup member-independent vlt` command if the DHCP and file servers are reachable using the interface configured as part of the VLT link aggregation group (LAG).

BMP eases configuration in the following ways:

- Switch access through all ports (management and user ports) with or without the DHCP-based dynamic IP address configuration of a switch.
- Booting up in Layer 3 mode with interfaces already in No Shutdown mode and basic protocols enabled to protect the system and network.
- Access to the configuration file or a preconfiguration script using the DHCP offer.
- Download and execute scripts before configurations are applied, allowing preconfiguration checks on the switch.

IOA Supports BMP

BMP is enabled on the IOA switch. Following are the characteristics of BMP feature in IOA:

- The default reload-type maintains as normal reload-type.
- The management interface initiates the DHCP discover packets.

 **NOTE:** In the IOA switch, the Management Interface is used because the physical interface does not support Internet Protocol (IP) address configurations.

- IOA switch supports the DHCP image download and upgrade processes.
- The switch also supports the DHCP configuration or script file download procedure.

Configure BMP

BMP supports two types of Reload modes: BMP mode and Normal mode.

Reload Modes

This section describes the following Reload modes:

- [BMP Mode](#)
- [Normal Mode](#)

BMP mode is the default boot mode configured for a new system arriving from the Dell factory. This mode obtains the Dell Networking OS image and configuration file from a network source (DHCP and file servers).

To boot up the switch with the management port in No Shutdown mode, use Normal mode. If the management IP address is present in the start-up configuration file, it is assigned. If the management IP

address is not present in the start-up configuration file, no IP address is assigned to the management interface.

- **BMP mode** (default) — the switch automatically configures all ports (management and user ports) as Layer 3 physical ports and acts as a DHCP client on the ports for a user-configured time (DHCP timeout). Set BMP mode using the `(conf-reload-type) # boot-type bmp-reload` command.
- **Normal mode** — the switch loads the Dell Networking OS image and startup configuration file stored in the local Flash. New configurations require that you manually configure the Management IP and Management Interface. Set Normal mode using the `(conf-reload-type) # boot-type normal-reload` command.

BMP Mode

In BMP mode, there are two types of contexts: factory-default context and normal context.


Factory-Default Context

BMP is enabled with the default parameters (`no dhcp-timeout` and `config-scr -download`). In this context, you cannot enter CLI commands. By default, the BMP syslog messages are disabled. When BMP is about to start, the following message displays:


```
This device is configured to enter Bare Metal Provisioning (BMP).
BMP will now attempt to download an image, configuration file or boot script
using DHCP.
```

You can only interact with the switch using the console. If you open the console and enter any key, the input is discarded and the following message displays:

```
This device is in Bare Metal Provisioning (BMP) mode.
BMP is attempting to download an image, configuration file or boot
script using DHCP.
To continue with the standard manual interactive mode, it is
necessary to abort BMP.
Press A to abort BMP now.
Press C to continue with BMP.
Press L to toggle BMP syslog and console messages.
Press S to display the BMP status.
[A/C/L/S]:
```


 **NOTE:** In the Factory-Default context, the console only accepts A/C/L/S input.

- Enter `s` to display the BMP status (`show boot bmp`). If you enter another key while BMP is running, it displays the previous message and repeats the process.
- Enter `A` to stop BMP. The following actions occur:
 - Aborts BMP
 - Disables BMP for the next reload (which is a Normal reload)
 - Initializes the BMP context variable in NOVRAM
 - Applies the startup configuration, if it exists, or the Default configuration
- Enter `C` to continue with the BMP process. If you enter another key while BMP is running, it displays the previous message and repeats the process.
- Enter `L` to toggle the BMP syslog messages. By default, the messages are disabled. The first `L` enables the BMP messages and the second `L` disables the BMP messages.


 **NOTE:** If the switch starts in Factory-Default context in the next reload, the BMP messages are disabled again irrespective of the `L` status in the previous session.

Normal Context

To auto-configure a switch, before you use BMP mode, first configure a DHCP, DNS, and file server in the network.

 **NOTE:** By default, a syslog severity level greater than two does not display. Enable syslog before applying the preconfiguration script which helps identify any configuration errors.

To help configure a new factory loaded switch, the switch boots up in the default BMP mode. You can reconfigure the switch to reload either in BMP mode or Normal mode.

 **NOTE:** To apply the startup configuration, cancel the default BMP setup using the `stop bmp` command from the console. To disable BMP for the next reload, use the `reload-type normal-reload` command.

Normal Mode

When reloaded in Normal mode, the switch boots up with the management port in No Shutdown mode. If the management IP address is present in the start-up configuration file, it is assigned to the management interface. If the management IP address is not present in the start-up configuration file, no IP address is assigned. You can connect to the management port with an IP address on the same network and log in to the system through a telnet or SSH session.

To configure a switch to reload using Normal mode, follow this step.

- Configure a switch to use Normal mode.
GLOBAL CONFIGURATION mode

`boot-type normal-reload`
- DHCP Configuration — Normal mode does not require a separate DHCP server configuration.
- Dell Networking OS Image Retrieval — The Dell Networking OS image is loaded from the local flash memory.

BMP Commands and Examples

You can configure BMP on supported switches using a series of commands.

To enable BMP mode on your switches, and to apply configurations or run scripts using the BMP commands, refer the [Bare Metal Provisioning CLI](#) section.

System Boot and Set-Up Behavior in BMP Mode

The following steps describe the system boot up process.

1. The system begins the boot up process in BMP mode (the default mode).
2. The system sends DHCP discover on all the interface-up ports.

```
00:02:14: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER  
sent on Ma 0/0.
```
3. The IP address, boot image filename, and configuration filename are reserved for the system and provides the DHCP reply (the one-file read method). The system receives its IP address, subnet mask, DHCP server IP, TFTP server address, DNS server IP, bootfile name, and configuration filename from the DHCP server.

4. If a DHCP offer has neither an image path nor a configuration file path, it is an invalid BMP DHCP offer and is ignored. The first DHCP offers the following to choose from:

- IP address
- Dell Networking OS image
- Configuration file or preconfiguration script
- IP address and Dell Networking OS image OR IP address and configuration file or preconfiguration script

5. DHCP OFFER is selected. All other ports except the port on which the offer was received and selected are set to Shutdown mode.

```
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_DHCP_OFFER: DHCP OFFER
received on Te 0/21 IS SELECTED.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_REQUEST: DHCP REQUEST sent
on Te 0/21.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_ACK: DHCP ACK received on
Te 0/21.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP acquired
IP 13.4.4.44 mask 255.255.255.0 server IP 13.4.4.1.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP tftp IP
NIL sname NIL dns IP NIL router IP NIL.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP image file
tftp://13.4.4.1/ftos-img-s4810.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP config
file scp://anvltest:force10@13.4.4.1//tftpboot/basic-l3.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: stacking info
NIL.
```

6. The system sends a message to the server to retrieve the named configuration file or preconfiguration script and/or boot file from the base directory of the server.

- a. If you use the optional `bootfile-name` command, the filename can be 256 bytes. If you specify the filename field in the DHCP offer, the filename can be 128 bytes. The name can be a fully qualified URL or a filename only.
- b. When a Dell Networking OS build image is found, the system compares that build image to the version currently loaded on the chassis.

- If there is a mismatch between the build images, the system upgrades to the downloaded version and reloads.

```
00:02:55: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Major Version : 1
00:02:55: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Minor Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Main Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Patch Version : 1216
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Major Version : 9
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Minor Version : 3
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Main Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Patch Version : 57
00:02:56: %STKUNIT0-M:CP %BMP-2-BMP_DOWNLOAD_START: The Dell
Networking OS image download has started.
00:03:29: %STKUNIT0-M:CP %BMP-5-BMP_DOWNLOAD: The Dell
Networking OS image download is successful.
00:03:31: %STKUNIT0-M:CP %BMP-5-BMP_MD5_VALIDATE_SUCCESS: The
Dell Networking OS image MD5 Checksum Validation Successful.
```

Erasing Sseries Primary Image, please wait

- If the versions match, the system downloads the configuration file or preconfiguration script.

```
00:03:07: %STKUNIT0-M:CP %BMP-2-BMP_DOWNLOAD_START: The config
file download has started.
00:03:19: %STKUNIT0-M:CP %BMP-5-BMP_DWNLD_FILE_IS_SCRIPT_FILE:
The downloaded file is a script file.
00:03:19: %STKUNIT0-M:CP
%BMP-5-BMP_DWNLD_CONFIG_SCRIPT_SUCCESS: The config/script file
download is successful.
00:03:21: %STKUNIT0-M:CP %BMP-5-BMP_PRE_CONFIG_SCRIPT_BEGIN:
The Pre-Config Script has started to Execute.
..
00:04:30: %STKUNIT0-M:CP %BMP-5-BMP_PRE_CONFIG_SCRIPT_END:
Pre-Config script completed with return status 0.
```

- c. If you download the configuration file or preconfiguration script from the server, the flash ignores any saved startup configuration. If you do not download the configuration file or preconfiguration script from the server, the startup configuration file on the flash loads in Normal reload.
7. When the Dell Networking OS build image and configuration file or preconfiguration script download, the IP address is released.



NOTE: In the preconfiguration script, the DHCP IP address is released after the script executes.

8. The system applies the configuration. The system is now up and running and you can manage it as usual.
9. SHA256/MD5 hash verifies the integrity of the files BMP downloads. For each file, the relevant filename.sha256 or filename.md5 must be present in the same location as the file.
 - a. When you disable FCoE initialization protocol (FIPs) mode:
 - If both filename.sha256 and filename.md5 are not present, the image is downloaded without validation.
 - If both filename.sha256 and filename.md5 are present, SHA256 validation has a higher priority compared with MD5 validation.
 - If filename.sha256 or filename.md5 alone is present, the corresponding type of validation is completed.
 - b. When you enable FIPs mode:
 - MD5 validation is not used. BMP checks only for the filename.sha256 and not filename.md5.
 - If filename.sha256 is present, it uses SHA256 validation or the image downloads without SHA256 validation.

- Syslog for SHA256VALIDATION success on downloading an image file and configuration/pre-script file:

```
00:03:07: %STKUNIT0-M:CP %BMP-5-BMP_SHA256_VALIDATE_SUCCESS: The
Dell
Networking OS image SHA256 Checksum Validation Succeeded.
```

```
00:01:15: %STKUNIT0-M:CP %BMP-5-BMP_SHA256_VALIDATE_SUCCESS: The
Config/Script SHA256 Checksum Validation Succeeded.
```

- Syslog for SHA256VALIDATION failure on downloading an image file and configuration/pre-script file:

```
00:03:22: %STKUNIT0-M:CP %BMP-2-BMP_SHA256_VALIDATE_FAILURE: The
Dell
Networking OS image SHA256 Checksum Validation Failed.
```

```
00:03:22: %STKUNIT0-M:CP %BMP-2-BMP_SHA256_VALIDATE_FAILURE: The
Config/Script SHA256 Checksum Validation Failed.
```

BMP Mode: Boot and Set-UP Behavior

When you configure a switch to reload in BMP mode, one of the following scenarios may occur.

- [Reload Without a DHCP Server Offer](#)
- [Reload with a DHCP Server Offer Without a Dell Networking OS Offer](#)
- [Reload with a DHCP Server Offer and no Configuration File](#)
- [Reload with a DHCP Server Offer Without a DNS Server](#)
- [Preconfiguration Scripts](#)
- [Post-Configuration Scripts](#)
- [Auto-Execution Scripts](#)

Reload Without a DHCP Server Offer

A switch is configured to reload in BMP mode. If the DHCP server cannot be reached, the system keeps on sending DISCOVER messages.

```
00:03:56: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Ma 0/0.
00:03:56: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Te 0/21.
00:03:55: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Te 0/8.
00:03:41: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Ma 0/0.
00:03:40: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Te 0/21.
00:03:40: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Te 0/8.
00:03:25: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Ma 0/0.
00:03:25: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Te 0/21.
00:03:25: %STKUNIT0-M:CP %BMP-5-BMP_DISCOVER: DHCP DISCOVER
sent on Te 0/8.
```

Reload with a DHCP Server Offer Without a Dell Networking OS Offer

You configure the switches to reload in BMP mode. If the switches reach a DHCP server but do not locate a downloadable Dell Networking OS image file on the server, they attempt to download the configuration file.

1. The system boots up with the BMP application.

```
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_DHCP_OFFER: DHCP OFFER
received on Te 0/21 IS SELECTED.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_REQUEST: DHCP REQUEST sent
on Te 0/21.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_ACK: DHCP ACK received on
Te 0/21.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP acquired
IP 13.4.4.44 mask 255.255.255.0 server IP 13.4.4.1.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP tftp IP
NIL sname NIL dns IP NIL router IP NIL.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP image file
NIL.
```

```
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP config
file scp://anvltest:force10@13.4.4.1//tftpboot/basic-l3.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: stacking info
NIL
```

2. If you enabled the `config-scr-download` command, the system downloads the `customer.conf` configuration file from the file-server address.
3. If the configuration download is successful, the following logs display:

```
file download has started.
00:01:23: %STKUNIT0-M:CP %BMP-5-BMP_DWNLD_FILE_IS_CONFIG_FILE:
The downloaded file is a configuration file.
00:01:23: %STKUNIT0-M:CP
%BMP-5-BMP_DWNLD_CONFIG_SCRIPT_SUCCESS: The config/script file
download is successful.
00:01:24: %STKUNIT0-M:CP %BMP-5-DOWNLOAD_INFO: /tftpboot/
signal.py config file successfully downloaded.
00:01:24: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE: DHCP RELEASE sent
on Ma 0/0.
00:01:24: %STKUNIT0-M:CP %BMP-5-BMP_DOWNLOAD: The config file
download is successful.
00:01:24: %STKUNIT0-M:CP %BMP-5-CFG_APPLY: The downloaded
config from dhcp server is being applied.
00:01:24: %STKUNIT0-M:CP %IFMGR-5-ASTATE_DN: Changed interface
Admin state to down: Ma 0/0
00:01:24: %STKUNIT0-M:CP %IFMGR-5-OSTATE_DN: Changed interface
state to down: Ma 0/0
00:01:25: %STKUNIT0-M:CP %SYS-5-CONFIG_LOAD: Loading
configuration file
```

Reload with a DHCP Server Offer and no Configuration File

A switch is configured to reload in BMP mode. If the switch reaches a DHCP server but cannot retrieve a configuration file, the switch looks for a configuration file on the file server if you enabled `config-scr-download`.

1. The system boots up with the BMP application.
2. The system receives a DHCP offer from a DHCP server with the following parameters:

```
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_DHCP_OFFER: DHCP OFFER
received on Te 0/21 IS SELECTED.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_REQUEST: DHCP REQUEST sent
on Te 0/21.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_ACK: DHCP ACK received on
Te 0/21.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_BOOT OFFER: DHCP acquired
IP 13.4.4.44 mask 255.255.255.0 server IP 13.4.4.1.
00:02:26: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP tftp IP
NIL sname NIL dns IP NIL router IP NIL.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP image file
tftp://13.4.4.1/ftos-img-s4810.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: DHCP config
file NIL.
00:02:27: %STKUNIT0-M:CP %BMP-5-BMP_BOOT_OFFER: stacking info
NIL.
```

3. The system downloads the build image from the file server.
4. The system compares the current local build image to the downloaded build image as follows:
 - a. If the build image versions match, the system does not try to load any image and comes up with the Dell prompt.

```
00:02:55: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Major Version : 1
00:02:55: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
```

```

Downloaded Image Minor Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Main Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Downloaded Image Patch Version : 1216
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Major Version : 1
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Minor Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Main Version : 0
00:02:56: %STKUNIT0-M:CP %BMP-5-BMP_RELEASE_HEADER_INFO:
Flash A Image Patch Version : 1216

```

- b. If the build images versions are different, the system stores the downloaded build image in local flash memory and loads the build image from the flash. This process repeats until the build image versions match.

The system looks for the configuration file on the file server. If the system does not find a configuration file, the download fails. If the build image processing is successful, BMP stops. If the build image download also fails, the current offer is treated as invalid and BMP tries to send DISCOVER messages.

Reload with a DHCP Server Offer Without a DNS Server

Reload the Dell Networking OS on a switch using a preconfiguration script.

To reload the Dell Networking OS on a switch using a preconfiguration script, the following must be true:

- BMP is enabled.
- The download of the script is from an external server and the location is specified in the DHCP offer.
- The first line of the script contains one of the following:

```

#!/usr/bin/expect
#!/usr/bin/tclsh
#!/usr/bin/zsh

```

- If you installed the SmartScripts package, the Dell Networking OS also supports preconfigured scripts in Perl, Python, and Ruby.

```

#!/usr/pkg/bin/perl
#!/usr/pkg/bin/python

```
- After the first line, but before the start of the script, the script contains the signature `#!/DELL-NETWORKING`.
- The preconfiguration script downloads instead of the configuration file.
- The preconfiguration script runs before applying the start-up configuration file.
- The preconfiguration script can use the Dell Networking OS CLI commands using the `f10do` utility.
- When the preconfiguration script completes, the start up configuration file automatically applies.

If a preconfiguration script file downloads successfully, the following SYSLOGs display:

```

00:03:07: %STKUNIT0-M:CP %BMP-2-BMP_DOWNLOAD_START: The config
file download has started.
00:03:19: %STKUNIT0-M:CP %BMP-5-BMP_DWNLD_FILE_IS_SCRIPT_FILE:
The downloaded file is a script file.
00:03:19: %STKUNIT0-M:CP
%BMP-5-BMP_DWNLD_CONFIG_SCRIPT_SUCCESS: The config/script file
download is successful.
00:03:21: %STKUNIT0-M:CP %BMP-5-BMP_PRE_CONFIG_SCRIPT_BEGIN:
The Pre-Config Script has started to Execute.

```

After the preconfiguration script executes successfully, the startup configuration file loads.

```

00:04:30: %STKUNIT0-M:CP %BMP-5-BMP_PRE_CONFIG_SCRIPT_END:
Pre-Config script completed with return status 0.

```

```
00:03:03: %STKUNIT0-M:CP %IFMGR-5-ASTATE_DN: Changed interface
Admin state to down: Ma 0/0.
00:03:03: %STKUNIT0-M:CP %IFMGR-5-OSTATE_DN: Changed interface
state to down: Ma 0/0.
00:03:03: %STKUNIT0-M:CP %SYS-5-CONFIG_LOAD: Loading
configuration file.
00:03:04: %STKUNIT0-M:CP %IFMGR-5-ASTATE_UP: Changed interface
Admin state to up: Ma 0/0.
```

If a post configuration script is present in the applied configuration file, it executes and the logs display:

```
00:04:22: %STKUNIT0-M:CP %BMP-5-BMP_POST_CONFIG_SCRIPT_BEGIN:
The Post-Config Script has started to Execute
..
00:05:15: %STKUNIT0-M:CP %BMP-5-BMP_POST_CONFIG_SCRIPT_END:
Post-Config script completed with return status 0.
```

If a post configuration script is not present in the applied configuration file, the logs display:

```
00:03:03: %STKUNIT0-M:CP %SYS-5-CONFIG_LOAD: Loading
configuration file.
00:03:04: %STKUNIT0-M:CP %IFMGR-5-ASTATE_UP: Changed interface
Admin state to up: Ma 0/0.
00:03:04: %STKUNIT0-M:CP %BMP-5-BMP_POST_SCRIPT_NOT_PRESENT:
The Post-Config Script is not present.
```

DHCP Offer Vendor-Specific Option for BMP

To configure FIPS mode for your system, use the vendor-specific DHCP OFFER option (code 231) called `fips-mode-enabled` so that BMP can decide to use the non-FIPS-approved algorithms for the image/config-file download process.

This option accepts the boolean type value TRUE/FALSE to enable or disable FIPS mode. BMP uses this setting temporarily and only for the image/config-file download process. BMP never stores this in either the `running-config` or `startup-config`. The DHCP OFFER controls using non-FIPS-approved algorithms by the BMP-initiated file download process. If you have not configured this option, by default BMP assumes you have disabled FIP mode.

- If the value is set to TRUE, BMP enables FIPS mode and uses the relevant FIPS-approved algorithms for the image/config-file download process.
- If the value is set to FALSE, BMP disables FIPS mode and uses the non-FIPS-approved algorithms for the image/config-file download process.

DHCP Offer Relay Option 82

Dell Networking OS supports Option 82 in DHCP offer. With option 82, the switch appends the interface details while sending the DHCP discover packet. To support option 82, the configured circuit ID and remote IDs are filled in the DHCP discover packets. The `reload-type` command has an option to enable or disable the DHCP offer option 82. By default, the option 82 is disabled. To enable the option 82, you have to configure the remote ID using the `reload-type` sub-command `relay remote-id "mac/hostname/custom string`.

- The circuit ID identifies the port originating the Protocol Data Unit (PDU) and has the interface details.
- The remote ID is configurable between the switch MAC address, the hostname, or a custom string. To set the remote ID, enable the relay up-front in `reload-type` sub mode. If the relay is enabled, by default, the remote ID is set to MAC address.

Software Upgrade Using BMP

BMP simplifies switch configuration by allowing you to boot images and run configurations that are specified in a DHCP server and also by automatically downloading files from a file server.

In a network topology with a large number of devices, use BMP to easily administer, upgrade, and maintain your devices.

Configure Auto-Configuration mode using the `reload-type bmp` command. Reload the switch in Configuration mode using the `reload` command.

Apply Configurations Using BMP Scripts

With Dell Networking OS version 9.1(0.0) or later, Open Automation supports a scripting environment when a BMP or Normal reload occurs. BMP uses pre-configuration scripts, while a Normal reload-type uses an auto-exec script. The BMP post-configuration script runs on its own, without any BMP or Normal reload-type restrictions.

Preconfiguration Scripts

In BMP, the Dell Networking OS first accesses the image, then the configuration file or preconfiguration script from the DHCP offer.

Use preconfiguration scripts to:

- verify the integrity of the Dell Networking OS image the DHCP offer downloads
- dynamically decide what types of configurations to apply to your system based on various factors such as network reachability, port status, or neighbor discovery
- use LLDP to monitor and generate reports for CPU and memory utilization, port traffic status, and to perform link and topology checking

You can provide a preconfiguration script in DHCP option 209 to either configure the switch or download a configuration file. The script can download a configuration file or apply CLI commands. If present in the downloaded configuration, post-configuration script CLIs are executed.

You can set the system to retry downloading a configuration up to six times or to automatically save the downloaded configuration or script on the switch.

After the configuration applies, it triggers a post-configuration script to ensure that the configurations and switch functions are correct.

Define configuration parameters on the DHCP server for each chassis based on the chassis MAC Address or Vendor-Class-Identifier in DHCP offer 60 or a combination of both.

The configuration scripts like PERL, Python, and Ruby are supported, if the SmartScript package is pre-installed on the system.

The system supports preconfiguration scripts in EXPECT, TCLSH, and ZSH.



NOTE: To display a properly formatted output string, Dell Networking recommends adding the following `clido` wrapper function at the beginning of TCLSH and EXPECT scripts:

```
# Execute clido and return the output string
proc Execclido {cmd_str} {
  set str [exec clido "$cmd_str"]
  set tmp_str [string map {\n \r\n} $str ]
  return $tmp_str
}
...
set out_str [Execclido "show version"]
puts $out_str
...
```

Auto-Execution Scripts

The autoexecution (`autoexec`) script is the same as a preconfiguration script except that it executes on every reboot in Normal mode.

Store scripts in a `flash://autoexec` file. Autoexec scripts are independent of BMP.

The `autoexec` script only executes when:

- BMP is disabled.
- The script is stored in a `flash://autoexec` file.
- Before reloading the system, you use the `reload-type normal-reload` command in the `reload-type` sub command mode.

If the `autoexec` script fails, the system generates a message indicating the failure and does not load the configuration file. Before continuing the upgrade, correct the error in the script.

Post-Configuration Scripts

The post configuration script runs once the start-up configuration is applied as `script post-config script-name`. In BMP mode, after the pre-configuration script completes and the configuration loads, you can run a post-configuration script if one is present in the configuration file. In normal reload, if there is `script post-config script-name` config in the startup config file, post-configuration script is executed.

To check the status of configured ports or protocols, set the host name of the system, or perform additional configuration settings, use the post-configuration script. The system supports post-configuration scripts in EXPECT, TCLSH, and ZSH. If you installed the SmartScripts package in your Dell Networking switches, the system also supports post-configuration scripts in Perl, Python, and Ruby.

Using Preconfiguration Scripts

To preconfigure scripts, follow these steps:

1. Decide what information you want to preconfigure; for example, request username and password information. Verify the integrity of the boot image the DHCP offer downloads and apply the configuration types.
2. Create a preconfiguration script in EXPECT, TCL or ZSH.
3. Store the script on any TFTP/FTP/SFTP server that is reachable from the system and mention the URL file (TFTP/FTP/SFTP) in the DHCP offer.
4. Change the `reload-type` to `BMP` and reload the switch. The system boots in BMP mode.

5. The system receives an IP address via the DHCP server which it uses to get a Dell Networking OS image to boot, a configuration file (if supplied), and a preconfiguration script.
6. The system runs the preconfiguration script.
 - The default timer on the script is 10 minutes. The maximum amount of time the script can run is one hour.
7. The preconfiguration script can access Dell Networking OS CLI commands through the `clido` utility. The `clido` utility has no pagination, is always set to `terminal length 0`, and has Dell Networking OS CLI privilege 15 enabled. It works in the following modes:
 - Continuous mode (`clido` command) — use to retain the Dell Networking OS context.
 - Reset mode (`clido -r` command) — use to reset the CLI command to its original context.
8. To execute CLI commands during boot time, the system uses the `clido` utility.

Using the Post-Configuration Script

To run the post-configuration script, ensure the following:

- You can write the post-configuration script in EXPECT, TCLSH, or ZSH. If you installed the SmartScripts package, you can also write the post-configuration script in Perl, Python, or Ruby.
- No restraints are required for the post-configuration script; for example, the signature `#/DELL-NETWORKING` that is required for the preconfiguration script is not required for the post-configuration script.
- Configure the post-configuration script by using the `script post-config` command.
- Reload the device (either in BMP mode or Normal reload-type mode).
- Execute the post-configuration script after the start-up configuration process is applied.
- The post-configuration script has the ability to use Dell Networking OS CLI commands using the utility name `clido`.
- If the `script post-config script` is present in the startup-config file, irrespective of the reload-type, upon reload and after the configuration file is loaded, the post-config runs.

Using Auto-Execution Script (Normal Mode Only)

To use the autoexec script, the following conditions must be true:

- BMP is disabled.
- The autoexecution script is stored in `flash://autoexec`.
- You can write the autoexecution script in EXPECT, TCLSH, or ZSH. If you installed the SmartScripts package, you can also write the post-configuration script in Perl, Python or Ruby.
- No restraints are required for the autoexec script; for example, the signature `#/DELL-NETWORKING` that is required for the preconfiguration script is not required for the post-configuration script.
- The autoexecution script can use Dell Networking OS CLIs using the utility name `F10do`.
- When the autoexecution script completes, the start-up configuration applies depending on the return status of the script:
 - `Success=0` — the start-up configuration applies.
 - `Failure-non-zero` — the start-up configuration does not apply.
- If you reboot the system with the reload-type set as `normal-reload` and an autoexec script is present in the Flash directory, the following log displays:

```
Starting Dell Networking application
00:00:13: %STKUNIT1-M:CP %RAM-6-ELECTION_ROLE: Stack unit 1 is
transitioning to Management unit.
```

```
00:00:15: %STKUNIT1-M:CP %CHMGR-5-STACKUNITDETECTED: Stack unit
1 present
00:01:02: %STKUNIT0-M:CP %BMP-5-AUTOEXEC_START: The AutoExec
Script is Started.
```

- After the script execution is successful, Dell Networking OS log displays:

```
00:04:05: %STKUNIT0-M:CP %BMP-5-AUTOEXEC_SUCCESS: The AutoExec
Script execution returned Success.
copy startup-config running-config
!
4774 bytes successfully copied
00:04:06: %STKUNIT0-M:CP %SYS-5-CONFIG_LOAD: Loading
configuration file
```

- If the script returns a non-zero value (indicating a failure), the configuration file does not load in the system. Debug it manually and reload the system with the corrected script. If the autoexec script fails, the system syslog displays:

```
00:00:34: %STKUNIT1-M:CP %BMP-5-AUTOEXE_FAILURE: The AutoExec
Script execution returned Failure.
```

Timers in Pre–configuration and Post–configuration Scripts

To avoid the scripts executing for a long time or an infinitely running script, Dell Networking OS has a session timer of maximum one hour. After this time, the device reloads and you can go to the DHCP server and correct the script errors.

You can manually control the maximum execution time of the script using the Dell Networking OS proprietary utility, `rstimer`. This utility resets or reloads the device after the mentioned time in the `rstimer`.

The timer can be set as `rstimer time`.

The `rstimer` range is from 1 to 60 minutes. The default is 10 minutes.

You can set this utility in pre–configuration or post–configuration script and this timer is triggered only after the line of execution of `rstimer` time. For example, if a script is running for 8 minutes and the next line of execution is `rstimer 4`, the device reloads only after 12 minutes from the start of script execution.

Script Examples

The following are BMP script examples.

Auto-Execution Script – Normal Mode

```
Dell#show reload-type
Reload-Type : normal-reload [Next boot : normal-reload]
Dell#show file flash://autoexec
#! /usr/bin/tclsh
puts [ exec f10do "show version" ]
puts [ exec date ]
puts "this is Autoexec script"
Dell#
Dell#
Dell#reload
System configuration has been modified. Save? [yes/no]: no
Proceed with reload [confirm yes/no]: yes
```

```
00:32:16: %STKUNIT1-M:CP %CHMGR-5-RELOAD: User request to
reload the chassis syncing disks... done
unmounting file systems...
unmounting /f10/flash (/dev/ld0h)...
unmounting /usr/pkg (/dev/ld0g)...
unmounting /usr (mfs:35)...
unmounting /f10 (mfs:21)...
unmounting /kern (kernfs)...
unmounting / (/dev/md0a)... done
rebooting
..
..
..
Starting Dell Networking application
00:00:13: %STKUNIT1-M:CP %RAM-6-ELECTION_ROLE: Stack unit 1 is
transitioning to Managementunit.
00:00:15: %STKUNIT1-M:CP %CHMGR-5-STACKUNITDETECTED: Stack unit
1 present.
```

The following line indicates the start of the auto-execution script.

```
Script is Started.
00:00:19: %STKUNIT1-M:CP %CHMGR-5-CHECKIN: Checkin from Stack
unit 1 (type S4810, 64 ports)
00:00:20: %00:00:20: %STKUNIT1-M:CP %CHMGR-0-PS_UP: Power
supply 0 in unit 1 is up
00:00:20: %STKUNIT1-M:CP %CHMGR-5-STACKUNITUP: Stack unit 1 is
up
00:00:21: %STKUNIT1-M:CP %CHMGR-5-SYSTEM_READY: System ready
00:00:21: %STKUNIT1-M:CP %RAM-5-STACK_STATE: Stack unit 1 is in
Active State.
00:00:22: %STKUNIT1-M:CP %IFMGR-5-OSTATE_UP: Changed interface
state to up: Ma 1/0
00:00:26: %S4810:1 %IFAGT-5-INSERT_OPTICS: Optics SFP inserted
in slot 1 port 30
00:00:27: %STKUNIT1-M:CP %CHMGR-5-PSU_FAN_UP: Fan 0 in PSU 0 of
Unit 1 is up
00:00:29: %S4810:1 %IFAGT-5-INSERT_OPTICS_PLUS: Optics SFP+
inserted in slot 1 port 11
00:00:36: %STKUNIT1-M:CP
%IFMGR-5-IFM_ISCSI_ACL_REGION_NOT_ALLOCATED: iSCSI Session
monitoring cannot be enabled without ACL regions allocated to
it. To enable iSCSI SessionMonitoring allocate cam-blocks to
iscsiptacl using cam-acl CLI and then save and reload.
00:00:36: %STKUNIT1-M:CP %IFMGR-5-IFM_ISCSI_ENABLE: iSCSI has
been enabled causing flowcontrol to be enabled on all
interfaces. EQL detection and enabling iscsi
profile-compellent on an interface may cause some automatic
configurations to occur like jumbo frames on allports and no
storm control and spanning tree port-fast on the port of
detection
00:00:36: %STKUNIT1-M:CP %SEC-5-LOGIN_SUCCESS: Login successful
on consoleDell>Dell#terminal length 0
```

The following line indicates that the auto-execution script is executing.

```
Dell#show version
Dell Networking Real Time Operating System Software
Dell Networking Operating System Version: 2.0
Dell Networking Application Software Version: 1-0(0-338)
Copyright (c) 1999-2012 by Dell Inc. All Rights Reserved.
Build Time: Thu Dec 27 21:32:28 2012
Build Path: /sites/sjc/work/build/buildSpaces/build06/
```

```

FIT-INDUS-1-0-0/SW/SRC
System image file is "dt-maa-s4810-72"
System Type: S4810Control
Processor: Freescale QorIQ P2020 with 2147483648 bytes of
memory.
128M bytes of boot flash memory.
1 52-port GE/TE/FG (SE)48 Ten GigabitEthernet/IEEE 802.3
interface(s)
4 Forty GigabitEthernet/IEEE 802.3 interface(s)
Dell#Wed Jan 2 22:47:34 GMT 2013
this is Autoexec script

```

The following line indicates the auto-execution script has completed successfully.

```

00:04:05: %STKUNIT0-M:CP %BMP-5-AUTOEXEC_SUCCESS: The AutoExec
Script execution returned Success.

```

The following line indicates that the configuration file is loaded into the switch.

```

Dell#00:00:51: %STKUNIT1-M:CP %SYS-5-CONFIG_LOAD:
Loading configuration file
00:00:52: %STKUNIT1-M:CP %IFMGR-5-ASTATE_UP: Changed interface
Admin state to up: Te 0/36
00:00:53: %STKUNIT1-M:CP %IFMGR-5-ASTATE_DN: Changed interface
Admin state to down: Ma 0/0

```

Pre-configuration Script – BMP Mode

```

#!/usr/bin/expect
#/DELL-NETWORKING
# Execute F10do and Print
proc print_f10do {cmd_str} {
set str [exec f10do "$cmd_str"]
set tmp_str [string map {\n \r\n} $str ]
puts $tmp_str
}
set ftp_ip "20.0.0.1"
set ftp_username "lab"
set ftp_passwd "lab"
set config_file "s4810-10-startup-config"
set post_conf "s4810-10-post-config.exp"
puts "Executing Pre-Config Script !!!\r\n"
exec rm -rf "$config_file"
exec rm -rf "$post_conf"
puts "Downloading Startup Config and Post-Config Script from
$ftp_ip ...\r\n"
spawn ftp "$ftp_ip"
expect "Name .*: "
send "$ftp_username\n"
expect "Password: "
send "$ftp_passwd\n"
send "cd scripts\n"
expect "ftp>"
send "ls\n"
expect "ftp>"
send "get $post_conf\n"
expect "ftp>"
send "get $config_file\n"
expect "ftp>"
send "bye\n"
expect eof
after 5000
puts "Download Complete !!!\r\n"
if {[file exists $config_file]} {

```

```

puts "Config File: $config_file downloaded successfully\r\n"
} else {
puts "ERROR: Config File: $config_file - Not Found\r\n"
}
if {[file exists $post_conf]} {
puts "Post Config Script: $post_conf downloaded
successfully\r\n"
} else {
puts "ERROR: Post Config Script: $post_conf - Not
Found\r\n"
# Copy Config to Startup Config
print_f10do "show version"
after 5000
print_f10do "copy flash://$config_file startup-config"
print_f10do "yes"
after 5000
puts "Pre-Config Script Execution Successful !!!!!\r\n"
exit 0

```

Post-configuration Script – BMP Mode

The following example shows the post-configuration script for the S4810 or S4820T platform:

```

#!/usr/bin/expect
#/DELL-NETWORKING
# Post Config Script for S4810-10
# Execute F10do and Print
proc print_f10do {cmd_str} {
set str [exec f10do "$cmd_str"]
set tmp_str [string map {\n \r\n} $str ]
puts $tmp_str
}
# Interfaces Configurations
set interface_list_slot [list "0/1" "0/5" "0/9"]
set interface_list [list "Te 0/1" "Te 0/5" "Te 0/9"]
set {remote_intf(Te 0/1)} "TenGigabitEthernet 0/1"
set {remote_intf(Te 0/5)} "TenGigabitEthernet 0/5"
set {remote_intf(Te 0/9)} "TenGigabitEthernet 0/9"
set hostname "S4810-10"
set max_min 10
set status_file "s4810-10-current-status.dat"
set ftp_ip "20.0.0.1"
puts [exec rstimer 30]
puts "\r\nReset Timer Complete\r\n"
# Open Staus File
set fp [open $status_file w]
puts $fp "=====\r\n"
puts $fp " Status: $hostname\r\n"
puts $fp "=====\r\n"
# Configure LLDP Protocol
puts "Configuring LLDP Protocol\r\n"
print_f10do "configure terminal"
print_f10do "protocol lldp"
print_f10do "no disable"
print_f10do "end"
# Check for Protocl LLDP
set lldp_output [ exec f10do "show runn | grep lldp" ]
if {[regexp "lldp" $lldp_output]} {
puts "LLDP is configured\r\n"
# Write the Result to Status File
puts $fp "LLDP is configured\n"
} else {
puts "ERROR: LLDP is not configured\r\n"
}

```

```

# Write the Result to Status File
puts $fp "ERROR: LLDP is not configured\r\n"
}
# Configure Interfaces
foreach intf_slot $interface_list_slot {
set intf "TenGigabitEthernet $intf_slot"
puts "Configuring $intf ...\n"
puts $fp "Configuring $intf ...\n"
print_f10do "configure terminal"
print_f10do "interface $intf"
print_f10do "no ip address"
print_f10do "no shutdown"
print_f10do "end"
after 200
}
# Wait for 2 mins for the Neighbor to come-up
puts "Wait for 1 min for the Neighbor to come-up\r\n"
after [expr {60 * 1000}]
puts $fp "=====\n"
puts $fp " Checking Conectivity thru LLDP\n"
puts $fp "=====\n"
# Check LLDP Configurations
foreach intf_slot $interface_list {
set min 0
set result 0
while {$result == 0 && $min < 5} {
set result_str [exec f10do "show lldp neighbors | grep
\"$intf_slot\""]
set tmp_str [string map {\n \r\n} $result_str]
puts $tmp_str
if {[regexp "$intf_slot" $result_str]} {
set result 1
if {[regexp "$remote_intf($intf_slot)" $result_str]} {
puts "Interface $intf_slot is Connected to
$remote_intf($intf_slot)\r\n"
puts $fp "Interface $intf_slot is Connected to
$remote_intf($intf_slot)\r\n"
} else {
puts "ERROR: Interface $intf_slot is Not Connected to Interface
$remote_intf($intf_slot)\r\n"
puts $fp "ERROR: Interface $intf_slot is Not Connected to
Interface$remote_intf($intf_slot) \r\n"
puts "LLDP Output for $intf_slot :\r\n $result_str \r\n"
puts $fp "LLDP Output for $intf_slot :\r\n $result_str \r\n"
}
}
continue
}
}
# Wait for 1 minute
puts "Interface is Not Connected\r\n"
puts "Wait for 1 min for the Neighbor to come-up\r\n"
after [expr {60 * 1000}]incr min
}
if {$result == 1} {
puts "Interface $intf_slot is Connected\r\n"
# Write Result to Status File
puts $fp "Interface $intf_slot is Connected\r\n"
} else {
puts "ERROR: Interface $intf_slot is Not Connected\r\n"
# Write the Result to Status File
puts $fp "ERROR: Interface
$intf_slot is Not Connected\r\n"
}}
# Close & FTP Status File
puts $fp "=====\n"
close $fp

```

```

# Configure FTP - Interface
print_f10do "configure terminal"
print_f10do "interface TenGigabitEthernet 0/22"
print_f10do "ip address 20.0.0.34/16"
print_f10do "no shutdown"
print_f10do "end"
puts "Uploading Status File($status_file) to $ftp_ip ...\n"
spawn ftp "$ftp_ip"
expect "Name .*: "
send "lab\n"
expect "Password:"
send "lab\n"
expect "ftp>"
send "cd scripts\n"
expect "ftp>"
send "ls\n"
expect "ftp>"
send "put $status_file\n"
expect "ftp>"
send "ls\n"
expect "ftp>"
send "bye\n"
expect eof
print_f10do "configure terminal"
print_f10do "interface TenGigabitEthernet 0/22"
print_f10do "no ip address"
print_f10do "shutdown"
print_f10do "end"
puts "Post-Config Script Execution Successfull !!!!!\r\n"
exit 0

```

BMP Operations on Servers Overview

The following sections describe how to prepare the different servers for BMP functionality.

- [Configuring the DHCP Server Settings](#)
- [File Server Settings](#)
- [Domain Name Server Settings](#)

DHCP Server

To configure the DHCP server use the following information.

Configuring the DHCP Server Settings

Before you can use BMP mode on a switch, first configure a DHCP server.


To assign an IP address to the system and other parameters, configure the DHCP server. For more information, refer to the *Dell Networking OS Configuration Guide*, the *Dynamic Host Configuration Protocol* chapter.

- Configure the DHCP server with the following set of parameters for each client switch. Update the following parameters on the appropriate DHCP server:
 - **Boot File Name** — the Dell Networking OS image loaded on the system. Option 67 in the DHCP offer is the boot filename; the filename is BOOTP payload. If you specify both, option 67 is used. The system supports the TFTP, HTTP, HTTPS, SFTP, SCP, FTP, FLASH, and USBFLASH protocols.

- **Configuration File Name** – the configurations applied to the system. The configuration filename is expected to use option 209. To configure the device by itself for a download configuration file, you can also create a preconfiguration script in option 209.
- **File Server Address** – the server where the image and configurations files are placed. The address is assumed to be a TFTP address unless it is given as a URL. The system supports the TFTP, HTTP, HTTPS, SFTP, SCP, and FTP protocols, as well as files stored in flash.
- **Domain Name Server (Optional)** – the DNS server contacted to resolve the host name.
- **IP Address** – dynamic IP address for the system. Use this IP address for file transfers only.

The following lists the DHCP option codes.

| DHCP Option Codes | Description |
|-------------------|-------------------------|
| 6 | Domain Name Server IP |
| 60 | Vendor class identifier |
| 61 | Class identifier |
| 66 | TFTP Server name |
| 67 | Boot filename |
| 150 | TFTP server IP address |
| 209 | Configuration file |
| 230 | User port stacking |

 **NOTE:** BMP eventually exits when a timeout occurs.

In the following scenarios, BMP requests a different DHCP offer.

- If the offer contains only a boot image that cannot be downloaded, BMP requests another DHCP offer.
- If you enable the `reload-type config-scr-download enable` command and the configuration file in the offer cannot be downloaded, BMP requests another DHCP offer.

DHCP Server IP Blacklist

If the process does not complete successfully, the DHCP server IP is blacklisted and the BMP process re-initiates.

A DHCP server IP is maintained in the blacklist for 10 minutes. If a DHCP offer is received from the blacklisted DHCP server, the offer is rejected until the IP is alive in the blacklist (10 minutes).

MAC-Based Configuration

To configure the DHCP server to assign a fixed IP address, Dell Networking OS image, and configuration file based on the system's MAC address, use BMP mode.

Using BMP, the same IP address is assigned to the system even on repetitive reloads and the same configuration file is retrieved when using the DNS server or the network-config file to determine the hostname.

The assigned IP address is only used to retrieve the files from the file server. It is discarded after the files are retrieved.

Following is a configuration example of a DHCP server included on the most popular Linux distribution. The `dhcpd.conf` file shows that the MAC-based IP and configuration file assignment are fixed.

```
option configfile code 209=text;
option bootfile-name code 67=text;
host HOST1{

#####MAC to IP mapping
hardware ethernet 00:01:e8:8c:4d:0e;
fixed-address 30.0.0.20;
```

FTP URL with IP address

HTTP URL with DNS

TFTP

```
#####bootfile-name could be given in the following way
option bootfile-name "ftp://
admin:admin@Guest-1/Dell-SE-8.3.10.1.bin";
option bootfile-name "http://30.0.0.1/
Dell-SE-8.3.10.1.bin";
```

FTP URL with DNS

HTTP URL with IP address

TFTP URL with IP address

MAC-Based IP Address Assignment

To assign a fixed IP address and configuration file based on the system's MAC address, configure the DHCP server to deploy in BMP mode.

In this way, the same IP address is assigned and the same configuration file is retrieved when the switch reloads.

Using a dynamic IP address assignment may cause the desired configuration to not load on the system because the IP address changes each time the system is reloaded.

For example, on a DHCP3 server, you can configure the assignment of a fixed MAC-based IP address and configuration file by entering the following lines of configuration parameters in the `dhcpd.conf` file on the server:

```
host S4810 {
hardware ethernet 00:01:e8:81:e2:39;
fixed-address 20.0.0.48;
option configfile "customer.conf";
}
```

Class-Based Configuration

By matching a part of the string from the vendor class identifier option 60 string, the image, configuration file, or script file is sent in the DHCP offer.

For example:

```
host dt-maa-z9000-11 {
  hardware ethernet 00:01:e8:a9:81:a3;
  fixed-address 10.16.151.175;
  option tftp-server-address 10.16.151.209;
  match if substring (option vendor-class-identifier,0,17) =
  "TY=DELLNTW-Z9000 "; {
  filename "tftp://10.16.151.209/FTOS-ZB-9.3.0.0.bin";
  option configfile "ftp://anvltest:force10@10.16.151.209//
  tftpboot/basic-l3-z9k <ftp://10.16.151.209//tftpboot/
  basic-l3-z9k> ";
  }
  match if substring (option vendor-class-identifier,0,17) =
  "TY=DELLNTW-s4810 "; {
  filename "tftp://10.16.151.209/FTOS-SE-9.3.0.0.bin";
  option configfile "ftp://anvltest:force10@13.4.4.4//
  tftpboot/basic-l3-across-s6410-s4810_z9k.cfg";
  }
}
```

Option 60 is also called vendor class identifier. It has five fixed fields that have fixed offsets. The fixed fields are:

- TY — denotes the type of the device
- HW — denotes the version of the box
- SN — denotes the serial number of the device
- ST — denotes the service tag of the device
- OS — refers to the Dell Networking OS version in the device
- US — refers to the user-defined string (a maximum string of 64 characters)

File Server Settings

Set up a file server and ensure connectivity.

To allow file transfers to the switch, configure the file server that holds the boot and configuration files. The system recognizes HTTP, HTTPS, SFTP, TFTP, FTP, USB, and Flash URLs.

For example:

- tftp://server ip or name/filename
- ftp://user:passwd@serverip or name//mypath/Dell-A.B.C.D.bin
- flash://filename
- http://host[:port]/file-path
- https://[<user:pass>@]host[:port]/file-path
- sftp://user:passwd@server//path/filename
- usbflash://path/filename

When loading the Dell Networking OS image, if the Dell Networking OS image on the server is different from the image on the local flash, the system downloads the image from the server onto the local flash

and reloads using that image. If the Dell Networking OS image on the server is the same image, the system loads the configuration file, if present, or the startup-config without downloading a new image.

Domain Name Server Settings

To determine the host name applied in the switch startup configuration, when no configuration file is retrieved from the DHCP server, set up a domain name server (DNS).

The DNS is contacted only when no configuration file is contained in a DHCP server response and the host name is not resolved from the network-config file on the switch.

BMP MIB Support

The Management Information Base (MIBs) and the corresponding configurations are as follows:

| MIB OID | Value and Functionality |
|-------------------------------|--|
| 1.3.6.1.4.1.6027.3.2 3.1.1 | 1 or 2 (BMP or Normal <code>reload-type</code>) |
| 1.3.6.1.4.1.6027.3.2 3.1.2 | 1 or 2 (Enable or Disable <code>auto-save</code> option) |
| 1.3.6.1.4.1.6027.3.2 3.1.3 | 1 or 2 (Enable or Disable <code>config-scr download</code> option) |
| 1.3.6.1.4.1.6027.3.2 3.1.4 | 0 - 50 (DHCP timeout value config) |
| 1.3.6.1.4.1.6027.3.2 3.1.5 | 0 - 50 (Set BMP Retry count) |
| 1.3.6.1.4.1.6027.3.2 3.1.6 | String (Set the vendor-class identifier values up to a maximum of 60 characters) |
| 1.3.6.1.4.1.6027.3.2 3.1.7 | 1 or 2 (Enable or Disable <code>relay</code> option) |
| 1.3.6.1.4.1.6027.3.2 3.1.8 | String (Set the mac/hostname/custom-string as <code>remote-id</code> option in DHCP offer option 82) |

Bare Metal Provisioning CLI

Bare metal provisioning commands are supported on the S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL platforms.

In a data center network, BMP automates configuring and updating the switches, ensuring standard configurations across the installed devices.

To set up a single switch or a stack of switches with minimal effort, use the auto-configuration function.

On S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL switch platforms running BMP:

- Specifies boot images and configuration files in a dynamic host configuration protocol (DHCP) server.
- Supports preconfiguration and postconfiguration scripts to automatically load predefined configurations and Dell Networking OS images using DHCP and common file transfer mechanisms.
- Allows switch access through all ports (management and user ports) with or without a DHCP-based dynamic IP address configuration of the switch.
- Automatically downloads configuration files from a file server and applies them to the switch or stack.
- Boots up in Layer 3 mode with interfaces already in No Shutdown mode. Some basic protocols are enabled to protect the system and network.

auto-save

Allows you to configure the auto save option for the downloaded configuration or script file which are not saved by default.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

| | | |
|----------------------|--|---|
| Syntax | <code>auto-save { enable disable }</code> | |
| | To reset the auto-save option, use the <code>no auto-save { enable disable }</code> command. | |
| Parameters | enable | Enable the option to save the downloaded configuration or script file. |
| | disable | Disable the option to save the downloaded configuration or script file. |
| Defaults | none | |
| Command Modes | RELOAD-TYPE CONFIGURATION | |

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000–ON, Z9000, Z9500, IOA, and MXL. |

Related Commands

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

bmp logging

Configure the syslog messages in a console while BMP is running.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `bmp logging {enable | disable}`

Parameters

| | |
|----------------|---|
| enable | Enables the syslog messages in the console while BMP is running. |
| disable | Disables the syslog messages in the console while BMP is running. |

Defaults none

Command Modes EXEC


Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S6000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.5(0.0) | Introduced on the S4810, S4820T, S6000, Z9000, and MXL. |

Usage Information This command is applicable only when BMP is running and enables or disables the syslog messages (with a Severity level >2, critical).

When BMP is not running, the following error message displays: % Error: bmp process is not running.

 **NOTE:** This CLI command is only available in BMP normal context, which is similar to providing the `L` option in the Factory-default context.

boot-type

Allows you to configure the reload type.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

Syntax `boot-type { bmp-reload | normal-reload }`
To reset the boot-type mode, use the `no boot-type { bmp-reload | normal-reload }` command.

Parameters

| | |
|----------------------|--|
| bmp-reload | Reloads the system in BMP mode. After performing the reload, the system retrieves the Dell Networking OS image and startup-configuration files from the flash. |
| normal-reload | Reloads the system in manual interactive mode. |

Defaults none

Command Modes RELOAD-TYPE CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

Related Commands

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

config-scr-download

Allows you to specify whether the configuration or script file needs to be downloaded.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

Syntax `config-scr-download { enable | disable }`
To reset the download option, use the `no config-scr-download { enable | disable }` command.

Parameters

| | |
|--|--|
| <code>config-scr-download {enable}</code> | Enable downloading a configuration file or preconfiguration script from the DHCP / file servers. |
| <code>config-scr-download {disable}</code> | Disable downloading a configuration file or preconfiguration script from the DHCP/ file servers. |

Defaults none

Command Modes RELOAD-TYPE CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, refer to the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

Usage Information For an initial setup, the `config-scr-download` parameter of the `reload-type` command is enabled. After the configuration file successfully downloads, the `config-scr-download` parameter is automatically disabled. You can enable it again using the `reload-type` command.


Related Commands

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

dhcp-timeout

Allows you to configure the DHCP timeout limit.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

| | | | |
|-----------------------------|---|-----------------------------|---|
| Syntax | <code>dhcp-timeout minutes</code> To reset the DHCP time out limit, use the <code>no dhcp-timeout minutes</code> command. | | |
| Parameters | <table><tr><td>dhcp-timeout minutes</td><td>Configure the DHCP timeout (in minutes) after which the BMP exits. The range is from 0 to 50. If you enter a range of 0, the timeout is 0 (no limit).</td></tr></table> <p> NOTE: Dell Networking recommends setting the value to 2 or higher.</p> | dhcp-timeout minutes | Configure the DHCP timeout (in minutes) after which the BMP exits. The range is from 0 to 50. If you enter a range of 0, the timeout is 0 (no limit). |
| dhcp-timeout minutes | Configure the DHCP timeout (in minutes) after which the BMP exits. The range is from 0 to 50. If you enter a range of 0, the timeout is 0 (no limit). | | |

Defaults Disabled

Command Modes RELOAD-TYPE CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

- Related Commands**
- [reload-type](#)
 - allows you to configure the reload-type options in reload-type sub command mode.
 - [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

relay

Allows you to configure the addition of option 82 in DHCP client packets.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

Syntax `relay { enable | disable | remote-id { hostname | MAC | Word } }`

To reset the options 82 information, use the `no relay { enable | disable | remote-id { hostname | MAC | Word } } command`.

Parameters

| | |
|---|--|
| enable | Enable the option 82 in BMP. The default value is the MAC address. |
| disable | Disable the option 82 in BMP. |
| remote-id { <i>hostname</i> <i>MAC</i> <i>Word</i> } | Enter the keyword <code>remote-id</code> , then the hostname or chassis MAC address or a text string to set as the remote ID in option 82. The default is the remote MAC ID for option 82. |

Defaults none

Command Modes RELOAD-TYPE CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

Related

Commands

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

reload conditional nvram-cfg-change

After saving the BMP configuration, to perform a reload on the chassis due to configuration changes that changes the NVRAM content, use this command.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL Switch

Syntax `reload conditional nvram-cfg-change`

Defaults none

Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|------------------------------------|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the MXL Switch. |
| 9.1(0.0) | Introduced on the Z9000 and S4810. |
| 8.3.19.0 | Introduced on the S4820T. |

Usage Information To upgrade any NVRAM changes to the chassis caused by the following CLI commands, use the `reload conditional nvram-cfg-change` command on a switch running BMP.

- `stack-unit {unit} stack-group`
- `stack-unit {unit} port {fanout-portid} portmode quad`
- `asf-mode`
- `cam-acl`
- `buffer-profile`

reload-type

Allows you to configure the `reload-type` options in `reload-type` sub command mode.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

| | |
|------------------------|---|
| Syntax | <code>reload-type</code> |
| Defaults | BMP |
| Command Modes | CONFIGURATION |
| Command History | This guide is platform-specific. For command information about other platforms, see the relevant <i>Dell Networking OS Command Line Reference Guide</i> . |

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |
| 9.6(0.0) | Introduced on the S5000. |

| Version | Description |
|----------|---|
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the MXL Switch. Updated the parameters for S4810, S4820T, and Z9000. |
| 9.1(0.0) | Updated the command mode from EXEC Privilege to GLOBAL CONFIGURATION. Updated the parameter from <code>jumpstart</code> to <code>bmp</code> . Added support for the <code>config-scr-download</code> and <code>user-defined-string</code> commands. Supported platforms are S4810, S4820T, and Z9000. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced in Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.1.0 | Introduced on the S4810. |

Related Commands

- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).
- [stop bmp](#) — stops the BMP process and prevents a loop if the DHCP server is not found.

retry-count

Allows you to configure the number of retries for downloading the Dell Networking OS image and configuration file.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

Syntax

`retry-count number`

To reset the number of retries, use the `no retry-count number`

Parameters

**retry-count
number**

Enter the keyword `retry-count`, then the number of times to retry downloading the Dell Networking OS image, configuration file, or script file from the DHCP / file servers, if the servers are not reachable. The retry limit is from 0 to 6. If the retry limit is 0, no retry performs. The default is 0.

Defaults

none

Command Modes

RELOAD-TYPE CONFIGURATION

Command History

This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

Related Commands

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

script post-config

To run the post-configuration script after the preconfiguration script is executed during the BMP reload, ensure that this command is present in the startup-configuration.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `script post-config {script-name}`

Parameters

| | |
|--------------------------|--|
| <i>scriptname</i> | Enter the name of the script to run after the BMP start-up configuration is applied. |
|--------------------------|--|

Defaults none

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the MXL switch and S4820T. |
| 9.1(0.0) | Introduced on the Z9000 and S4810. |

show boot bmp

Displays the current state of the BMP process.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

| | |
|------------------------|---|
| Syntax | show boot bmp |
| Defaults | none |
| Command Modes | EXEC |
| Command History | This guide is platform-specific. For command information about other platforms, see the relevant <i>Dell Networking OS Command Line Reference Guide</i> . |

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the Z9000, S4810, S4820T, and MXL switch. |

Example

```
Dell# show boot bmp
Config Download
via DHCP: enabled
BMP State : Waiting for boot options
...
BMP State : Received DHCP offer from DHCP server 25.1.1.1
***** SELECTED OFFER DETAILS *****
Server type= DHCP
Acquired IP= 25.1.1.25
Subnet-mask = 255.255.0.0
Image file = tftp://25.1.1.1/boot_file.bin
config file = tftp://25.1.1.1/config_file.cfg
Server IP = 25.1.1.1
TFTP Server IP = NIL
DNS IP = 25.1.1.1
Routers = NIL
*****
...
BMP State : Downloading image boot_file.bin from 25.1.1.1
...
BMP State : Image boot_file.vin successfully downloaded
...
BMP State : BMP process is successfully completed
```

**Related
Commands**

[reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.

show config

Displays the reload type options that are configured on the system.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL Switch

| | |
|----------------------|---------------------------|
| Syntax | show config |
| Defaults | none |
| Command Modes | RELOAD-TYPE CONFIGURATION |

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

Example

```
Dell(conf-reload-type)#show config
!
reload-type
auto-save enable
boot-type bmp-reload
config-scr-download enable
dhcp-timeout 12
retry-count 3
vendor-class-identifier Sample
relay enable
relay remote-id TEST
Dell(conf-reload-type)#
```

**Related
Commands**

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [auto-save](#) — allows you to configure the auto save option for the downloaded configuration or script file which are not saved by default.
- [boot-type](#) — allows you to configure the reload type.
- [config-scr-download](#) — allows you to specify whether the configuration or script file needs to be downloaded.
- [dhcp-timeout](#) — allows you to configure the DHCP timeout limit.

- [relay](#) — allows you to configure the addition of option 82 in DHCP client packets.
- [retry-count](#) — allows you to configure the number of retries for downloading the Dell Networking OS image and configuration file.
- [vendor-class-identifier](#) — allows you to configure the optional vendor class identifier for DHCP option 60.

show reload-type

Displays the reload type currently configured on the system.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `show reload-type`

Defaults none

Command Modes EXEC Privilege

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the MXL switch. Updated the parameters for S4810, S4820T, and Z9000. |
| 9.1.0.0 | Updated the parameters for S4810 and Z9000. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.11.4 | Introduced on the Z9000. |
| 8.3.10.0 | Introduced on the S4810. |

Usage Information To check the currently configured Auto-Configuration mode (BMP or Normal reload) on a switch running BMP, use the `show reload-type` command.

To display the current reload mode for BMP, use the `show bootvar` or `show system brief` commands. The `show bootvar` command includes the path of the Dell Networking OS image file retrieved from a DHCP server when BMP is running, but not after you exit BMP.

Example

```
Dell#show reload
reload-type           : bmp [Next boot : bmp]
                     : enable
config-scr-download  : enable
dhcp-timeout         : 1
vendor-class-identifier : Device 609
retry-count          : 3
Dell#
```

Related Commands

[reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.

stop bmp

To prevent an infinite loop, stop the switch from reloading in BMP mode.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax

```
stop bmp
```



NOTE: This command replaces the `stop jumpstart` command.

Defaults

none

Command Modes

EXEC

Command History

This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. Replaces the <code>stop jumpstart</code> command. |
| 9.2(0.0) | Introduced on the Z9000, S4810, S4820T, and MXL switch. Replaces the <code>stop jumpstart</code> command. |

| Version | Description |
|----------|--|
| 9.1(0.0) | Introduced on the Z9000 and S4810. Replaces the <code>stop jumpstart</code> command. |

Usage Information If the switch enters a loop while reloading in BMP mode, use the `stop bmp` command on a switch running BMP. A loop occurs when the switch is continuously trying to contact a DHCP server and a DHCP server is not found. The `stop bmp` command stops the switch from connecting to the DHCP server. After you use the `stop bmp` command, the next default reload type is a normal reload as indicated in the `show reload-type` or `show system brief` commands.

The `stop bmp` command behaves differently in different circumstances:

- While the Dell Networking OS image upgrade is in progress, it aborts the BMP process after the Dell Networking OS image upgrades.
- When applying configurations from the file, it aborts the BMP process after all the configurations are applied in the system.
- When running the pre-configuration or post-configuration script, it stops the script and aborts the BMP process immediately.
- When downloading the configuration or script file, it aborts the BMP process after the download and does not apply the configuration or run the script.

When you enter GLOBAL CONFIGURATION mode during the BMP process, to avoid configuration conflicts between you and the BMP process, warning / error messages display.

Related Commands [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.

Vendor class identifier

Allows you to configure the optional vendor class identifier for DHCP option 60.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, IOA, and MXL

Syntax `vendor-class-identifier description`
 To reset the vendor-class-identifier option, use the `no vendor-class-identifier description` command.

Parameters

| | |
|---|---|
| vendor-class-identifier <i>description</i> | Enter the keyword <code>vendor-class-identifier</code> , then a brief description for the user-defined field in Option 60. The maximum is 64 characters long. The user string is appended with Type, Hardware, Serial Number, Service Tag and OS Version. |
|---|---|

Defaults none

Command Modes RELOAD-TYPE CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL. |

Related Commands

- [reload-type](#) — allows you to configure the reload-type options in reload-type sub command mode.
- [show reload-type](#) — displays the current Reload mode (BMP mode or Normal mode).

verify

Validates the software image on the flash drive after the image is transferred to the system but before the image is installed.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9000, Z9500, IOA, and MXL

Syntax `verify {md5 | sha256} [flash://] img-file [hash-value]`

Parameters

| | |
|-------------------------|---|
| md5 | MD5 message-digest algorithm. |
| sha256 | SHA256 Secure hash algorithm. |
| [flash://] flash | (Optional) Specifies the flash drive. The default is to use the flash drive. Enter the filename of the image. |
| img-file | Enter the name of the Dell Networking software image file to validate. |
| hash-value | (Optional) Specify the relevant hash published on the Dell support page. |

Defaults none

Command Modes EXEC


Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the IOA and S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.0) | Introduced on the S4810, S4820T, S6000, Z9000, and Z9500. |

Usage Information You can enter the `verify` command in the following ways:

- `verify md5 flash://img-file`
- `verify md5 flash://img-file<hash-value>`
- `verify md256 flash://img-file`
- `verify md256 flash://img-file<hash-value>`

 **NOTE:** The `hash type md5` option is available only when you disable FIPS mode; the `hash type md5` and `sha256` options are available when you enable FIPS mode.

Example

Output example without entering the Hash Value for verification using SHA256:

```
Dell# verify sha256 flash://Dell-SE-9.5.0.0.bin
SHA256 hash for Dell-SE-9.5.0.0.bin:
e6328c06faf814e6899ceead219afbf9360e986d692988023b749e6b2093e933
```

Output example of entering the Hash Value for verification using SHA256:

```
Dell# verify sha256 flash://Dell-SE-9.5.0.0.bin
e6328c06faf814e6899ceead219afbf9360e986d692988023b749e6b2093e933
SHA256 hash VERIFIED for Dell-SE-9.5.0.0.bin
```

Open Management Infrastructure

Dell Networking OS supports the Open Management Infrastructure (OMI) framework. OMI is supported on the S4810, S4820T, S3048-ON, S6000, and Z9500 platforms.

Overview

OMI is an open source package which manages infrastructure components such as servers, storage, and networking. OMI supports the distributed management task force (DMTF) common information model (CIM) / web-based enterprise management (WBEM) standards. To manage the configurations received through OMI, the Dell Networking OS uses the representational state transfer (REST) interface. OMI also uses the Python and Ruby libraries of the SmartScripts package.

The desired state configuration (DSC) is a new management platform in Windows PowerShell. It deploys and manages configuration data for software services and manages the environment in which these services run. The switches support the switch view class schema to declaratively configure Ethernet switches using the DSC.

The managed object format (.MOF) corresponds to the windows managed object files. You can configure the definition of classes and instances in the MOF files. The management station sends the OMI configuration to the switches in MOF file format.

For additional information, refer <http://blogs.technet.com/b/privatecloud/archive/2014/05/19/powershell-dsc-for-linux-step-by-step.aspx>.

Installing the OMI Agent Package

To install the OMI agent package, follow these steps.

Smart Scripting is a prerequisite for using OMI. You must first install the Smart Scripting package before you can run the OMI package (For detailed information about SmartScripts, refer to [Smart Scripting](#)).

1. Go to the Dell Networking web portal at <https://www.force10networks.com/CSPortal20/Main/SupportMain.aspx>. Click **Login**, enter your user ID and password, and click the **Login** button.
2. On the Customer Support page, click the **Software Center** tab.
3. In the left-hand column, click **Automation Software**.
4. At the bottom of the Terms and Conditions page, click **I agree**.
5. On the Automation Software page, under **Software**, select the file for the switch from the following list:
 - OMI-P-9.8.0.0.tar.gz for S4810 and S4820T.
 - OMI-I-9.8.0.0.tar.gz for S6000 and Z9500.

Prerequisites

Before using OMI on the Dell Networking OS switch, configure the following:

1. Create a user, with the user privilege **15**, where this user can configure the box from the management station through OMI.
2. Install the smartsript package. For detailed information, please refer to the **Installing SmartScripting** section of the [Smart Scripting](#) chapter.
3. Install the OMI package.

EXEC Privilege mode

```
package install <ip-address>/OMI-I-9.8.0.0.tar.gz
```

Configuring the OMI

The OMI package supports the following configurations:

1. Enable the REST API.

CONFIGURATION mode

```
rest-server http
```

2. Configure script path /usr/pkg/omi108/bin/

All OMI configurations can now be just `script execute omi-server-config args start`

3. Start the OMI server.


CONFIGURATION mode

```
script execute /usr/pkg/omi108/bin/omi-server-config args start
```

4. Stop the OMI server.

CONFIGURATION mode

```
script execute /usr/pkg/omi108/bin/omi-server-config args stop
```

 **NOTE:** The OMI server supports both secured (port 5986) and unsecured (port 5985) connections.

Other Scripts


To view the OMI agent status, use the `omi-server-show` script.

```
script execute omi-server-show args "arguments" [{top | bottom} number of lines]
```

The `args` options are:

- server — show omi-server log
- provider — show dell-provider log
- agent — show omi-agent log
- client — show omi-client log
- dsc — show dsc log

- top — show the status from top
- bottom — show the status from bottom
- number of lines — specify the number of lines

 **NOTE:** The script path is `/usr/pkg/omi108/bin`.

Getting Started with OMI

This section describes how you start configuring the OMI. To configure the OMI, follow these steps:

1. Get the chassis ready with the packages installed.
 - a. Copy the MOF file tar to flash `asflash:/dscmof.tar.gz`
EXEC mode


```
script execute omi-server-config args "get-mof"
```
 - b. Copy the `dscmof.tar.gz` to the windows server.
2. Configure `rest server http`.
3. To communicate from Windows server, configure the user with privilege level **15**. The default user is **admin**.
4. On the Windows server, create folders under the following paths as DSCResources:
 - `%SystemRoot%\system32\WindowsPowerShell\v1.0\Modules\`
 - `%SystemRoot%\Program Files\WindowsPowerShell\`
5. Create folders for each of the view in the **tar** file and copy the individual **schema.mof**.
For example, copy `MSFT_NetworkDevice_EthernetPortView.schema.mof` to `MSFT_NetworkDevice_EthernetPortView` folder.
6. In the Windows PowerShell environment, create the configuration block.
Following example uses the name `MSFT_NetworkDevice_EthernetPortViewConfig`.

```
Configuration MSFT_NetworkDevice_EthernetPortViewConfig
{
    Import-DscResource -Name MSFT_NetworkDevice_EthernetPortView

    Node ("<ip-address or name of the node>")
    {
        MSFT_NetworkDevice_EthernetPortView MyPort
        {
            PortId = "tengig-0-1"
            EnabledState = "Enabled"
        }
    }
}
```

- `PortId` identifies the interface. This example configures `tengigabitethernet 0/1`.
 - `EnabledState` identifies the status of the port. In this example, "Enabled" brings up the port.
7. In Powershell, create the Ethernet MOF using configuration block.
`MSFT_NetworkDevice_EthernetPortViewConfig -OutputPath c:\EthernetPortViewMof`
This configuration creates the MOF in the `C:\EthernetPortViewMof` directory.
 8. Connect to OMI.
`$cred=Get-Credential -UserName:"admin" -Message:"Admin User?"`
 9. Enter the credential of the user **admin**.

To create a CIM-Session, use the following scripts:

- `$opt = New-CimSessionOption -UseSsl:$true -SkipCACheck:$true -SkipCNCheck:$true -SkipRevocationCheck:$true`
- `$box1 =New-CimSession -Credential:$cred -ComputerName:chassis-1 -Port:5986 -Authentication:basic -SessionOption:$opt`

chassis-1 is the IP address of the chassis.

10. Apply the configuration using the following script:

```
PS C:\windows\system32> Start-DscConfiguration -CimSession:$box1 -Path:"C:\EthernetPortViewMof" -Verbose -Wait
```

- CimSession is the **box1** created in step 9.
- Path is the path of the **MOF** created in step 7.

11. Verify in the chassis for interface tengigabitethernet 0/1.

- `show running-config tengig 0/1`
- `show interface tengig 0/1`

Examples

This section shows various OMI configurations.

Access Control List (ACL) Configuration

```
instance of MSFT_NetworkDevice_AclEntry as $MSFT_NetworkDevice_AclEntry1ref
{
Source = "Any";
SourceMask = "0";
Action = 0;
Destination = "Any";
DestinationMask = "0";
Sequence = 70;
AclEntryProtocol = 5;
};

instance of MSFT_NetworkDevice_AclView as $MSFT_NetworkDevice_AclView1ref
{
ResourceID = "[MSFT_NetworkDevice_AclView]MyAclconfig";
ElementName = "testprefix3";
AclProtocol = 0;
Ensure = 0;
AclType = 1;
IsPolicyMapMatchCondition = True;

SourceInfo = "C:\\ConfigFiles\
\MSFT_NetworkDevice_AclView_config.ps1::6::9::MSFT_NetworkDevice_AclView";
AclList = {
    $MSFT_NetworkDevice_AclEntry1ref
};
ModuleName = "DSCModules";
ModuleVersion = "1.0";
};

instance of OMI_ConfigurationDocument
{
```

```

Version="1.0.0";
Author="Administrator";
GenerationDate="12/31/2014 00:11:54";
GenerationHost="WIN-FMOQ86BVN3F";
};

```

Ethernet Configuration

```

Configuration MSFT_NetworkDevice_EthernetPortViewConfig
{
    Import-DscResource -Name MSFT_NetworkDevice_EthernetPortView

    Node ("172.168.1.2")
    {
        MSFT_NetworkDevice_EthernetPortView MyPort
        {
            PortId = "tengig-0-11"
            Description = "Interface connecting to S4810"
        }
    }
}
MSFT_NetworkDevice_EthernetPortViewConfig -OutputPath c:\EthernetPortViewMof

```

The mof for the node "172.168.1.2" will get stored in the path C:\EthernetPortViewMof.

```

PS C:\> Start-DscConfiguration -CimSession $dellBox -Path C:\
\EthernetPortViewMof -Wait -Force
PS C:\>

```

Border Gateway Protocol (BGP) Configuration

For a global configuration with virtual routing and forwarding (VRF) vrf-1 (use default for default-vrf):

```

instance of MSFT_NetworkDevice_IpAddress as $MSFT_NetworkDevice_IpAddress1ref
{
    IpAddressFamily = 1;
    IpAddress = "12.1.5.11";
    NetworkPrefix = 32;
};

instance of MSFT_NetworkDevice_IpAddress as $MSFT_NetworkDevice_IpAddress2ref
{
    IpAddressFamily = 1;
    IpAddress = "10.10.10.20";
    NetworkPrefix = 32;
};

instance of MSFT_NetworkDevice_IpAddress as $MSFT_NetworkDevice_IpAddress3ref
{
    IpAddressFamily = 1;
    IpAddress = "10.10.10.15";
    NetworkPrefix = 32;
};

instance of MSFT_NetworkDevice_BgpConfigurationView as
$MSFT_NetworkDevice_BgpConfigurationView1ref
{
    EnabledState = True;
};

```

```

Id = "1-1-vrf-1";
ExtAsNumber = 1;
Ensure = 0;
ResourceID = "[MSFT_NetworkDevice_BgpConfigurationView]MyBgpConfiguration";
RouterId = $MSFT_NetworkDevice_IpAddress1ref;
SourceInfo = "C:\\ConfigFiles\\
\\MSFT_NetworkDevice_BgpConfigurationViewConfig.ps1::8::9::MSFT_NetworkDevice_Bgp
ConfigurationView";
ModuleVersion = "1.0";
VrfContext = "vrf-1";
ModuleName = "DSCModules";
RedistributeConnected = True;
RedistributeStatic = True;
RedistributeConnectedRouteMap = "abc";
RedistributeStaticRouteMap = "abc";
/* MaxEBgpPaths = 7;
MaxIBgpPaths = 7;*/
BgpNetworks = { $MSFT_NetworkDevice_IpAddress2ref,
$MSFT_NetworkDevice_IpAddress3ref };
AddressFamily = 1;
};

```

<snip>

The ID, the key, is of the form,
Id = "1-1-vrf-1";

The key is of the form
AddressFamily-ExtAsNumber-VRFName.

For an ASN 100 on default-VRF (IPv4) it will be
1-100-default
For IPv6 context, it will be
2-100-default (corresponds to
Router bgp 100
Address-family ipv6 unicast)
</snip>

BGP Peer View Configuration

```

instance of MSFT_NetworkDevice_IpAddress as $MSFT_NetworkDevice_IpAddress1ref
{
  IpAddressFamily = 1;
  IpAddress = "1.2.3.4";
  NetworkPrefix = 0;
};

instance of MSFT_Credential as $MSFT_Credential1ref
{
  Password = "force10";
  UserName = "Any\\Any";
};

instance of MSFT_NetworkDevice_BgpPeerView as
$MSFT_NetworkDevice_BgpPeerView1ref
{
  Id = "1-vrf-1-1.2.3.4";
  ResourceID = "[MSFT_NetworkDevice_BgpPeerView]MyBgpPeer";
  SourceInfo = "C:\\ConfigFiles\\
\\MSFT_NetworkDevice_BgpPeerViewConfig.ps1::21::9::MSFT_NetworkDevice_BgpPeerView

```

```

";
/* Name = "NEIGHBOR";*/
RemoteRouterId = $MSFT_NetworkDevice_IpAddress1ref;
/*PeerCredential = $MSFT_Credential1ref;*/
//MultiHopTimeToLive = 200;
/* IsEnabled = True;*/
VrfContext = "vrf-1";
ExtAsNumber = 3;
Ensure = 0;
KeyEncryptionMethod = 3;
ModuleVersion = "1.0";
ModuleName = "DSCModules";
AddressFamily = 1;
};

```

<snip>

The ID here is of the format
AddressFamily-VRFName-NeighborIp

For a default VRF and neighbor 1.2.3.4, it will be
1-default-1.2.3.4

</snip>

Sample BGP peer view configuration for enumerating an instance:

```
Get-CimInstance -ClassName MSFT_NetworkDevice_BgpPeerView -CimSession $cb -
Namespace root/Microsoft/DesiredStateConfiguration
```

```

Id                : 1-default-1.2.3.4
Name              :
PeerCredential    : MSFT_Credential
AddressFamily     : 1
VrfContext       : default
IsEnabled        : False
ActivateIPv6     : True
AllowAsIn        :
MultiHopTimeToLive :
RouteMapIn       : rmap1
RouteMapOut      : rmap2
Ensure           : 0
KeyEncryptionMethod : 3
RemoteRouterId   : MSFT_NetworkDevice_IpAddress
UpdateSourceInterface :
UpdateSourceInterfaceType :
ExtAsNumber      : 3
PSComputerName   : 10.16.129.106

Id                : 1-vrf1-1.2.3.4
Name              :
PeerCredential    : MSFT_Credential
AddressFamily     : 1
VrfContext       : vrf1
IsEnabled        : False
ActivateIPv6     : True
AllowAsIn        :
MultiHopTimeToLive :
RouteMapIn       :
RouteMapOut      :
Ensure           : 0
KeyEncryptionMethod : 3
RemoteRouterId   : MSFT_NetworkDevice_IpAddress

```

```

UpdateSourceInterface      :
UpdateSourceInterfaceType :
ExtAsNumber                : 3
PSComputerName             : 10.16.129.106

Id                          : 1-vrf1-2.3.4.5
Name                        :
PeerCredential              : MSFT_Credential
AddressFamily               : 1
VrfContext                  : vrf1
IsEnabled                   : False
ActivateIPv6                : True
AllowAsIn                   :
MultiHopTimeToLive         :
RouteMapIn                  :
RouteMapOut                 :
Ensure                      : 0
KeyEncryptionMethod         : 3
RemoteRouterId              : MSFT_NetworkDevice_IpAddress
UpdateSourceInterface      :
UpdateSourceInterfaceType  :
ExtAsNumber                 : 3
PSComputerName              : 10.16.129.106

Id                          : 1-vrf2-1.2.3.4
Name                        :
PeerCredential              : MSFT_Credential
AddressFamily               : 1
VrfContext                  : vrf2
IsEnabled                   : False
ActivateIPv6                : True
AllowAsIn                   :
MultiHopTimeToLive         :
RouteMapIn                  :
RouteMapOut                 :
Ensure                      : 0
KeyEncryptionMethod         : 3
RemoteRouterId              : MSFT_NetworkDevice_IpAddress
UpdateSourceInterface      :
UpdateSourceInterfaceType  :
ExtAsNumber                 : 3
PSComputerName              : 10.16.129.106

Id                          : 1-vrf-1-1.2.3.4
Name                        :
PeerCredential              : MSFT_Credential
AddressFamily               : 1
VrfContext                  : vrf-1
IsEnabled                   : False
ActivateIPv6                : True
AllowAsIn                   :
MultiHopTimeToLive         :
RouteMapIn                  : rmap1
RouteMapOut                 : rmap2
Ensure                      : 0
KeyEncryptionMethod         : 3
RemoteRouterId              : MSFT_NetworkDevice_IpAddress
UpdateSourceInterface      :
UpdateSourceInterfaceType  :
ExtAsNumber                 : 3
PSComputerName              : 10.16.129.106

```

```

Id          : 2-vrf1-1.2.3.4
Name        :
PeerCredential : MSFT_Credential
AddressFamily : 2
VrfContext  : vrf1
IsEnabled   :
ActivateIPv6 : True
AllowAsIn   :
MultiHopTimeToLive :
RouteMapIn  :
RouteMapOut :
Ensure      : 0
KeyEncryptionMethod : 3
RemoteRouterId : MSFT_NetworkDevice_IpAddress
UpdateSourceInterface :
UpdateSourceInterfaceType :
ExtAsNumber :
PSComputerName : 10.16.129.106

Id          : 2-vrf1-2.3.4.5
Name        :
PeerCredential : MSFT_Credential
AddressFamily : 2
VrfContext  : vrf1
IsEnabled   :
ActivateIPv6 : True
AllowAsIn   :
MultiHopTimeToLive :
RouteMapIn  :
RouteMapOut :
Ensure      : 0
KeyEncryptionMethod : 3
RemoteRouterId : MSFT_NetworkDevice_IpAddress
UpdateSourceInterface :
UpdateSourceInterfaceType :
ExtAsNumber :
PSComputerName : 10.16.129.106

```

The above configuration corresponds to the following running-config:

```

router bgp 1
  bgp router-id 12.1.5.11
  network 10.10.10.15/32
  network 10.10.10.20/32
  neighbor 1.2.3.4 remote-as 3
  neighbor 1.2.3.4 route-map rmap1 in
  neighbor 1.2.3.4 route-map rmap2 out
  neighbor 1.2.3.4 shutdown
  !
  address-family ipv4 vrf vrf1
  network 10.10.10.15/32
  network 10.10.10.20/32
  maximum-paths ebgp 3
  redistribute static route-map rmap3
  redistribute connected route-map rmap1
  neighbor 1.2.3.4 remote-as 3
  neighbor 1.2.3.4 description descr1
  neighbor 1.2.3.4 allowas-in 3
  neighbor 1.2.3.4 shutdown
  neighbor 2.3.4.5 remote-as 3
  neighbor 2.3.4.5 shutdown
  neighbor 1::1 remote-as 3
  neighbor 1::1 shutdown

```

```
exit-address-family
!
address-family ipv4 vrf vrf2
neighbor 1.2.3.4 remote-as 3
neighbor 1.2.3.4 shutdown
exit-address-family
!
address-family ipv4 vrf vrf-1
network 10.10.10.15/32
network 10.10.10.20/32
redistribute static route-map rmap1
redistribute connected route-map rmap2
neighbor 1.2.3.4 remote-as 3
neighbor 1.2.3.4 route-map rmap1 in
neighbor 1.2.3.4 route-map rmap2 out
neighbor 1.2.3.4 shutdown
exit-address-family
!
!
address-family ipv6 unicast vrf vrf1
redistribute static route-map rmap3
redistribute connected route-map rmap1
neighbor 2.3.4.5 activate
exit-address-family
```

Puppet

Dell Networking OS supports the Puppet automation framework. Puppet Agent is supported on the S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, and Z9500 platforms.

Overview

Puppet is a configuration management utility, that defines the state of system, then automatically enforces the correct state. Puppet Agent is a daemon that runs on all the client servers. Using Puppet, you can automate the repetitive tasks, quickly deploy critical applications, and proactively manage changes, scaling from 10s of servers to 1000s, on-premise or in the cloud. The Puppet Standalone/Agent applies configurations on the Dell Networking OS according to the puppet manifest file declarations. The Puppet 1.0 provides support for Ruby, Puppet, and the user interface to both the Puppet Agent and Puppet Standalone.

Installing Puppet

To install the Puppet package, follow these steps.

Smart Scripting is a prerequisite for using Puppet. You must first install the Smart Scripting package before you can run the Puppet application (refer to [Puppet Agent Configuration Scripts](#)).

1. On a PC or other network device, go to the Dell Networking web portal at <https://www.force10networks.com/CSPortal20/Main/SupportMain.aspx>. Click **Login**, enter your user ID and password, and click the **Login** button.
2. On the Customer Support page, click the **Software Center** tab.
3. In the left-hand column, click **Automation Software**.
4. At the bottom of the Terms and Conditions page, click **I agree**.
5. On the Automation Software page, under **Software**, select the file for the switch from the following list:
 - PUPPET-P-9.8.0.0.tar.gz for S4810, S4820T, and S5000.
 - PUPPET-I-9.8.0.0.tar.gz for S6000, S6000-ON, and Z9500.

Prerequisites

Before using the Puppet Agent on the Dell Networking OS switch, configure the following:

1. Create the puppet user, with the user privilege **15**.
2. Install the smartscript package.


EXEC Privilege mode

```
Dell#package install tftp://10.16.127.35/SMARTSCRIPTS-I-9.6.9.5.tar.gz
```

3. Install the puppet package.

EXEC Privilege mode

```
package install tftp://10.16.127.35/PUPPET-I-9.8.0.0.tar.gz
```

 **NOTE:** You must keep the clock of the chassis in sync with the Puppet master. The synchronization is done by the Network Transfer Protocol (NTP) server.

Configuring the Puppet Agent

To configure the Puppet Agent, use the following steps.

1. Enable the REST API.

CONFIGURATION mode

```
rest-server http
```

2. Configure `script path /usr/pkg/puppet/bin`.

All puppet configurations can now be just `script execute puppet-config args username guestuser`.

3. Configure the puppet agent.

EXEC Privilege mode

```
script execute puppet-config args username guestuser
```

4. Run the puppet agent once to poll the master. This creates the SSL certificate on the agent and sends the request to the master.

EXEC Privilege mode

```
script execute puppet-agent start now args poll-master username guestuser
```

5. Sign the certificate on master.

```
puppet cert sign "Agent Cert Name"
```

6. Run the puppet agent again with the signed certificate.

EXEC Privilege mode

```
script execute puppet-agent start now args poll-master username guestuser
```

7. Start the puppet agent which runs in the background.

EXEC Privilege mode

```
script execute puppet-agentd username puppet bg
```

8. Monitor the puppet agent.

EXEC Privilege mode

```
script execute puppet-show args "status"
```

Puppet Agent Configuration Scripts

This section describes about the various Puppet Agent configuration scripts:

puppet-config

To configure or make changes to the agent's parameters, use the `puppet-config` script.

```
puppet-config [-d] option=value
```

Following are the mandatory puppet configuration options:

```
agentrestipaddr    : IP address to be used for rest apis.  
                   : The default IP address is 127.0.0.1.  
agentrestpasswd   : Password of the user given in agentrestuser.  
agentrestport     : Port number to be used for rest apis.  
                   : The default port is 8008.  
agentrestproto    : Protocol (http or https) to be used for rest apis.  
                   : The default protocol is HTTP.  
agentrestuser     : Username to be used for the rest apis.  
                   : The default username is the Puppet user.  
agentsyncinterval : Interval in minutes at which the Dell puppet agent  
                   : daemon should poll the master.  
                   : The default time interval is 60.  
certname         : Certname of the puppet agent running in the device.  
server          : Hostname/IP address of the puppet master.
```

puppet-show

To view the agent-daemon status, use the `puppet-show` script.

```
puppet-show config | log | status
```

Following are the show puppet options:

```
config : Show the Dell puppet agent config  
log    : Show the logs of Dell puppet agent daemon  
status : Show the status of Dell puppet agent daemon
```

puppet-agent

To perform the agent specific operations, use the `puppet-agent` script.

```
puppet-agent clear-ssl | pause | poll-master | resume
```

Following are the puppet agent options:

```
clear-ssl : Clear puppet agent ssl certificates  
pause    : Request Dell puppet agent daemon to pause from polling  
          : master  
poll-master : Run puppet agent now to get config from master  
resume    : Request Dell puppet agent daemon to resume polling master  
          : if it is paused
```

puppet-apply

To apply any puppet manifest file, use the `puppet-apply` script.

```
puppet-apply -f file | -c command
```

Following are the puppet apply script options:

```
-h, --help                show this help message and exit
-f FILENAME, --file=FILENAME Apply the given puppet manifest file
-c COMMAND, --command=COMMAND Apply the given puppet command
```

Building the Manifest File

Puppet consists of a custom declarative language to describe the system configuration, which can be either applied directly on the system or compiled into a catalog and distributed to the destination system via the client-server paradigm (using REST API). The agent uses the system-specific providers to enforce the resource specified in the manifests.

The Puppet Agent network devices are managed using the following configurations:

- Hostname Configuration
- Layer 2 interface configuration
- Link aggregation group (LAG) configuration
- Physical interface configuration
- Virtual local area network (VLAN) configuration

Hostname Configuration

Apply the hostname configurations using both the Puppet Standalone and Puppet Agent. You can read and write the hostname configuration by using the `netdev_device`.

Example for configuring the Hostname.

```
node dell_s4810_node1
{
  netdev_device { hostname: }
}
```

Layer 2 Interface Configuration

Apply the Layer 2 interface configurations using both the Puppet Standalone and Puppet Agent. You can read and write the Layer 2 interface configurations by using the `netdev_l2_device` type.

```
node dell_s4810_node1
{
  netdev_device { hostname: }
  netdev_l2_interface
  { "name":
    ensure => (present | absent),
    active => (true | false),
    description => "interface-description",
    tagged_vlans => (vlan | [vlan1, vlan2, vlan3, ...]),
    untagged_vlan => vlan,
    vlan_tagging => (enable | disable)
  }
```

```
}  
}
```

LAG Configuration

Apply the LAG configurations using both the Puppet Standalone and Puppet Agent. You can read and write the LAG configurations by using the `netdev_lagtype`.

```
node dell_s4810_node1  
{  
  netdev_device { hostname: }  
  netdev_lag  
  { "name":  
    ensure => (present | absent),  
    active => (true | false),  
    links => ('interface-name' | ['interface-name1', 'interface-  
name2' ...]),  
    lacp => (active | disabled | passive),  
    minimum_links => minimum  
  }  
}
```

Physical Interface Configuration

Apply the physical interface configurations using both the Puppet Standalone and Puppet Agent. You can read and write the physical interface configuration by using the `netdev_interface` type.

```
node dell_s4810_node1  
{  
  netdev_device { hostname: }  
  netdev_interface  
  { "name":  
    ensure => (present | absent),  
    active => (true | false),  
    admin => (up | down),  
    description => "interface-description",  
    speed => speed,  
    duplex => (auto | full | half),  
    mtu => mtu  
  }  
}
```

VLAN Configuration

Apply the VLAN configurations using both the Puppet Standalone and Puppet Agent. You can read and write the VLAN configurations by using the `netdev_vlan` type.

```
node dell_s4810_node1  
{  
  netdev_device { hostname: }  
  netdev_vlan  
  { "name":  
    ensure => (present | absent),  
    active => (true | false),  
    vlan_id => id,  
    description => "vlan-description",  
  }  
}
```

Using the Puppet Factor

Dell Networking OS supports the standalone tool called the Puppet Factor. This tool is based on Ruby and provides complete system information about the agent node to the puppet master. To apply the configurations using puppet, you must install the Puppet Factor tools on all the nodes.

Following are the facts pair in the manifest files.

```
osdistribution="Dell Real Time Operating System"
hostname={Chassis Hostname}
osdistribution_version={Dell OS version}
platformtype=[S4810|S4820|Z9000]
netdev_interfaces={list of interfaces}
netdev_vlans={list of vlans configured}
domain=dell.com
osfamily=NetBSD
architecture="x86"|"ppc"
puppetversion={Puppet Version}
rubyversion={Ruby Version}
uptime={System uptime}
```

Setting up the Puppet Master

Puppet uses the client-server model. The client-server regularly connects to the master server to synchronize their configurations and to report the changes, if any, back to the central nodes. When you install or upgrade Puppet, you must upgrade the Puppet Master component first. To install and set up the Puppet Master, follow these steps:

1. Install the Ruby script.
`sudo yum install ruby`
2. Add the puppet repository.
`sudo rpm -ivh http://yum.puppetlabs.com/puppetlabs-release-el-7.noarch.rpm`
3. Install the puppet master.
`sudo yum install puppet-server`
4. Go to the Dell Networking web portal at <https://www.force10networks.com/CSPortal20/Main/SupportMain.aspx>. Download the dell-dnos puppet module and install it in the puppet master.
5. Install the rest_client ruby gem.
`gem install rest-client`
6. Open port **8140** in iptables.
`iptables -I INPUT 2 -p tcp -- dport 8140 -j ACCEPT`
7. Run the puppet master.
`puppet master`
8. Kill the process and restart the puppet master.
`ps aux | grep puppet`

Smart Scripting

Smart Scripting is supported on the S4810, S4820T, S3048-ON, S5000, S6000, Z9500, and MXL switch platforms.

Smart Scripting allows you to add functionality to Dell Networking switches without updating the Dell Networking OS release. Smart Scripting is available as a separate installable package that supports SQLite DB and NET SNMP applications and TCL, Expect, Perl, Python, Ruby, and Unix scripting languages.

The Smart Scripting package supports the smart utility (SmartUtils) application programming interfaces (APIs) providing you an easier way to invoke switch operations by creating and running Tcl, Expect, Perl, Python, Ruby, and ZSH scripts on the Dell Networking OS. API library files describe the functions supported in Perl, Python, and Ruby scripts.

A separate package is extended with HTTP and HTTPS daemons based on CGI scripts and a Web-based graphical user interface. For more information about this package, refer to [Web Server with HTTP Support](#). For information about HTTP requests the REST API supports, refer to the [REST API CLI](#).

Overview

To manage and interact with Dell Networking switches in the network, you can use Smart Scripting to create custom Expect, Perl, Python, Ruby, Tcl, UNIX, and ZSH shell scripts.

Smart Scripting provides support for:

- Modules that implement requested Python features, such as AMQP (message queuing), XML-RPC (arbitrary data exchange), and Twisted (event-driven networking engine).

With Smart Scripting, you don't need to learn proprietary scripting languages; allowing for faster development and deployment of custom scripts.

Smart Scripting also offers solutions in a UNIX environment useful to cloud administrators who are familiar with working in a UNIX shell. Smart Script support in a UNIX environment allows you to invoke standard UNIX utilities such as `netstat`, `tcpdump`, `ls`, `chmod`, `chown`, and so on.

Smart Scripting includes a convenient set of API function libraries that you can refer to when you create Expect, Perl, Python, Ruby, Tcl, UNIX, and ZSH shell scripts. API libraries provide a representation of CLI functions to retrieve data from the Dell Networking OS and change configuration parameters on Dell Networking switches. Script writers include API function calls made directly on the Dell Networking OS CLI in Expect, Perl, Python, Ruby, Tcl, UNIX, and ZSH shell scripts.

For example, the API functions a script uses includes setting up a telnet session, gathering data on the switch, sending information to the CLI, and closing telnet sessions. By using simple function calls, you do not have to include the parsing code required for telnet sessions and retrieving configuration information.

Smart Scripting supports running a script either from the Dell Networking OS CLI or directly from a UNIX shell. You can run scripts periodically, based on events, at boot up or after the switch is initialized. In addition, you can run scripts in the foreground or background and display the progress of a script. If required, you can cancel, stop, or resume scripts.

Smart Scripting allows you to automate common management and maintenance tasks, such as:

- Building visibility and/or discovery programs.
- Creating custom logging.
- Reporting configuration information.
- Reporting switch memory usage, configured virtual local area networks (VLANs), and other operation and configuration parameters.
- Creating custom APIs for external applications to access the switch.
- Automating custom provisioning of network devices to support server virtualization.

For example, you can automate any of the following tasks:

- Monitor the configuration of switch ports to verify that no change occurs and generate an alarm if a configuration change is detected as part of a cloud-computing deployment.
- Stage CLI command requests received from a customer. If a link flaps, the command completion status is held in the script so you can see when the management plane reconnects.
- Generate time-based reports to receive updates on network status on a periodic basis.
- Query an external, configuration management database on a remote server to retrieve information on port operation and reconfigure switch ports based on the data received.
- Apply additional time-based access-control lists (ACLs) to limit after hours access.
- Monitor network requests; for example, *find a specified MAC address or generate a health-check heartbeat*.
- Create a simple menu of options that a non-network administrator can use to create requests to send to the network.

Smart Scripting consolidates management data inside a switch and sends it to management consoles, databases, or applications — reducing polling and network traffic. For example, you can use a script as part of a cloud-computing deployment to detect when the network has changed, query a database server for configuration management database (CMDB) information, and ultimately apply network changes based on the data.

Download the Smart Scripting Package

Download the Smart Scripts package from the Dell Networking OS website.

The Smart Script filenames are:

- SMARTSCRIPTS-P-9.8.0.0.tar.gz for S4810, S4820T, and S5000
- SMARTSCRIPTS-I-9.8.0.0.tar.gz for S6000, Z9500
- SMARTSCRIPTS-M-9.8.0.0.tar.gz for MXL Switch

To download the SmartScript package to a dedicated location, use one of the following methods: TFTP, FTP, Flash, or NFSMount.

The Smart Scripting package includes the following files and functionality:


- Dell Smartsript Utilities supports Perl, Python, and Ruby APIs.

- SQLite database and Perl, Python, Ruby, TCL interface to SQLite.
- NetSNMP client and Perl, Python, Ruby interface to the client.
- Perl interpreter and associated files.
- Python interpreter and associated files.
- Ruby interpreter and associated files.
- Expanded set of UNIX utilities.
- Web-Server with HTTP support (refer to [Web Server with HTTP Support](#)).

Installing Smart Scripting

After you download the Smart Scripts package, install the file from local flash memory on a switch or from an external drive on a network server.

Because the installation takes time, installing Smart Scripting performs in the background. When the download completes, a message displays on the console. The package installation updates the running configuration file.

 **CAUTION: You can modify (for example, edit or rename) the files Smart Scripting downloads only in the directory in which you install the package. Never modify the files in other system directories.**

To install the Smart Scripting package, download it from the Dell Networking web portal.

1. On a PC or other network device, go to the Dell Networking web portal at <https://www.force10networks.com/CSPortal20/Main/SupportMain.aspx>. Click **Login**, enter your user ID and password, and click the **Login** button.
2. On the Customer Support page, click the **Software Center** tab.
3. In the left-hand column, click **Automation Software**.
4. At the bottom of the Terms and Conditions page, click **I agree**.
5. On the Automation Software page, under Software, select the file for the switch from the following list:
 - SMARTSCRIPTS-P-9.8.0.0.tar.gz for S4810, S4820T, and S5000
 - SMARTSCRIPTS-I-9.8.0.0.tar.gz for S6000, Z9500
 - SMARTSCRIPTS-M-9.8.0.0.tar.gz for the MXL switch
6. In the dialog box, select the path for the local flash on the switch or a directory path on a network server where you want to download the SMARTSCRIPTS package.
7. To install the Smart Scripting package, when the download completes, use the `package install` command from the Dell Networking OS CLI.
8. Install the Smart Scripting package from local flash memory or a network server to a dedicated location on your server for script storage.

EXEC Privilege mode

```
package install {flash://filename | ftp://userid:password@host-ipaddress/
dir-path | tftp:// host-ipaddress/dir-path | nfsmount://filepath}
```

- `flash://filename` installs the Smart Scripting file stored in flash memory on the switch.
- `ftp://userid:password@host-ipaddress/filepath` logs in and installs Smart Scripting from a file stored on an FTP server.
- `tftp://host-ipaddress/filepath` installs Smart Scripting from a file stored on a TFTP server.

- `nfsmount://filepath` copies from the nfs mount file system.

To remove an installed Open Automation package, such as Smart Scripting, use the `package uninstall` command.

To follow the progress of a package installation (or removal), use the `show packages` command.

Displaying Installed Packages

To view the Open Automation packages currently installed on a switch, including version numbers and content, use the following command.

- View package information.


EXEC Privilege mode

```
show packages
```

```
show packages system
```

Uninstalling Smart Scripting

Uninstalling the Smart Scripting package removes it from the internal flash memory.

 **CAUTION:** Before you uninstall the Smart Scripting package, stop all scripts that are running using the `no script script-name` command. You must also manually stop the HTTP server daemon.

- Uninstall the Smart Scripting package stored on the switch.

EXEC Privilege

```
package uninstall package-name
```

Enter the name of the Smart Scripting package exactly as it appears in the `show packages` list.

Dell Proprietary Utilities

Dell Networking OS provides the following utilities which are unique and helpful while scripting.

1. `clido`
2. `dellsh`
3. `dellsyslog`

clido


To configure any command on the device directly without handling it via the `telenet` or `ssh` sessions, use the `clido` system command.

Syntax: `clido "CLI command"`

Example

```
clido "configure terminal"
clido "interface vlan 10"
clido "no shutdown"
```

```
clido "end"
clido "show vlan"
```

 **NOTE:** `clido` maintains context upon execution in the system. To handle such cases, switch between different submodes wherever required for the needed configurations.

The **clido** utility has no pagination, is always set to "terminal length 0", and has Dell Networking OS CLI privilege 15 enabled. It works in the following modes:

- Continues mode (**clido** command) — use to retain the Dell Networking OS context.
- Reset mode (**clido -r** command)
 - use to reset the CLI command to its original context.

Sample script for `clido` in `zsh`

```
#!/usr/bin/zsh
clido -r "configure terminal"      #usage of -r is to forcefully go to exec mode
clido "interface vlan 10"
clido "no shutdown"
clido "name MailServer_Vlan"
exit 0
```

dellsh

To parse a list of commands line by line in any given text file and configure it on the device, use the `dellsh` system command.

Syntax: `dellsh <cli-file>`

Sample script for `dellsh` in Python

```
#!/usr/pkg/bin/python
import os
os.system("dellsh '/usr/pkg/ss-scripts/dellsh.txt'")
```

Dellsh can also be used as

```
Dell#script execute dellsh args /usr/pkg/ss-scripts/dellsh.txt
```

Sample `dellsh` parser file

```
#!/usr/bin/dellsh
configure terminal
interface vlan 10
no shutdown
exit
interface loopback 10
ip address 10.10.10.10/24
end
do show vlan
show running | grep loopback
```

dellsyslog

A Dell proprietary utility which helps you in generating custom SYSLOGS with the configured severity.

Syntax: `dellsyslog [-s <emergency|alert|critical|error|warning|notify|info|debug|0-7>] {syslog message}`

The severity level maps as follows:

- 0 – emergency
- 1 – alert
- 2 – critical
- 3 – error
- 4 – warning
- 5 – notify
- 6 – infor
- 7 – debug

The default severity level is **6**.

Sample script for `dellsyslog` in Python

```
#!/usr/pkg/bin/python
import sys
import os
print sys.argv[1]
os.system("dellsyslog -s %s 'The Syslog is generated by python'" %
(sys.argv[1]))
```

Sample script for `dellsyslog` in Perl

```
#!/usr/pkg/bin/perl
#system("dellsyslog" , "-s $ARGV[0] 'The Syslog is generated by Perl script'")
`dellsyslog -s $ARGV[0] "The Syslog is generated by Perl script"`
```

Dell SmartScript Utilities

Dell SmartScript utility is an easy API handler for scripting. When you install the Smart Scripting package, sample Perl, Python, and Ruby scripts install in the `/usr/pkg/scripts/sample_scripts` directory. You can also create your own customized scripts and store them anywhere on the switch, such as in a `/f10/flash_scripts` directory.

In addition, Dell SmartScript Utils uses the Perl, Python, and Ruby APIs to create scripts that invoke function calls directly in the Dell Networking OS commands.

Following are the Dell Networking OS APIs:

- `DellExec` – If you need to execute the script inside another script, you can make use of this API.
Syntax: `DellExec (scriptName, [args])`
- `DellCliConfig` – If you need to configure the list of Dell Networking OS commands, you can make use of this API.

Syntax: `DellCliConfig (CLI_list)`

where `CLI_list` can be defined as (`"interface vlan 10", "no shutdown"`)

- `DellCli` – If you need to execute a EXEC mode Dell Networking OS command, you can make use of this API

Syntax: `Dellcli ("<exec command>")`

Sample script for Perl using DellExec, DellCli, and DellConfig

```
#!/usr/pkg/bin/perl
#importing DellSSUtils
require "/usr/pkg/scripts/DellSSUtils/DellSmartUtils.pl";
#Assigning Value to the vairable which is used as an argument to the
#dell library functions
$cli = "show version";
$script = "/usr/pkg/scripts/sample_scripts/sum.py";
$args = "1 2 3";
$ExtraArgs = "10";
@list = ("interface vlan $ARGV[0]", "no shutdown", "ip address $ARGV[0].
$ARGV[0].$ARGV[0].$ARGV[0]/24", "exit");
open (MYFILE, '>>/f10/flash/dellcliop.txt');
print MYFILE DellCli($cli);
print MYFILE "\n";
print MYFILE DellExec($script, $args);
print MYFILE "\n";
print DellCliConfig(@list);
```

Sample script for Python using DellExec, DellCli, and DellConfig

```
#!/usr/pkg/bin/python
import time
import sys
sys.path.append('/usr/pkg/scripts/DellSSUtils/')
import DellSmartUtils
for input1 in sys.argv[1:]:
    print DellSmartUtils.DellCliConfig(["%s" %str("interface vlan %s" %str(input1)
) , "no shutdown"])
arg_val=""
for b in sys.argv[1:] :
    arg_val=arg_val+" %s" %(b)
print DellSmartUtils.DellExec("loopback.zsh" , arg_val)
f = DellSmartUtils.DellCli("show version")
fh = open("/f10/flash/Dellcli.txt", "w")
fh.writelines(f)
fh.close()
```

For instructions about how to run a Perl, Python, or Ruby script from the Dell Networking OS CLI, refer to [Scheduling Time / Event-based Scripts](#).

SQLite

The Dell Networking OS supports the SQLite database engine as it is self-contained, serverless, zero-configuration, and transactional.

This database performs the following tasks:

- Stores all the periodically collected data and/or statistics from the Dell Networking OS.
- Stores the Dell Networking OS configuration files.
- Stores other script logs, updated, deleted, and transferred using this database.

Dell Networking OS provides Perl, Python, Ruby, and TCLSH interfaces for SQLite. As a result, you can create custom scripts to handle all the database information and operations.

NET SNMP Client

To manage the device using the simple network management protocol (SNMP) through scripts, the Dell Networking OS supports the NetSNMP client.

You can run scripts that use SNMP to manage the data center and devices from within the Dell Networking OS. The Dell Networking OS provides Perl, Python, and Ruby interfaces for the NetSNMP client. To execute the SNMP operations, use the `tc1sh exec` command. As a part of the NetSNMP client, the following utility applications are supported:

- `snmpwalk`
- `snmpget`
- `snmpset`
- `snmpbulkget`
- `snmpbulkwalk`
- `snmpbulknext`

Limits on System Usage

To restrict CPU and memory usage, Smart Scripting establishes limits on system processes for the following attributes (regardless of the user-privilege level or scripting method).

Table 1. Limits on System Attributes

| System Attributes | Value | Description |
|---------------------------|-----------|--|
| <code>cputime</code> | unlimited | Maximum amount of time a process uses. |
| <code>filesize</code> | unlimited | Largest file size (in bytes) that can be created. |
| <code>datasize</code> | 131,072 | Maximum size (in bytes) of the data segment for a process; this value defines how far a program may extend its break with the <code>sbrk(2)</code> system call. |
| <code>stacksize</code> | 2,408 | Maximum size (in bytes) of the stack segment for a process; this value defines how far a program's stack segment can be extended. The system automatically performs stack extension. |
| <code>coredumpsize</code> | unlimited | Largest size (in bytes) of a core file that can be created. |
| <code>memory use</code> | 233,244 | Maximum size (in bytes) to which a process's resident set size may grow. This value imposes a limit on the amount of physical |

| System Attributes | Value | Description |
|-------------------|--------|--|
| | | memory a process is given; if memory is tight, the system prefers to take memory from processes that are exceeding their declared resident set size. |
| memorylocked | 77,741 | Maximum size (in bytes) which a process may lock into memory using the <code>mlock(2)</code> function. |
| maxproc | 160 | Maximum number of simultaneous processes allowed for the user ID. |
| openfiles | 64 | Maximum number of open files for this process. |

Supported UNIX Utilities

Smart Scripting supports the invocation of the following UNIX utilities in the scripts you use.

| UNIX Utility | Function |
|-----------------|---|
| arp | Address resolution display and control. |
| awk | Pattern scanning and processing language. |
| basename | Return filename or directory portion of <code>pathname</code> . |
| bc | An arbitrary precision calculator language. |
| cat | Concatenate and print files. |
| chmod | Change file modes. |
| chown | Change file owner and group. |
| cksum | Display file checksums and block counts. |
| cut | Select portions of each line of a file. |
| date | Display or set date and time. |
| dd | Convert and copy a file. |
| df | Display free disk space. |
| env | Set and print environment. |
| expr | Evaluate expression. |
| fc | List the history of commands on the computer. |
| fg | Change the background process to the foreground process. |
| file | Determine the file type. |
| find | Walk a file hierarchy. |

| UNIX Utility | Function |
|---------------------|--|
| ftp | Internet file transfer program. |
| getopts | Called each time you process an argument. |
| grep | Print lines matching a pattern. |
| hostname | Set or print the name of the current host system. |
| ifconfig | Configure network interface parameters. |
| iostat | Report I/O statistics. |
| ln | Make links. |
| ls | List directory contents. |
| md5 | Calculates and verifies 128-bit MD5 hashes. |
| more | Filter for browsing text files. |
| netstat | Show the network status |
| nice | Execute a utility with an altered scheduling priority. |
| nohup | Invoke a command immune to hangups. |
| paste | Join files horizontally. |
| ping | Send ICMP ECHO_REQUEST packets to the network hosts. |
| ps | Process status. |
| pwd | Return working directory name. |
| sed | Stream editor. |
| sleep | Suspend execution for an interval of time. |
| sort | Sort or merge text files. |
| ssh | Open the SSH client (remote login program). |
| stty | Change the settings of a UNIX computer terminal. |
| tail | Display the last part of a file. |
| test | Condition evaluation utility. |
| ulimit | Get and set process limits. |
| umask | Set the file creation mode mask. |
| vmstat | Report virtual memory statistics. |
| wait | Await process completion. |
| wc | Word, line, and byte count. |
| who | Display the users who are currently logged in. |

Creating a User Name and Password for Smart Scripting

Before you run a script from the Dell Networking OS CLI, you may want to configure an additional user name and password to use only to run scripts on a switch.

Use the user name and password to log in to a UNIX shell and apply the read-write privileges assigned to the user name when a script runs with the `script` command from the Dell Networking OS CLI.

The user name is an optional keyword in the `script` command (refer to [Scheduling Time / Event-based Scripts](#)). To satisfy the requirements for a UNIX BSD login, the username must be fewer than 16 characters. A username you use to run scripts cannot contain special characters.

- Create an additional user name and password that you use to log in to a shell and apply read-write privileges when you run a script.

CONFIGURATION mode

```
username name password password
```

Logging in to a NetBSD UNIX Shell

To log in to the NetBSD UNIX shell on a switch to directly use any of the UNIX commands described in Supported UNIX Utilities or to run a script, use the `start shell` command.

You are prompted to enter a user name and password before you can access the shell. Login is performed using SSHv2.

- Access the shell to run UNIX commands or a script (refer to [Running a Script from the UNIX Shell](#)).

EXEC Privilege mode

```
start shell
```

Downloading Scripts to a Switch

Download a script to the switch using TFTP, FTP, or FLASH. Save the script to the dedicated script storage location, `/usr/pkg/ss-scripts`.

1. Configure the folders to mount a remote directory in the local Dell Networking OS path through a network file system (NFS).

CONFIGURATION mode

```
mount nfs nfs-server-ip: remote_dir mount_name [username username | password password]
```

2. Copy a script to a switch. Downloaded files are stored in the following path: `/usr/pkg/ss-scripts`.

EXEC

```
script get url
```

3. Remove a script from a switch.

EXEC

```
script remove {file_name file-name | all}
```

Setting a Search Path for Scripts

Create a path to the location where you stored the scripts. The Dell Networking OS uses this path when searching for the scripts. This way you do not have to specify a specific path when executing a script.

- Set a search path for a script in the Dell Networking OS.
CONFIGURATION mode

```
script path path-name
```

The script path can contain a network file-system-mounted directory (defined in the `mount nfs` command). The path is added to a script search list, allowing the system to search all locations for the script name. If the script is in multiple locations, the system uses the first instance of the script found.

Scheduling and Executing Scripts

Schedule scripts to execute periodically, based on an event and a specific time, such as at boot up or after you configure the switch.

Manage scripts to stop executing after a set time or to run at optimal times for critical resources, such as switch CPU load or packet loss.

To access the Dell Networking OS CLI via scripts, the Dell Networking OS provides a utility called `clido`, which is also a system command. For more information about the properties of `clido`, refer to [Dell Proprietary Utilities](#).

Script Scheduler from EXEC mode

To trigger scripts to run periodically or based on an event, use the following commands.

- **Schedule a script to run in EXEC mode.**

Schedule a script to execute at a specific time and optionally stop after a specified time. By default, the script runs in the foreground. To run the script in the background, use the `bg` parameter.

EXEC mode

```
script execute script-name [start {at | date-time | after time}] [stop {at  
date-time | after time}] [bg] [arguments arg[..arg]] [username username]
```

Following example shows the script execution at 22:22 10/30/14 (MM/DD/YY) and stops after 1 hour and runs the script in the background:

```
Dell#script execute collect_log.py start at 22:22-10/30/14 stop after  
0001:0000 args "Te 0/5" bg
```

- Scheduled scripts can be unconfigured/stopped/killed/resumed only by the configured user (test) or higher privileged user.
- In general, the Privilege level 15 (highest privilege user in Dell Networking OS CLI) user can configure/reconfigure the explicit username in `script execute` CLI.

The following example shows how you can schedule the script named `sample.zsh` to start after one hour and 12 minutes from now and runs the script in the background:

```
Dell#script execute sample.zsh start 01:12 bg
```

- **Schedule a script to run periodically.**

Schedule scripts to run at a certain time, to be repeated, to stop at a specified time or by a string of arguments. This script runs in the background.

CONFIGURATION mode

```
script execute script-name [start {at date-time | after time}] [stop {at
date-time | after time}] [watch] [arguments arg[..arg]] [username username]
```

Following example shows the script execution at 22:22 (FTOS clock time) and stops after 15 minutes:

```
Dell(conf)#script execute monitor_status.pl start at 22:22 stop after 15
watch args "Te 0/4"
```

The keyword `watch` is used to monitor the script. If the script execution terminates before 10 minutes, the system will rerun the script.

- **Executing a script based on event trigger.**

Define an event to trigger scripts to run by a log event (for example, SYSLOG in Dell Networking OS), a time-based event, or when CPU or memory usage reaches a predetermined percentage. The script runs in the background.

CONFIGURATION mode

```
script trigger-event event-name {log-event [{tag-or-ex tags | tag-and-ex
tags}] [severity severity level]} | time-event {[start {at date-time | after
time}] [stop {at date-time | after time | count number}] [interval time]
[weekday day] [ month month] [monthday nday]} | cpu-usage percentage | mem-
usage percentage}
```

Following example shows the event to trigger a script at 22:22 and then triggers every Monday midnight and stops at 02:02–11/10/14:

```
Dell(conf)#script trigger-event Event2 time-event start at 22:22 stop at
02:02-11/10/14 weekday mon
```

Following example shows the event triggering the script once the `cpu-usage` reaches 30%:

```
Dell(conf)#script trigger-event event1 cpu-usage 30
```

Following example shows how you can define the event to trigger a script when a SYSLOG event with the `OSTATE_UP` pattern occurs:

```
Dell(conf)#script trigger-event Port_State_Event log-event tag tag-regex
"OSTATE.*up"
```

Following example shows how you can define the event to trigger a script even if any one of the pattern is matched:

```
Dell(conf)#script trigger-event Server_Reporter_Event log-event tag tag-or-
ex "Major alarm" "Minor alarm"
```

- **Schedule a script to run based on defined events.**

Schedule a script to provide mapping between the scripts and trigger-events.

CONFIGURATION mode

```
script event-handler event-name
```

Following snippet triggers the script at 22:22 and then triggers every monday midnight and stops at 02:02-11/10/14:

```
Dell(conf)#script trigger-event Event2 time-event start at 22:22 stop at
02:02-11/10/14 weekday mon
Dell(conf)#script event-handler handler1
Dell(conf-screvt-handler)#execute Report_server.tcl args 4
Dell(conf-screvt-handler)#triggered-by Event2
Dell(conf-screvt-handler)#no disable
```

Following snippet triggers the script once the cpu-usage reaches 30%:

```
Dell(conf)#script trigger-event event1 cpu-usage 30
Dell(conf)#script event-handler handler1
Dell(conf-screvt-handler)#execute Report_server.tcl args 4
Dell(conf-screvt-handler)#triggered-by event1
Dell(conf-screvt-handler)#no disable
```

Following snippet triggers a script of the system throws the SYSLOG like Oct 29 05:28:16:

```
%RPM0-P:CP %IFMGR-5-OSTATE_UP: Changed interface state to up: Ma 0/0:
Dell(conf)#script trigger-event Port_State_Event log-event tag tag-regex
"OSTATE.*up"
Dell(conf)#script event-handler handler1
Dell(conf)#execute Report_server.tcl args 4
Dell(conf)#triggered-by Port_State_Event
Dell(conf)#no disable
```

Following snippet triggers a script even if any one of the pattern matches:

```
Dell(conf)#script trigger-event Server_Reporter_Event log-event tag tag-or-
ex "Major alarm" "Minor alarm"
Dell(conf)#script event-handler handler1
Dell(conf-screvt-handler)#execute Report_server.tcl args 4
Dell(conf-screvt-handler)#triggered-by Server_Reporter_Event
Dell(conf-screvt-handler)#no disable
```

Managing Executed Scripts

Use any of the following commands to manage or control the scripts; for example, stop, kill, resume, unscheduled, and clear.

Protect system resources by scheduling scripts to run when resources are within the threshold parameters you configured. System resources include the switch CPU, memory, and file system. For example:

```
Dell#show script process detail
JobID Script Type Script Status Username Script Name Args
000 TRIGGER Running admin admin.pl
001 REPEAT Running user1 sample.zsh hello
002 CONF WATCH Watch Blocked user2 args.pl
003 EXEC Scheduled user2 args.pl hi
Dell#
```

The script status are determined as follows:

- **Scheduled** — The script is scheduled to run.
- **Running** — The script is currently running.
- **Stopped** — The script is suspended.


- **Completed** — The script run is successful and exits normally.
- **Watch Blocked** — The script is monitored but terminates within ten minutes.
- **SSMGR-Stopped** — Dell Networking OS stops the script by itself, if the `cpu/disk-io/mem` goes beyond the threshold.
- To monitor the running script, use the `watch` command.

```
Dell#show script watch details
JobID Script Type Script Status Reset Count Watch StartTime Script NameArgs
Username
002 CONF WATCH Watch Blocked 3 Tue Dec 3 18:04:28 2013 args.pl
user2
```


- To control the script that is running based on the system resources such as the CPU, memory or disk I/O usage, use the `switch script limit cpu <percentage> memory <percentage> diskio <percentage>` command.

The scripting framework is enabled with system resource usage protection. Therefore, when the system is running at a high CPU, memory or disk I/O, the program automatically suspends all the user-configured running scripts and also prevents scheduling new scripts until the system resource usage returns to nominal levels. The following example suspends the scripts when the CPU crosses 70, memory usage is above 60, or the disk I/O is greater than 70.

```
Dell(conf)#switch script limit cpu 70 mem 60 diskio 70
```

 **NOTE:** For disk I/O, the `/usr/pkg` partition disk usage is considered.

- To suspend a currently running script, use the `script stop` command.
Dell#script stop script-name admin.pl
- To resume suspended script, use the `script resume` command.
Dell#script resume job-id 0
- To clear the supervised script that is in blocked state, use the `script clear` command.
Dell#script clear script-name args.pl
- To clear all the watch blocked state scripts, use the `script clear all` command.
Dell#script clear all
- To terminate a script that is running, use the `script kill` command.
Dell#script kill script-name admin.pl
- To terminate all the scripts that are running, use the `script kill all` command.
Dell#script kill all
- To unschedule an EXEC mode script, that is scheduled to run later, use the `script unschedule` command.
Dell#script unschedule job-id 4

 **NOTE:** For information about running a script directly from a UNIX shell without using the Dell Networking OS CLI, refer to [Running a Script from the UNIX Shell](#).

Viewing Script Information

To view information on currently stored, scheduled, and running SmartScripts, use the following commands.

- Display a list of stored files in the script path. To show the detail output of the file, enter `detail`. EXEC Privilege mode

```
show script file [detail]
```

- Display a list of scripts that are scheduled or running. To display the detailed status of the scripts, enter `detail`.

EXEC Privilege mode

```
show script process [detail]
```

- Display a list of supervised scripts that are scheduled or running. To display the detailed status of the supervised scripts, enter `detail`.

EXEC Privilege mode

```
show script watch [detail]
```

Running a Script from the UNIX Shell

You can run any Perl, Python, Ruby, and UNIX script stored on a switch from either the Dell Networking OS CLI or directly from a NetBSD shell on the switch.

For more information about running a script from the Dell Networking OS CLI, refer to [Scheduling Time / Event-based Scripts](#).

1. When you run a script from a UNIX shell, first access the shell using the `start shell` command.
2. You are prompted to enter a user name and password configured with the `username` command.

For more information, refer to [Creating a User Name and Password for Smart Scripting](#).

The following shows examples of how to execute a Perl, Python, and UNIX shell script directly from a NetBSD shell on Dell Networking OS.

- The first bold line shows logging in to a UNIX shell.
- The second bold line lists the existing scripts.
- The third bold line executes the Perl script using either command
- The fourth bold line executes the Python script using either command

```
Dell# start shell
4.4 BSD UNIX () (ttyp0)
login: admin
Password:
Copyright (c) 1996, 1997, 1998, 1999, 2000, 2001, 2002
The NetBSD Foundation, Inc. All rights reserved.
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
The Regents of the University of California. All rights reserved.
$ cd /f10/flash/scripts
$ ls
createVlans.pl createVlans.py createVlans.sh
$ ./createVlans.pl 12 14
$ perl createVlans.pl 12 14
$ ./createVlans.py 12 14
$ python createVlans.py 12 14
$ ./createVlans.sh 12 14
```

Running Scripts with User Privileges

Use these scripts to administer any Expect, PERL, Python, Ruby, Tcl, UNIX and ZSH shell scripts stored on the switch from the Dell Networking OS CLI.

To apply the associated read-write privileges while running a script from the Dell Networking OS CLI, specify an optional username (refer to XXXCreating a User Name and Password for Smart Scripting). If you do not specify a user name, the script runs with the privileges of the configured user.

To run a Perl, Python, Ruby or UNIX script from the Dell Networking OS CLI, use the `script` command. Enter the script name and directory path to start the script. The script can invoke any of the supported UNIX utilities listed in XXXSupported UNIX Utilities. Enter the command multiple times to run more than one script at the same time.

- Run an installed script. For `script-path`, enter the directory path and filename.
CONFIGURATION mode

```
script [username name] script-path [script-parameter script-parameter ...]
```

For `script-path`, enter the directory path and filename.

(Optional) For `username name`, enter the user name whose read-write privileges are applied when the script runs. A username running scripts cannot contain special characters.

(Optional) For `script-parameter`, enter the values of up to three parameters to be applied when the script runs. Enter a blank space between parameter values; for example: `script username admin /f10/flash/createVlans.py 1 2`

Smart Scripting CLI

Smart Scripting is supported on the S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL platforms.

disable

Allows you to enable or disable the current event handler configuration.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `disable`
 To reset the event handler, use the `[no] disable` command.

Defaults `none`

Command Modes `SCRIPT EVENT HANDLER CONFIGURATION`

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, and MXL. |

execute

Allows you to run the script based on the configured trigger event.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `execute script-name [arguments arg[..arg]]`

Parameters

| | |
|---|---|
| <i>script-name</i> | Enter the name of the script to schedule for execution. |
| <i>arguments arg</i> <i>[..args]</i> | Enter the keyword <code>arguments</code> then the arguments to script. The maximum length is 64 characters. The arguments can |

be any number of words within quotes and separated by a space.

Defaults none

Command Modes SCRIPT EVENT HANDLER CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, and MXL. |

mount nfs

Share the network file system the local Dell Networking OS file system uses.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `mount nfs nfs-server-ip: remote_dir mount_name [username username password password]`

To unmount the network file system, use the `no mount` command.

Parameters

| | |
|--|--|
| remote_dir mount_name | Enter the directory path where the network file system mounts. |
| username username | (Optional) Enter the keyword <code>username</code> then a text string up to 40 characters long as the user name. |
| password password | (Optional) Enter the keyword <code>password</code> then a text string up to 40 characters long as the password. |

Defaults none

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-----------------------------|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |

| Version | Description |
|----------|--|
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL. |

Usage Information

- All the mount points are maintained in the `/f10/mnt/nfs` folder.
- Only the relative path (mount point name) is acceptable. If the path you enter is either complete or absolute, an error occurs.
- If the mount point exists under the `f10/mnt`, you can reuse it or it is created under `/f10/mnt/nfs` and used.

Related Commands

[script path](#) — configures the search path to point to the local file share of NFS and runs the scripts in the NFS as unnamed.

package install

Install the Smart Scripting package. This command downloads the package from the specified location and installs it in the internal flash memory on a switch.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `package install location`

Parameters

location Enter the download and install location the Open Automation package, where location is one of the following values:

- From the local flash: `flash://filename`
- From an FTP server: `ftp://userid:password@host-ipaddress/filepath`
- From a TFTP server: `tftp://host-ipaddress/filepath`
- From an NFS mount server: `nfsmount://filepath`

Defaults none

Command Modes EXEC Privilege

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-------------------------------|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.2(0.0) | Introduced on the MXL switch. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.8.0 | Introduced on the S4810. |

Usage Information Because installing an Open Automation package may take time, the installation performs in the background when the download finishes. A message displays on the console when the installation is complete.

To follow the progress of a package installation, use the `show packages` command.

package uninstall

Remove an installed Open Automation package, such as Smart Scripting, from the system.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `package uninstall package-name`

Parameters *package-name* Enter the name of an Open Automation package, exactly as it appears in the `show packages` list.

Defaults none

Command Modes EXEC Privilege


Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-----------------------------|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |

| Version | Description |
|----------|-------------------------------|
| 9.5(0.1) | Introduced on the Z9500. |
| 9.2(0.0) | Introduced on the MXL switch. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.8.0 | Introduced on the S4810. |

Usage Information When you uninstall an Open Automation package, it is removed from the local flash memory.

 **CAUTION: Before you uninstall the Smart Scripting package, first stop all scripts that are running using the `no script script-name` command. Manually stop the `http-server` daemon by using the `no http-server {http | secure-http}` command.**

To follow the uninstall progress, use the `show packages` command.

Released Commands [show packages](#) — displays all Open Automation packages installed on the switch.

script (run)

Run an Expect, Perl, Python, Ruby, Tcl, UNIX, and ZSH shell script from the Dell Networking OS CLI.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `script [username name] script-name [script-parameter script-parameter ...]`

Parameters

username name (Optional) Enter the user name with read-write privileges. A username cannot contain special characters.

script-name Enter the name of the script to run, including the directory path and filename; for example:

- Perl script: `/usr/pkg/scripts/sample_scripts/cmd-server.pl`
- Python script: `/usr/pkg/scripts/sample_scripts/DisplayAlarms.py`
- Ruby script: `/usr/pkg/ss-scripts/interface_monitor.rb`
- UNIX shell script: `/usr/pkg/home/admin/test.sh`

script-parameter (Optional) Enter the values of up to three parameters to apply when the script runs. Enter a blank space between

parameter values. For example, `script username admin /f10/flash/createVlans.py 1 2`.

Defaults none

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-------------------------------|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.2(0.0) | Introduced on the MXL switch. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.8.0 | Introduced on the S4810. |

Usage Information To run more than one script at the same time, use the `script (run)` command multiple times; for example:

```
Dell(conf)#script username root /usr/pkg/scripts/  
sample_scripts/  
DisplayAlarms.py  
Dell(conf)#script username root /usr/pkg/bin/python /usr/pkg/  
scripts/  
VSNAgent/Xen/hpAgtMain.py
```

When you run a script from the Dell Networking OS CLI with the `script (run)` command, you can specify an optional user name to apply the read-write privileges assigned to the user name when the script runs. To configure the username and password, use the `username` command. If you do not specify a user name with the `script (run)` command, the script runs with the privileges of the current user.

For information about how to run a script directly from a UNIX shell, see the [Running a Script from the UNIX Shell](#).

To stop a running script, use the `no script script-name` command

To display the scripts that are currently running, including the scripts you have stopped, use the `show running-config | grep` command.

script (stop/resume/clear/kill/unschedule)

Stop, resume, clear, kill, or unschedule an Expect, Perl, Python, Ruby, Tcl, UNIX, and ZSH shell script from the Dell Networking OS CLI.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `script {stop | resume | clear | kill | unschedule} {script-name script-name | job-id job-id | all}`

Parameters

| | |
|--------------------|--|
| stop | Enter the keyword <code>stop</code> to stop a script from being run. |
| resume | Enter the keyword <code>resume</code> to restart a script that was stopped. |
| clear | Enter the keyword <code>clear</code> to restart the supervised scripts that were blocked. |
| kill | Enter the keyword <code>kill</code> to end a script from running. |
| unschedule | Enter the keyword <code>unschedule</code> to delete a script that was scheduled in EXEC mode. |
| script-name | Enter the keywords <code>script-name</code> then the name of the script <code>stop</code> , <code>resume</code> , <code>clear</code> , <code>kill</code> , or <code>unschedule</code> . |
| job-id | Enter the keywords <code>job-id</code> then the job identifier of the specific job <code>stop</code> , <code>resume</code> , <code>clear</code> , <code>kill</code> , or <code>unschedule</code> . |
| all | Enter the keyword <code>all</code> to stop, resume, clear, kill or unschedule all scripts. |

Defaults none

Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |

Usage Information Use the keyword `unschedule` only on scripts that are not currently running and that were scheduled using the `script execute` command in EXEC mode.

script event-handler

Allows you to provide mapping between the scripts and trigger events.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `script event-handler event-name`
To reset the script event, use the `no script event-handler` command.

Parameters `event-name` Enter the name of the script event to trigger.

Defaults none

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, and MXL. |

script execute (EXEC mode)

Schedule the scripts from EXEC mode either to run in foreground or background.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `script execute script-name [start {at | date-time | after time}] [stop {at date-time | after time}] [bg] [arguments arg[..arg]] [username username]`
To cancel, use the `script unschedule` command.

Parameters `script-name` Enter the name of the script to schedule for execution.
`start at date-time` Enter the keywords `start at` then the time and date in HH:MM-MM/DD/YY format to start running the script at a specific time and date or in HH:MM format to begin the script at the specified time on the same day (if the

mentioned time is elapsed, the script runs on the next day (HH:MM mentioned). The date can be the present or a future date.

start after time Enter the keywords `start after` then the time in [HHHH:] MMMM format to run the script after the set time. The date can be the present or a future date.

stop at date-time Enter the keywords `stop at` then the time and date in HH:MM-MM/DD/YY format to stop the script running at the specified time or in HH:MM format to stop the script at the specified time on the same day. The date must be a future date.

stop after time Enter the keywords `stop after` then the time in [HHHH:] MMMM format to stop running the script at the specified time.

bg Enter the keyword `bg` to schedule scripts to run in the background.

arguments arg [. . args] Enter the keyword `arguments` then the arguments to script. The maximum length is 64 characters. The arguments can be any number of words within quotes and separated by a space.

username username Enter the keyword `username` to use when the script runs. The maximum length is 16 characters. The default username is the user configuring the CLI.

bg Enter the keyword `bg` to schedule scripts to run in the background.

Defaults The script runs in the foreground.

Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, and MXL. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |

Usage Information All times selected follow the system time of the switch. The maximum number of scripts to configure is 100. An error message displays when you exceed the limit.

- Only the person who configured the scheduled scripts (or a higher privileged user) can manage the script.
- If you do not provide the username, the system uses current session username. If the script is triggered from the console, the system uses the Dell Networking OS default username (smartuser).

Example

```
Dell#script execute collect_log.py start at 22:22-10/30/14
stop after 0001:0000 args "Te 0/5" bg
```

This example illustrates that the script execution starts at 22:22 10/30/14 (MM/DD/YY) and stops after 1 hour and the script runs in the background.

script execute (CONFIGURATION mode)

Schedule the scripts to run across reloads.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax

```
script execute script-name [start {at date-time | after time}]
[stop {at date-time | after time}] [watch] [arguments
arg[..arg]] [username username]
```

To cancel the script execution, use the `no script execute` command.

Parameters

| | |
|-------------------------------------|--|
| <i>script-name</i> | Enter the name of the script to schedule for execution. |
| <i>start</i> | (Optional) Enter the keywords <code>start</code> to begin executing the script. |
| <i>start at date-time</i> | (Optional) Enter the keywords <code>start at</code> then the time and date either in HH:MM-MM/DD/YY format to start running the script at a specific time and date or in HH:MM format to begin the script at the specified time on the same day (if the mentioned time is elapsed, the script runs on the next day HH:MM mentioned). The date can be the present or a future date. |
| <i>start after date-time</i> | (Optional) Enter the keywords <code>start after</code> then the time in [HHHH:] MMMM format to start running the script after a set time. |
| <i>stop at date-time</i> | (Optional) Enter the keywords <code>stop at date-time</code> with the time and date either in HH:MM-MM/DD/YY format to stop running the script or in HH:MM format to stop the script at the specified time on the same day. The date must be a future date. |

| | |
|------------------------------|---|
| stop after time | (Optional) Enter the keywords <code>stop after time</code> then the time in [HHHH:] MMMM format to indicate the time after which the script stops running. For example, <code>stop after 00:30</code> indicates to stop the script execution 30 minutes after the start time. |
| watch | (Optional) Enter the keyword <code>watch</code> to monitor the script. |
| args arguments | (Optional) Enter the keyword <code>args</code> then the arguments script. The maximum length is 64 characters. The arguments can be any number of words within quotes and separated by a space. |
| username username | (Optional) Enter the username to use when the script runs. The maximum length is 16 characters. The default username is the user configuring the CLI. |

Defaults The script runs in the background.

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, and MXL. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |

Usage Information All times selected follow the system time of the switch. The maximum number of scripts to execute is 100. An error message displays when you exceed the limit.

If the supervised scripts exit (normally or in an unexpected manner):

- Within 10 minutes from the start, the supervised scripts restart (three retries).
- More than three times within 10 minutes, the script is in the Blocked state, which you can reset using the `script clear` command.
- TACACS/RADIUS users are not supported.
- Only the person who configures the scheduled scripts (or a higher privileged user) can change them.

- If you do not provide the username, the system uses current session username. If the script is triggered from the console, the system uses the Dell Networking OS default username (`smartuser`).

script get

Copy a script to a switch.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

| | | |
|------------------------|---|---|
| Syntax | <code>script get url</code> | |
| Parameters | url | Enter the keyword <code>url</code> then the URL location of the script to download to a switch. |
| Defaults | none | |
| Command Modes | EXEC | |
| Command History | This guide is platform-specific. For command information about other platforms, see the relevant <i>Dell Networking OS Command Line Reference Guide</i> . | |


The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|-------------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |

Usage Information The following formats are supported: Flash, FTP, TFTP, HTTP, and SCP. To retrieve the script files, use the following formats:

- Flash: Copy from the flash file system (`flash://filepath`)
- FTP: Copy from the remote file system using FTP (`ftp://userid:password@hostip/filepath`)
- HTTP: Copy from the remote file system using HTTP (`http://hostip/filepath`)
- SCP: Copy from the remote file system using SCP (`scp://userid:password@hostip/filepath`)
- TFTP: Copy from the remote file system using TFTP (`tftp://hostip/filepath`)

The downloaded files are stored into a dedicated folder (`/usr/pkg/ss-scripts`).

 **NOTE:** For stack systems, the script from `ss-scripts` synchronizes across the stack for every one hour. When the stack forms first, the script synchronization from the master to members happens only after 10 minutes.

script path

Configure the path for the script on the switch.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

| Syntax | <code>script path path-name</code> To remove the path for the script, use the <code>no script path</code> command. | | | | | | | | | | | | | | |
|--------------------------|--|-------------------------|---|------------|-----------------------------|----------|-----------------------------|----------|--------------------------|----------|--------------------------|----------|--------------------------|----------|---|
| Parameters | <table><tr><td><i>path-name</i></td><td>Enter the full path of the location of the script. Specify multiple paths using a colon (:). When initialized, the default path is <code>/user/pkg/ ss-scripts</code>.</td></tr></table> | <i>path-name</i> | Enter the full path of the location of the script. Specify multiple paths using a colon (:). When initialized, the default path is <code>/user/pkg/ ss-scripts</code> . | | | | | | | | | | | | |
| <i>path-name</i> | Enter the full path of the location of the script. Specify multiple paths using a colon (:). When initialized, the default path is <code>/user/pkg/ ss-scripts</code> . | | | | | | | | | | | | | | |
| Defaults | none | | | | | | | | | | | | | | |
| Command Modes | CONFIGURATION | | | | | | | | | | | | | | |
| Command History | <p>This guide is platform-specific. For command information about other platforms, see the relevant <i>Dell Networking OS Command Line Reference Guide</i>.</p> <p>The following is a list of the Dell Networking OS version history for this command.</p> <table><thead><tr><th>Version</th><th>Description</th></tr></thead><tbody><tr><td>9.8(0.0P2)</td><td>Introduced on the S3048-ON.</td></tr><tr><td>9.7(0.0)</td><td>Introduced on the S6000-ON.</td></tr><tr><td>9.6(0.0)</td><td>Introduced on the S5000.</td></tr><tr><td>9.5(0.1)</td><td>Introduced on the Z9500.</td></tr><tr><td>9.3(0.0)</td><td>Introduced on the S6000.</td></tr><tr><td>9.2(0.0)</td><td>Introduced on the S4810, S4820T, Z9000, and MXL Switch.</td></tr></tbody></table> | Version | Description | 9.8(0.0P2) | Introduced on the S3048-ON. | 9.7(0.0) | Introduced on the S6000-ON. | 9.6(0.0) | Introduced on the S5000. | 9.5(0.1) | Introduced on the Z9500. | 9.3(0.0) | Introduced on the S6000. | 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |
| Version | Description | | | | | | | | | | | | | | |
| 9.8(0.0P2) | Introduced on the S3048-ON. | | | | | | | | | | | | | | |
| 9.7(0.0) | Introduced on the S6000-ON. | | | | | | | | | | | | | | |
| 9.6(0.0) | Introduced on the S5000. | | | | | | | | | | | | | | |
| 9.5(0.1) | Introduced on the Z9500. | | | | | | | | | | | | | | |
| 9.3(0.0) | Introduced on the S6000. | | | | | | | | | | | | | | |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. | | | | | | | | | | | | | | |
| Usage Information | The path can contain a network file system (NFS) mounted directory (refer to the <code>mount nfs</code> CLI command). The path is added to a script search list allowing the system to search all locations for the script name. If the script is in multiple locations, the system uses the first instance of the script found. | | | | | | | | | | | | | | |
| Related Commands | mount nfs — sets up the folders in the NFS-mounted directory. | | | | | | | | | | | | | | |

script remove

Remove a script from a switch.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `script remove {file-name | all}`

Parameters

| | |
|-------------------------|--|
| <i>file-name</i> | Enter the filename of the script to remove from the switch. |
| all | Enter the keywords <code>all</code> to remove all files from the dedicated folder on the switch. |

Defaults none

Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |

Usage Information This command removes the files from the dedicated folder `/usr/pkg/ss-scripts` on the switch.

script trigger-event

Configure the event that causes the script to run.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500 and MXL

Syntax `script trigger-event event-name {log-event {tag [tag-or-ex tags | tag-and-ex tags | tag-regex tag]} [severity severity level]} | time-event {[start {at date-time | after time}] [stop {at date-time | after time | count number}] [interval time] [weekday day] [month month] [monthday nday]} | cpu-usage percentage | mem-usage percentage}`

To delete the trigger event, use the `no script trigger-event` command.

Parameters

| | |
|--|--|
| event-name | Enter the name of the script event to trigger. |
| log-event tag tag-or-ex tags | Enter the keywords <code>log-event tag tag-or-ex</code> then the pattern (tag) in the syslog message to trigger the event if any pattern matches the syslog message. You can enter a maximum of three tags. Separate each tag by commas. |
| log-event tag tag-and-ex tags | Enter the keywords <code>log-event tag tag-and-ex</code> then the pattern (tag) in the syslog message to trigger the event if all the patterns match the syslog message. You can enter a maximum of three tags. Separate each tag by commas. |
| log-event tag tag-regex tags | Enter the keywords <code>log-event tag tag-regex</code> then the regex pattern (tag) in the syslog message to trigger the event based on the regular expression. You can enter only one tag. |
| log-event severity severity level | Enter the keywords <code>log-event severity</code> then the severity level to define a trigger based on the syslog message severity level. Valid message security levels are from 1 to 6. |
| time-event start at date-time | Enter the keywords <code>time-event start at</code> then the time and date either in HH:MM – MM/DD/YY format to begin the trigger event at the specified time and date or in HH:MM format to begin the trigger event at the specified time on the same day (if the mentioned time is elapsed, the event runs on the next day HH:MM mentioned). The date can be the present or future date. |
| time-event start after time | Enter the keyword <code>start after</code> then the time in [HHHH:] MMMM format to begin the trigger event after the specified time. |
| time-event stop at date-time | Enter the keyword <code>stop at</code> then the time and date either in HH:MM – MM/DD/YY format to stop the trigger event at the specified time and date or in HH:MM format to begin the trigger event at the specified time on the same day (if the mentioned time is elapsed, the event runs on the next day HH:MM mentioned). The date must be a future date. |
| time-event stop after time | Enter the keyword <code>stop after</code> then the time in [HHHH:] MMMM format to stop the trigger event after the specified time. |
| count number | Enter the keyword <code>count</code> then the number of times to trigger the script event. The range is from 1 to 100. |
| interval time | Enter the keyword <code>interval</code> then the time in [HHHH:] MMMM format, in minutes, to schedule the script execution interval. |

| | |
|---------------------------------------|--|
| weekday <i>day</i> | Enter the keyword <code>weekday</code> then the day of a week to run the script. Select one or more days from the following values: <code>mon</code> , <code>tue</code> , <code>wed</code> , <code>thu</code> , <code>fri</code> , <code>sat</code> , and <code>sun</code> . Separate each day by commas. |
| month <i>month</i> | Enter the keyword <code>month</code> then the month to run the script. Select one or more months from the following values: <code>jan</code> , <code>feb</code> , <code>mar</code> , <code>apr</code> , <code>may</code> , <code>jun</code> , <code>jul</code> , <code>aug</code> , <code>sep</code> , <code>oct</code> , <code>nov</code> , and <code>dec</code> . Separate each month by commas. |
| monthday <i>nday</i> | Enter the keyword <code>monthday</code> then the day to run the script event. The range is from 1 to 31. Separate each day by commas. |
| cpu-usage percentage | Enter the keywords <code>cpu-usage</code> then a percentage value between 20 and 90 to define the trigger event based on CPU usage. |
| mem-usage percentage | Enter the keywords <code>mem-usage</code> then a percentage value between 20 and 90 to define the trigger event based on memory usage. |

Defaults none

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.


| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, Z9500, and MXL. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL Switch. |

Usage Information The maximum number of events you can configure is 20. The maximum number of allowed tag patterns in a log event trigger is three. The tags of the log events must be an exact comparison. The three tags work in a logical OR operation format (match with any one of the tags occurs).



NOTE: Before deleting a trigger event, you **MUST** unlink the script from the event.

If you use the caret symbol (^) as a pattern, the Dell Networking OS generates blank space as the starting character of the SYSLOG. The special character \$ is not allowed to use in regular expressions, as the last word in the SYSLOG is line feed.

 **NOTE:** The time-events schedule time takes effect once after the device configures the trigger-event (event).

Related Command [script event-handler](#) — Allows you to provide mapping between the scripts and trigger events.

Examples Following example shows how the event triggers a script, even if any one of the pattern matches.

```
Dell(conf)# script trigger-event Server_Reporter_Event log-event tag tag-or-ex "Major alarm" "Minor alarm"
```

Following example shows how the event triggers a script, if the system throws the SYSLOG message like: Oct 29 05:28:16: %RPM0-P:CP %IFMGR-5-OSTATE_UP: Changed interface state to up: Ma 0/0.

```
Dell(conf)# script trigger-event Port_State_Event log-event tag tag-regex "OSTATE.*up"
```

Following example shows the script triggering at 22:22 and then triggers every Monday midnight and stops at 02:02–11/10/14.

```
Dell(conf)# script trigger-event Event2 time-event start at 22:22 stop at 02:02-11/10/14 weekday mon
```

Following example shows the cpu-usage limit.

```
Dell(conf)# script trigger-event event1 cpu-usage 30
```

show packages


Display the installed Open Automation packages, including version number and contents.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `show packages [system]`

Parameters

| | |
|---------------|---|
| system | (Optional) Enter the keyword <code>brief</code> to display system information about the package, version, and status of the package in all stack-units. |
|---------------|---|

 **NOTE:** This option is only available on switches that allow stacking.

Defaults none

Command Modes

- EXEC

- EXEC Privilege

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-------------------------------|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.2(0.0) | Introduced on the MXL Switch. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.8.0 | Introduced on the S4810. |

Usage Information In a stack setup, the Dell Networking OS automatically synchronizes the software version installed on the master stack unit with the other switches in the stack. To view the installed software versions on the stack units, use the `show packages system` command.

Example (show packages)

```
Dell# show packages
* Package Name: SMARTSCRIPTS Version: 2.0.0
  Python 2.6.5
  Perl 5.8.8
  Data::Dumper 2.126
  Class::MethodMaker 2.16
  ExtUtils::MakeMaker 6.56
  XML::NamespaceSupport 1.11
  XML::SAX 0.96
  XML::LibXML 1.70
  Compress::Raw::Bzip2 2.027
  Compress::Raw::Zlib 2.027
  IO::Compress 2.027
  URI 1.54
  HTML::Tagset 3.20
  HTML::Parser 3.65
  LWP 5.836
  Net::Telnet 3.03
  OSSP::uuid 1.0602
  UUID 0.02
  version 0.82
  Class::Inspector 1.24
  Task::Weaken 1.03
  Algorithm::Diff 1.1902
  Text::Diff 1.37
  SOAP::Lite 0.712
  Crypt::SSLeay 0.57
```

```

URI::urn::uuid 0.03
UUID 0.03
Crypt::SSLey 0.57
Net::SNMP 6.0.0
Net::Telnet::Cisco 1.10

```

```

HTTP Server
mini_httpd 1.19
Perl and Python function library for Force10 SmartScripts
smartutils 2.0.0
WebConnect Web UI and CGI scripts
htdocs 2.0.0

```

Example (show packages system)

```

Dell#show packages system

Package Information
-----
Unit Package Name Version Status
-----
0 SMARTSCRIPTS 2.9.9.2 Installed
  nano 2.2.6nb1 Installing
1 SMARTSCRIPTS 2.9.9.2 Installing
  nano 2.2.6nb1 Installing
2 not present
3 not present
4 not present
5 not present
6 not present
7 not present
8 not present
9 not present
10 not present
11 not present

```

show script

Display the stored, scheduled, and running scripts.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `show script {file | process | watch} | [detail]`

Parameters

- file** Enter the keyword `file` to list the stored files in the script path. Enter the optional keyword `detail` to show detailed output of the scripts including job-id, script type, and script status.
- process** Enter the keyword `process` to list the scripts that are scheduled or running. Enter the optional keyword `detail` to show detailed output of the relevant arguments that are scheduled or running.
- watch** Enter the keyword `watch` to list the supervised scripts and their relevant details.

detail (Optional) Enter the keyword `detail` to show detailed output of the file including CPU percentage, memory percentage, next scheduled time, and any script name or relevant arguments.

Defaults none

Command Modes EXEC Privilege

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S6000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL switch. |

start shell

Start a NetBSD UNIX shell.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `start shell`

Defaults none

Command Modes EXEC Privilege

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-------------------------------|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.2(0.0) | Introduced on the MXL Switch. |

| Version | Description |
|----------|---------------------------|
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.8.0 | Introduced on the S4810. |

Usage Information To invoke the Dell Networking OS operations, start an NetBSD shell on a switch before you can enter UNIX commands or run a script directly from the shell (see [Running a Script from the UNIX Shell](#)).

After you start a shell, you are prompted to enter a username and password.

Related Commands [show packages](#) — displays all the Open Automation packages installed on the switch.

triggered-by

Allows you to associate the events to a script in Event Handler Configuration mode.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax `triggered-by trigger-name`

Parameters `trigger-name` Enter the name of the script event to trigger.

Defaults none

Command Modes SCRIPT EVENT HANDLER CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000–ON, Z9000, Z9500, and MXL. |

switch script limit

To control the script that is running based on CPU, memory, or disk IO usage, use the switch script limit.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, Z9500, and MXL

Syntax `switch script limit {cpu percentage mem percentage disk percentage}`
To return to the default value, use the `no switch script limit` command.

Parameters

- cpu *percentage*** Enter the keyword `cpu` and the maximum percentage limit for the script execution. The range is from 20 to 90 percent.
- mem *percentage*** Enter the keyword `mem` and the maximum percentage limit for the script execution. The range is from 20 to 90 percent.
- disk *percentage*** Enter the keyword `disk` and the maximum percentage limit for the script execution. The range is from 20 to 90 percent. Only `/usr/pkg` is monitored.

Defaults none

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.3(0.0) | Introduced on the S6000. |
| 9.2(0.0) | Introduced on the S4810, S4820T, Z9000, and MXL switch. |

Usage Information When the system reaches any maximum percentage limit, all running scripts suspend and additional scripts are not scheduled. Scripts begin running when any of the resource limits reaches 10 percent less than the configured value. Details are entered into syslog.

username (event handler)

Allows you to configure the username to run the scripts.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500 and MXL

Syntax `username username privilege privilege`

Parameters

| | |
|--------------------------------------|--|
| username <i>username</i> | Enter the keyword <code>username</code> then the name of the user to access the script. The maximum length is 16 characters. The default username is the user configuring the CLI. |
| privilege <i>privilege</i> | Enter the keyword <code>privilege</code> then the privilege level to run the scripts. The range is from 1 to 15. |

Defaults none

Command Modes SCRIPT EVENT HANDLER CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000–ON, Z9000, Z9500, and MXL. |

Usage Information

- Only the person who configures the scheduled scripts (or a higher privileged user) can change them.
- If you do not provide the username, the system uses current session username. If the script is triggered from the console, the system uses the Dell Networking OS default username (`smartuser`).
- The privilege level mentioned should be equal to or less than the user who is configuring the command.

Example `username smartuser privilege 15`

Smartuser is the default username used for running the scripts on the device, which takes the privilege based on the user session.

username

Configure an additional username and password to run the scripts on a switch. Use the username and password to log in to a UNIX shell and apply the read-write privileges assigned to the username when a script runs.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, Z9500, and MXL

Syntax

```
username name password password
```

To remove the username and password, use the `no username` command.

Parameters

| | |
|--|--|
| <i>name</i> | Enter a username to access the UNIX shell. The length of the username can be up to 16 characters to satisfy the BSD UNIX login requirements. A username running the scripts cannot contain any special characters. |
| <i>password</i> <i>password</i> | Enter a password to access the UNIX shell. |

Defaults

none

Command Modes

CONFIGURATION

Command History

This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|-------------------------------|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.2(0.0) | Introduced on the MXL Switch. |
| 9.0.2.0 | Introduced on the S6000. |
| 9.0.0.0 | Introduced on the Z9000. |
| 8.3.19.0 | Introduced on the S4820T. |
| 8.3.8.0 | Introduced on the S4810. |

Usage Information

When you run a script from the Dell Networking OS CLI with the `script (run)` command, you can specify an optional username to apply the read-write privileges assigned to the username when the script runs (see [Scheduling Time / Event-based Scripts](#)).

REST API

Representational state transfer (REST) application programming interface (API) is an integrated part of the Dell Networking operating system (OS).

The Dell Networking OS supports the S4810, S4820T, S3048-ON, S5000, S6000, and Z9500 platforms.

HTTP and HTTPS

Use REST API to configure and monitor a Dell Networking switch over the hyper text transfer protocol (HTTP) and hyper text transfer protocol secure (HTTPS).

HTTP and HTTPS are the common protocols that support read, create, update, and delete operations with the actions using methods such as GET, POST, PUT, PATCH, and DELETE. For more information, see *RFC 2616, Hypertext Transfer Protocol - HTTP/1.1*.

XML

Extensible markup language (XML) is a standardized, easy-to-read and easy-to-parse method to represent data. Use the XML protocol data unit (PDU) to exchange data between the Dell Networking switch and the HTTP/HTTPS client.

XML-based implementation uses *RFC 4741, NETCONF Configuration Protocol, and RFC 6020, YANG — a Data Modeling Language for NETCONF* standards. The system handles up to four simultaneous REST API requests.

You can use REST API on switches without having to code individual CLI commands and open telnet, SSH, or console connections for each command.

In addition to REST API, you can use third-party management tools and other industry-standard management protocols to manage Dell Networking switches.

Important Points to Remember

- Internet protocol (IP) reachability is required from the REST client to the Dell Networking switch.
- For enabling and disabling the REST API, see the [REST API CLI](#).
- You cannot delete physical interfaces using REST API.
- Port 8008 is a nonsecure port HTTP uses; port 8888 is a secure port HTTPS uses.

REST Authentication

The REST API authenticates and authorizes the user based on the Dell Networking OS AAA configuration. You can locally configure or validate through the AAA infrastructure. For more information about the AAA Accounting mode, see the *Dell Networking OS Configuration Guide, Security* chapter.

The Dell Networking OS CLI user level privilege dictates the level of the REST API access. Users with privilege level 0 or 1 have read-only access; the allowed REST API method is GET. Users with privilege levels 2 through 15 have read-write access in REST API. The allowed methods are GET, PATCH, PUT, POST, and DELETE.

For information about the privilege level command, see the *Dell Networking OS Command Line Reference Guide, Security* chapter.

Request Query Parameters

Following are the various request query parameters supported on REST.

- **deep** — Retrieves a resource with all the subresources inline.
- **shallow** — Retrieves a resource with no subresources inline.
- **select** — Client uses to select the nodes and subresources.
- **with-defaults** — Client uses to control and display the default data in GET request.

Sample BGP Configurations

The following are the sample BGP configurations via REST API.



Figure 2. BGP Topology

The PDU structure for configuring the IP address on an interface is as follows:

```
cat ipaddr_config_put.xml
<interface>
  <name>tengig-0-0</name>
  <ip>
    <address>
      <primary>
        <address>11.1.1.1/24</address>
      </primary>
    </address>
  </ip>
</interface>
```

```
</ip>
</interface>
```

The following configuration depicts the IP address configuration for router interfaces via **REST- PUT** request:

```
curl -v -u admin:admin http://10.16.151.159:8008/api/running/dell/interfaces/
interface/tengig-0-0 -X PUT -T ipaddr_config_put.xml

* About to connect() to 10.16.151.159 port 8008
*   Trying 10.16.151.159... connected
* Connected to 10.16.151.159 (10.16.151.159) port 8008
* Server auth using Basic with user 'admin'
> PUT /api/running/dell/interfaces/interface/tengig-1-37 HTTP/1.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.15.5 (i686-redhat-linux-gnu) libcurl/7.15.5 OpenSSL/0.9.8b
zlib/1.2.3 libidn/0.6.5
> Host: 10.16.151.159:8008
> Accept: */*
> Content-Length: 171
> Expect: 100-continue
>
< HTTP/1.1 100 Continue
< Server: ConfD/5.2
< Allow: GET, POST, OPTIONS, HEAD
< Content-Length: 0
< HTTP/1.1 204 No Content
< Server: ConfD/5.2
< Cache-control: private, no-cache, must-revalidate, proxy-revalidate
< Date: Thu, 04 Dec 2014 13:13:25 GMT
< Allow: GET, POST, OPTIONS, HEAD
< Content-Length: 0
< Content-Type: text/html
< Etag: 1417-698805-339353@0
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0
```

Following is the current running configurations applied on the system:

```
Dell#show running-config interface tengigabitethernet 0/0
!
interface TenGigabitEthernet 0/0
 ip address 11.1.1.1/24
 shutdown
Dell#
```

The following configuration retrieves the interface running-config via REST API.

```
curl -u admin:admin http://10.16.151.159:8008/api/running/dell/interfaces/
interface/tengig-0-0\?deep
<interface xmlns="http://www.dell.com/ns/dell:0.1/root" xmlns:y="http://tail-
f.com/ns/rest" xmlns:dell="http://www.dell.com/ns/dell:0.1/root">
  <name>tengig-0-0</name>
  <mtu>1554</mtu>
  <shutdown>>true</shutdown>
  <ip>
    <address>
      <primary>
        <address>11.1.1.1/24</address>
      </primary>
    </address>
```

```

</ip>
<speed>auto</speed>
<rate-interval>299</rate-interval>
</interface>
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0

```

The PDU structure for configuring BGP on the Router 1 is as follows:

```

cat BGP_HTTP_REST_POST_PDU.xml
<bgp>
  <as-name>100</as-name>
  <neighbor>
    <neighbor-router>11.1.1.2</neighbor-router>
    <remote-as>200</remote-as>
    <dmzlink-bw>false</dmzlink-bw>
    <shutdown>false</shutdown>
  </neighbor>
</bgp>

```

The following configuration depicts the BGP configurations for router through the **POST** request:

```

curl -v -u admin:admin http://10.16.151.159:8008//api/running/dell/router -X
POST -T BGP_HTTP_REST_POST_PDU.xml

* About to connect() to 10.16.151.159 port 8008
*   Trying 10.16.151.159... connected
* Connected to 10.16.151.159 (10.16.151.159) port 8008
* Server auth using Basic with user 'admin'
> POST /api/running/dell/router HTTP/1.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.15.5 (i686-redhat-linux-gnu) libcurl/7.15.5 OpenSSL/0.9.8b
zlib/1.2.3 libidn/0.6.5
> Host: 10.16.151.159:8008
> Accept: */*
> Content-Length: 493
> Expect: 100-continue
>
< HTTP/1.1 100 Continue
< Server: ConfD/5.2
< Allow: GET, POST, OPTIONS, HEAD
< Content-Length: 0
< HTTP/1.1 201 Created
< Server: ConfD/5.2
< Location: http://10.16.151.159:8008/api/running/dell/router/bgp/100
< Cache-control: private, no-cache, must-revalidate, proxy-revalidate
< Date: Thu, 04 Dec 2014 13:21:03 GMT
< Allow: GET, POST, OPTIONS, HEAD
< Content-Length: 0
< Content-Type: text/html
< Etag: 1417-699263-712815@0
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0

```

Following is the current configurations applied on the system:

```

Dell#show config
!
router bgp 100
  neighbor 11.1.1.2 remote-as 200
  neighbor 11.1.1.2 no shutdown
Dell#

```

The following configuration retrieves the BGP running-config via REST API:

```
curl -u admin:admin http://10.16.151.159:8008/api/running/dell/router/bgp
<collection xmlns:y="http://tail-f.com/ns/rest">
  <bgp>
    <as-name>100</as-name>
    <timers>
      <bgp>
        <keepalive>60</keepalive>
        <hold-time>180</hold-time>
      </bgp>
    </timers>
    <maximum-paths>
      <ebgp>1</ebgp>
      <ibgp>1</ibgp>
    </maximum-paths>
    <bgp>
      <dmzlink-bw>>false</dmzlink-bw>
    </bgp>
    <neighbor>
      <neighbor-router>11.1.1.2</neighbor-router>
    </neighbor>
  </bgp>
</collection>
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0
```

The following configuration depicts adding another BGP peer to the configuration via **REST- PATCH** request:

```
cat BGP_HTTP_REST_PATCH_PDU.xml
```

```
<bgp>
  <as-name>100</as-name>
  <neighbor>
    <neighbor-router>11.1.1.3</neighbor-router>
    <remote-as>200</remote-as>
    <dmzlink-bw>>false</dmzlink-bw>
    <shutdown>>false</shutdown>
  </neighbor>
</bgp>
```

```
curl -v -u admin:admin http://10.16.151.159:8008/api/running/dell/router/bgp -X
PATCH -T BGP_HTTP_REST_PATCH_PDU.xml
```

```
* About to connect() to 10.16.151.159 port 8008
* Trying 10.16.151.159... connected
* Connected to 10.16.151.159 (10.16.151.159) port 8008
* Server auth using Basic with user 'admin'
> PATCH /api/running/dell/router/bgp HTTP/1.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.15.5 (i686-redhat-linux-gnu) libcurl/7.15.5 OpenSSL/0.9.8b
zlib/1.2.3 libidn/0.6.5
> Host: 10.16.151.159:8008
> Accept: */*
> Content-Length: 493
> Expect: 100-continue
>
< HTTP/1.1 100 Continue
< Server: ConfD/5.2
< Allow: GET, POST, OPTIONS, HEAD
< Content-Length: 0
```

```
< HTTP/1.1 204 No Content
< Server: Confd/5.2
< Cache-control: private, no-cache, must-revalidate, proxy-revalidate
< Date: Thu, 04 Dec 2014 13:25:31 GMT
< Allow: GET, POST, OPTIONS, HEAD
< Content-Length: 0
< Content-Type: text/html
< Etag: 1417-699531-877544@0
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0
```

Following is the current configurations applied on the system:

```
Dell#show config
!
router bgp 100
 neighbor 11.1.1.2 remote-as 200
 neighbor 11.1.1.2 no shutdown
 neighbor 11.1.1.3 remote-as 200
 neighbor 11.1.1.3 no shutdown
Dell#
```

The following configuration retrieves the BGP running-config via REST API:

```
curl -u admin:admin http://10.16.151.159:8008/api/running/dell/router/bgp\?deep
<collection xmlns:y="http://tail-f.com/ns/rest">
  <bgp>
    <as-name>100</as-name>
    <timers>
      <bgp>
        <keepalive>60</keepalive>
        <hold-time>180</hold-time>
      </bgp>
    </timers>
    <maximum-paths>
      <ebgp>1</ebgp>
      <ibgp1>1</ibgp1>
    </maximum-paths>
    <bgp>
      <dmzlink-bw>false</dmzlink-bw>
    </bgp>
    <neighbor>
      <neighbor-router>11.1.1.2</neighbor-router>
      <remote-as>200</remote-as>
      <dmzlink-bw>false</dmzlink-bw>
      <shutdown>false</shutdown>
    </neighbor>
    <neighbor>
      <neighbor-router>11.1.1.3</neighbor-router>
      <remote-as>200</remote-as>
      <dmzlink-bw>false</dmzlink-bw>
      <shutdown>false</shutdown>
    </neighbor>
  </bgp>
</collection>
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0
```

The following configuration is to fetch the operational data for the **BGP NLRI** table via REST API (Router 1):

```
curl -u admin:admin http://10.16.151.159:8008/api/operational/mib/bgp4-v2/
f10BgpM2NlriTable\?deep
<collection xmlns:y="http://tail-f.com/ns/rest">
  <f10BgpM2NlriTable>
    <f10BgpM2PeerIndex>1</f10BgpM2PeerIndex?
    <f10BgpM2NlriAfi>1</f10BgpM2NlriAfi>
    <f10BgpM2NlriSafi>1</f10BgpM2NlriSafi>
    <f10BgpM2NlriPrefix>5.0.0.0</f10BgpM2NlriPrefix>
    <f10BgpM2NlriPrefixLen>24</f10BgpM2NlriPrefixLen>
    <f10BgpM2NlriIndex>0</f10BgpM2NlriIndex>
    <f10BgpM2NlriBest>true</f10BgpM2NlriBest>
    <f10BgpM2NlriCalcLocalPref>100</f10BgpM2NlriCalcLocalPref>
    <f10BgpM2PathAttrIndex>1860238788</f10BgpM2PathAttrIndex
    <f10BgpM2NlriOpaqueType>none</f10BgpM2NlriOpaqueType>
    <f10BgpM2RouteFlag>active</f10BgpM2RouteFlag>
  </f10BgpM2NlriTable>
</collection>
* Connection #0 to host 10.16.151.159 left intact
* Closing connection #0
```

HTTP Status Error Codes

The REST API server returns the following HTTP status error codes.

| Status Error Codes | Description |
|----------------------------|--|
| 200 OK | The request was successfully completed. A response body returns containing a representation of the resource. |
| 201 Created | A resource is created and the new resource URI returns in the "Location" header. |
| 204 No Content | The request is successfully completed but no response body returns. |
| 400 Bad Request | The request could not be processed because it contains missing or invalid information (such as a validation error on an input field, a missing required value, and so on). |
| 401 Unauthorized | The request requires user authentication. The response includes a "WWW-Authenticate" header field for basic authentication. |
| 403 Forbidden | Access to the resource is denied by the server due to authorization rules. |
| 404 Not Found | The requested resource does not exist. |
| 405 Method Not Allowed | The HTTP method specified in the request (DELETE, GET, HEAD, PATCH, POST, PUT) is not supported for this resource. |
| 406 Not Acceptable | The resource this request identifies is not capable of generating the requested representation, specified in the "Accept" header or in the "format" query parameter. |
| 409 Conflict | This code is used if a request tries to create a resource that already exists. |
| 415 Unsupported Media Type | The format of the request is not supported. |
| 500 Internal Error | The server encountered an unexpected condition which prevented it from fulfilling the request. |

| Status Error Codes | Description |
|--------------------|-------------|
|--------------------|-------------|

| | |
|---------------------|--|
| 501 Not Implemented | The server does not (currently) support the functionality required to fulfill the request. |
|---------------------|--|

| | |
|-----------------|---|
| 503 Unavailable | The server is currently unable to handle the request because the resource is being used by someone else or is temporarily overloaded. |
|-----------------|---|

REST API — Protocol Data Unit (PDU) Structure

The following features are supported to configure the REST API.

- Physical interface
- Logical interface
- BGP
- Infrastructure
- Miscellaneous

Each node is printed as: `<status> <flags> <name> <opts> <type>`.

- `<status>` is one of the following:
 - + for current
 - x for deprecated
 - o for obsolete
- `<flags>` is one of the following:
 - rw for configuration data
 - ro for non-configuration data
 - -x for rpcs
 - -n for notifications
- `<name>` is the name of the node.
 - (`<name>`) refers that the node is a choice node
 - :(`<name>`) refers that the node is a case node
 - -x for rpcs.



NOTE: If the node is augmented into the tree from another module, the name is printed as `<prefix>:<name>`.

- `<opts>` is one of the following:
 - ? for an optional leaf or presence container
 - * for a leaf-list
 - [`<keys>`] for a list's keys
- `<type>` is the name of the type for leafs and leaf-lists.

Configurations

This section describes the properties of all the configurations.

BGP

The following definition configures and displays the properties of a BGP configuration.

Module: BGP
URI: /api/running/dell/router/bgp
Methods Supported: GET, POST, PUT, PATCH, and DELETE

BGP Configuration Model

```
+--rw dell
  +--rw system
  +--rw interfaces
  +--rw router
    +--rw bgp [as-name]
      +--rw as-name          dell:as-number
      +--rw bgp
        +--rw asnotation?    enumeration
        +--rw always-compare-med? boolean
        +--rw bestpath?
          +--rw as-path?      enumeration
          +--rw med?
            +--rw confed?      boolean
            +--rw missing-as-best? boolean
            +--rw router-id-ignore? boolean
        +--rw client-to-client?
          +--rw reflection?    boolean
        +--rw cluster-id?      union
        +--rw confederation?
          +--rw identifier?    dell:as-number
          +--rw peers*         dell:as-number
        +--rw default?
          +--rw local-preference? uint32
        +--rw dmzlink-bw?      boolean
        +--rw enforce-first-as? boolean
        +--rw fast-external-fallover? boolean
        +--rw four-octet-as-support? boolean
        +--rw graceful-restart?
          +--rw restart-time?  uint16
          +--rw stale-path-time? uint16
          +--rw role?          enumeration
        +--rw log-neighbor-changes? boolean
        +--rw non-deterministic-med? boolean
        +--rw recursive-bgp-next-hop? boolean
        +--rw regex-eval-optz-disable? boolean
        +--rw connection-retry-timer? dell:seconds
        +--rw router-id?       dell:ip-address
        +--rw add-path?
          +--rw addpath?       enumeration
          +--rw path-count?    uint8
          +--rw enable?        boolean
        +--rw dampening?
          +--rw half-life?     uint32
```

```

|         +---rw reuse?                uint32
|         +---rw suppress?             uint32
|         +---rw max-suppress-time?   uint32
|         +---rw route-map?           dell:route-map-name
+---rw default-metric?                uint32
+---rw description?                   string
+---rw timers
|   +---rw bgp
|     +---rw keepalive?               dell:seconds
|     +---rw hold-time?               dell:seconds
+---rw maximum-paths
|   +---rw ebgp?                      uint32
|   +---rw ibgp?                      uint32
+---rw network [ipv4-address]
|   +---rw ipv4-address               dell:ipv4-address-with-prefix
|   +---rw route-map?                 dell:route-map-name
|   +---rw backdoor?                  boolean
+---rw peer-group [peer-group-name]
|   +---rw peer-group-name            dell:peer-group-name
|   +---rw passive?                   boolean
|   +---rw limit?                     uint32
|   +---rw match-af?                  boolean
|   +---rw remote-as?                 dell:as-number
|   +---rw description?               string
|   +---rw ebgp-multihop?
|     | +---rw ttl?                   uint8
|     +---rw fall-over?                boolean
|   +---rw send-community?
|     | +---rw community-type?        enumeration
|   +---rw shutdown?                  boolean
|   +---rw update-source-loopback?    uint16
|   +---rw weight?                    uint16
|   +---rw timers
|     | +---rw keepalive?             dell:seconds
|     | +---rw hold-time?             dell:seconds
|   +---rw add-path?
|     | +---rw addpath?                enumeration
|     | +---rw path-count?            uint8
|   +---rw local-as?
|     | +---rw as-number?              dell:as-number
|     | +---rw no-prepend?             boolean
|   +---rw password?
|     | +---rw encryption-type?        enumeration
|     | +---rw password-value?        string
|   +---rw activate?                  boolean
|   +---rw advertisement-interval?     dell:seconds
|   +---rw allowas-in?                 uint8
|   +---rw default-originate?
|     | +---rw route-map?             dell:route-map-name
+---rw distribute-list [direction]
|   | +---rw direction                dell:bgp-opt-dir
|   | +---rw name?                    dell:prefix-list-name
+---rw filter-list [direction]
|   | +---rw direction                dell:bgp-opt-dir
|   | +---rw name?                    dell:aspath-access-list-name
+---rw maximum-prefix?
|   | +---rw maximum?                  uint32
|   | +---rw threshold?                uint8
|   | +---rw warning-only?             boolean
+---rw next-hop-self?                  boolean
+---rw remove-private-as?              boolean
+---rw route-map [direction]
|   | +---rw direction                dell:bgp-opt-dir
|   | +---rw name?                    dell:route-map-name

```

```

| | +--rw route-reflector-client?      boolean
| | +--rw sender-side-loop-detection?  boolean
| | +--rw soft-reconfiguration?        enumeration
| | +--rw subnet?                      dell:ip-address-with-prefix
| | +--rw dmzlink-bw?                  boolean
|--rw neighbor [neighbor-router]
| | +--rw neighbor-router              union
| | +--rw peer-group-name?            dell:peer-group-name
| | +--rw remote-as?                   dell:as-number
| | +--rw description?                 string
| | +--rw ebgp-multihop?
| | | +--rw ttl?                       uint8
| | +--rw fall-over?                   boolean
| | +--rw send-community?
| | | +--rw community-type?            enumeration
| | +--rw shutdown?                    boolean
| | +--rw update-source-loopback?      uint16
| | +--rw weight?                      uint16
| | +--rw timers?
| | | +--rw keepalive?                 dell:seconds
| | | +--rw hold-time?                 dell:seconds
| | +--rw add-path?
| | | +--rw addpath?                   enumeration
| | | +--rw path-count?                uint8
| | +--rw local-as?
| | | +--rw as-number?                 dell:as-number
| | | +--rw no-prepend?                boolean
| | +--rw password?
| | | +--rw encryption-type?           enumeration
| | | +--rw password-value?            string
| | +--rw activate?                    boolean
| | +--rw advertisement-interval?      dell:seconds
| | +--rw allowas-in?                  uint8
| | +--rw default-originate?
| | | +--rw route-map?                 dell:route-map-name
| | +--rw distribute-list [direction]
| | | +--rw direction                  dell:bgp-opt-dir
| | | +--rw name?                      dell:prefix-list-name
| | +--rw filter-list [direction]
| | | +--rw direction                  dell:bgp-opt-dir
| | | +--rw name?                      dell:aspath-access-list-name
| | +--rw maximum-prefix?
| | | +--rw maximum?                   uint32
| | | +--rw threshold?                 uint8
| | | +--rw warning-only?              boolean
| | +--rw next-hop-self?                boolean
| | +--rw remove-private-as?            boolean
| | +--rw route-map [direction]
| | | +--rw direction                  dell:bgp-opt-dir
| | | +--rw name?                      dell:route-map-name
| | +--rw route-reflector-client?      boolean
| | +--rw sender-side-loop-detection?  boolean
| | +--rw soft-reconfiguration?        enumeration
| | +--rw dmzlink-bw?                  boolean
| | +--rw advertisement-start?          dell:seconds
|--rw aggregate-address [ip-address]
| | +--rw ip-address                    union
| | +--rw advertise-map?                dell:route-map-name
| | +--rw as-set?                       boolean
| | +--rw attribute-map?                 dell:route-map-name
| | +--rw summary-only?                 boolean
| | +--rw suppress-map?                 dell:route-map-name
|--rw distance?
| | +--rw bgp?

```

```

|         +---rw external-distance?   uint8
|         +---rw internal-distance?   uint8
|         +---rw local-distance?      uint8
+---rw redistribute?
|   +---rw connected?
|   |   +---rw route-map?             dell:route-map-name
|   +---rw static?
|   |   +---rw route-map?             dell:route-map-name
+---rw imported-bgp?
|   |   +---rw route-map?             dell:route-map-name
+---rw ospf [process-id]
|   |   +---rw process-id             uint16
|   |   +---rw match?
|   |   |   +---rw external1?         boolean
|   |   |   +---rw external2?         boolean
|   |   |   +---rw internal?          boolean
|   |   +---rw route-map?             dell:route-map-name
+---rw isis [tag]
|   +---rw tag                        string
|   +---rw level?                     enumeration
|   +---rw metric?                    uint32
|   +---rw route-map?                 dell:route-map-name
+---rw address-family [family-type vrf]
|   +---rw family-type                enumeration
|   +---rw vrf                        dell:vrf-name
+---rw bgp?
|   +---rw asnotation?                enumeration
|   +---rw always-compare-med?         boolean
|   +---rw bestpath?
|   |   +---rw as-path?                enumeration
|   |   +---rw med?
|   |   |   +---rw confed?             boolean
|   |   |   +---rw missing-as-best?    boolean
|   |   |   +---rw router-id-ignore?    boolean
+---rw client-to-client?
|   |   +---rw reflection?             boolean
+---rw cluster-id?                    union
+---rw confederation?
|   |   +---rw identifier?             dell:as-number
|   |   +---rw peers*                  dell:as-number
+---rw default?
|   |   +---rw local-preference?       uint32
+---rw dmzlink-bw?                     boolean
+---rw enforce-first-as?                boolean
+---rw fast-external-fallover?          boolean
+---rw four-octet-as-support?           boolean
+---rw graceful-restart?
|   |   +---rw restart-time?           uint16
|   |   +---rw stale-path-time?       uint16
|   |   +---rw role?                   enumeration
+---rw log-neighbor-changes?            boolean
+---rw non-deterministic-med?           boolean
+---rw recursive-bgp-next-hop?          boolean
+---rw regex-eval-optz-disable?         boolean
+---rw connection-retry-timer?          dell:seconds
+---rw router-id?                       dell:ip-address
+---rw add-path?
|   |   +---rw addpath?                enumeration
|   |   +---rw path-count?             uint8
|   |   +---rw enable?                 boolean
+---rw dampening?
|   |   +---rw half-life?               uint32
|   |   +---rw reuse?                  uint32
|   |   +---rw suppress?               uint32

```

```

|         +--rw max-suppress-time?   uint32
|         +--rw route-map?           dell:route-map-name
+--rw network [ipv4-address]
|   +--rw ipv4-address               dell:ipv4-address-with-prefix
|   +--rw route-map?                 dell:route-map-name
|   +--rw backdoor?                  boolean
+--rw network-v6 [ipv6-address]
|   +--rw ipv6-address               dell:ipv6-address
|   +--rw route-map?                 dell:route-map-name
|   +--rw backdoor?                  boolean
+--rw aggregate-address [ip-address]
|   +--rw ip-address                 union
|   +--rw advertise-map?             dell:route-map-name
|   +--rw as-set?                    boolean
|   +--rw attribute-map?             dell:route-map-name
|   +--rw summary-only?              boolean
|   +--rw suppress-map?              dell:route-map-name
+--rw distance?
|   +--rw bgp?
|     +--rw external-distance?      uint8
|     +--rw internal-distance?      uint8
|     +--rw local-distance?         uint8
+--rw redistribute?
|   +--rw connected?
|     +--rw route-map?              dell:route-map-name
|   +--rw static?
|     +--rw route-map?              dell:route-map-name
|   +--rw imported-bgp?
|     +--rw route-map?              dell:route-map-name
|   +--rw ospf [process-id]
|     +--rw process-id              uint16
|     +--rw match?
|       +--rw external1?            boolean
|       +--rw external2?            boolean
|       +--rw internal?              boolean
|       +--rw route-map?            dell:route-map-name
|   +--rw isis [tag]
|     +--rw tag                      string
|     +--rw level?                   enumeration
|     +--rw metric?                  uint32
|     +--rw route-map?              dell:route-map-name
+--rw default-metric?                uint32
+--rw timers?
|   +--rw bgp
|     +--rw keepalive?              dell:seconds
|     +--rw hold-time?              dell:seconds
+--rw maximum-paths?
|   +--rw ebgp?                      uint32
|   +--rw ibgp?                      uint32
+--rw peer-group [peer-group-name]
|   +--rw peer-group-name            dell:peer-group-name
|   +--rw remote-as?                 dell:as-number
|   +--rw description?               string
|   +--rw ebgp-multihop?
|     +--rw ttl?                     uint8
|   +--rw fall-over?                 boolean
|   +--rw send-community?
|     +--rw community-type?          enumeration
|   +--rw shutdown?                  boolean
|   +--rw update-source-loopback?    uint16
|   +--rw weight?                    uint16
+--rw timers?
|   +--rw keepalive?                 dell:seconds
|   +--rw hold-time?                 dell:seconds

```

```

|     +---rw add-path?
|     |   +---rw addpath?      enumeration
|     |   +---rw path-count?   uint8
|     +---rw local-as?
|     |   +---rw as-number?    dell:as-number
|     |   +---rw no-prepend?   boolean
|     +---rw password?
|     |   +---rw encryption-type? enumeration
|     |   +---rw password-value? string
|     +---rw activate?          boolean
|     +---rw advertisement-interval? dell:seconds
|     +---rw allowas-in?        uint8
|     +---rw default-originate?
|     |   +---rw route-map?    dell:route-map-name
|     +---rw distribute-list [direction]
|     |   +---rw direction     dell:bgp-opt-dir
|     |   +---rw name?         dell:prefix-list-name
|     +---rw filter-list [direction]
|     |   +---rw direction     dell:bgp-opt-dir
|     |   +---rw name?         dell:aspath-access-list-name
|     +---rw maximum-prefix?
|     |   +---rw maximum?      uint32
|     |   +---rw threshold?    uint8
|     |   +---rw warning-only? boolean
|     +---rw next-hop-self?     boolean
|     +---rw remove-private-as? boolean
|     +---rw route-map [direction]
|     |   +---rw direction     dell:bgp-opt-dir
|     |   +---rw name?         dell:route-map-name
|     +---rw route-reflector-client? boolean
|     +---rw sender-side-loop-detection? boolean
|     +---rw soft-reconfiguration? enumeration
|     +---rw dmzlink-bw?        boolean
+---rw neighbor [ip-address]
|     +---rw ip-address          union
|     +---rw peer-group-name?    dell:peer-group-name
|     +---rw remote-as?          dell:as-number
|     +---rw description?        string
|     +---rw ebgp-multihop?
|     |   +---rw ttl?           uint8
|     +---rw fall-over?          boolean
|     +---rw send-community?
|     |   +---rw community-type? enumeration
|     +---rw shutdown?          boolean
|     +---rw update-source-loopback? uint16
|     +---rw weight?            uint16
|     +---rw timers?
|     |   +---rw keepalive?     dell:seconds
|     |   +---rw hold-time?     dell:seconds
|     +---rw add-path?
|     |   +---rw addpath?      enumeration
|     |   +---rw path-count?   uint8
|     +---rw local-as?
|     |   +---rw as-number?    dell:as-number
|     |   +---rw no-prepend?   boolean
|     +---rw password?
|     |   +---rw encryption-type? enumeration
|     |   +---rw password-value? string
|     +---rw activate?          boolean
|     +---rw advertisement-interval? dell:seconds
|     +---rw allowas-in?        uint8
|     +---rw default-originate?
|     |   +---rw route-map?    dell:route-map-name
+---rw distribute-list [direction]

```

```

|         | +--rw direction      dell:bgp-opt-dir
|         | +--rw name?         dell:prefix-list-name
|         +--rw filter-list [direction]
|         | +--rw direction    dell:bgp-opt-dir
|         | +--rw name?       dell:aspath-access-list-name
|         +--rw maximum-prefix?
|         | +--rw maximum?     uint32
|         | +--rw threshold?   uint8
|         | +--rw warning-only? boolean
|         +--rw next-hop-self? boolean
|         +--rw remove-private-as? boolean
|         +--rw route-map [direction]
|         | +--rw direction    dell:bgp-opt-dir
|         | +--rw name?       dell:route-map-name
|         +--rw route-reflector-client? boolean
|         +--rw sender-side-loop-detection? boolean
|         +--rw soft-reconfiguration? enumeration
|         +--rw dmzlink-bw?    boolean
|         +--rw advertisement-start? dell:seconds
+--rw management
+--rw ip
+--rw ipv6

```

Global Configuration

The following definition displays the global configuration entries:

```

Module:          SYSTEM
URI:            /api/running/dell/system
Methods         GET, POST, PUT, PATCH, and DELETE
Supported:

```

Global Configuration Model


```

+--rw dell
+--rw system
| +--rw hostname?      string
| +--rw banner?
| | +--rw login?      string
| +--rw credentials?
| | +--rw username?   string
| | +--rw password?   string
| | +--rw privilege?  uint32
| +--rw default-vlan?
| +--rw vlan-id?      vlan-id-num
+--rw ip
| +--rw dhcp?
| | +--rw server?     boolean
| | +--rw relay?
| | +--rw info-option? boolean
+--rw helper-address?
| +--rw hop-count?    boolean
| +--rw multicast-map?
| +--rw ip-address     ip-address
| +--rw bcast-address  ip-address

```

Interface

The following definition configures and displays the properties of an Interface:

| | |
|-------------------|---|
| Module: | INTERFACE |
| URI: | /api/running/dell/interfaces/interface |
| Methods | GET, POST, PUT, PATCH, and DELETE |
| Supported: |  NOTE: The POST and DELETE operations are not supported on Physical Interfaces. |

Use the following formats for various interface types:

1. **Physical Interface:** <iftype>-<slot>-<port>
 - The <iftypes> can be TenGigabitEthernet or FortyGigabitEthernet.
 - Example: tengig-0-0, fortygig-0-48
2. **Logical Interface:** <iftype>-<number>
 - The <iftype> can be vlan, port-channel, loopback, or management.
 - **VLAN:** vlan-<number>
Example: vlan-10, vlan-1024
 - **Port-channel:** portchan-<number>
Example: portchan-10
 - **Loopback:** loopback-<number>
Example: loopback-40
 - **Management:** mgmt-<slot>-<port>
Example: mgmt-5-0

Interface Configuration Model

```
+--rw dell!  
  +--rw interfaces  
    | +--rw interface [name]  
    |   +--rw name                               dell:interface-name  
    |   +--rw flowcontrol?  
    |   | +--rw rx?   enumeration  
    |   | +--rw tx?   enumeration  
    |   +--rw portmode?  
    |   | +--rw hybrid?  boolean  
    |   +--rw switchport?  
    |   | +--rw backup?  
    |   | | +--rw interface  
    |   | |   +--rw name      dell:backup-tag-interface  
    |   | +--rw mode?  
    |   |   +--rw private-vlan?  private-vlan-mode  
    |   +--rw mac?  
    |   | +--rw access-group?  
    |   | | +--rw in?  
    |   | | | +--rw WORD      access-list-name  
    |   | | | +--rw vlan*     vlan-range  
    |   | +--rw out?
```

```

| | | +--rw WORD access-list-name
| | +--rw learning-limit?
| | | +--rw count? uint32
| | | +--rw (operations)?
| | | | +--:(sticky)
| | | | | +--rw sticky? boolean
| | | | +--:(non-sticky)
| | | | | +--rw dynamic? boolean
| | | | | +--rw station-move? boolean
| | +--rw learn-limit-violation?
| | | +--rw (action)?
| | | | +--:(log)
| | | | | +--rw log boolean
| | | | +--:(shutdown)
| | | | | +--rw shutdown boolean
| | +--rw station-move-violation?
| | | +--rw (action)?
| | | | +--:(log)
| | | | | +--rw log boolean
| | | | +--:(shutdown-both)
| | | | | +--rw shutdown-both boolean
| | | | +--:(shutdown-offending)
| | | | | +--rw shutdown-offending boolean
| | | | +--:(shutdown-original)
| | | | | +--rw shutdown-original boolean
| +--rw channel-members [name]
| | +--rw name dell:phy-intf-name
| +--rw min-links? uint32
| +--rw lacp?
| | +--rw timeout? enumeration
| | +--rw fast-switchover? boolean
| +--rw vlan-name? vlan-name
| +--rw mode?
| | +--rw remote-port-mirroring? empty
| +--rw tagged [name]
| | +--rw name dell:backup-tag-interface
| +--rw untagged [name]
| | +--rw name dell:backup-tag-interface
| +--rw protocol?
| | +--rw lldp?
| | | +--rw mode? protocol-lldp-mode
| +--rw description? string
| +--rw mtu? uint32
| +--rw shutdown? boolean
| +--rw speed? enumeration
| +--rw duplex? intf-duplex-mode
| +--rw negotiation?
| | +--rw auto? boolean
| +--rw port-channel-protocol?
| | +--rw lacp?
| | | +--rw port-channel? port-channel
| | | +--rw mode? enumeration
| | | +--rw port-priority? uint32
| +--rw rate-interval? uint32
| | +--rw ip
| | | +--rw vrf?
| | | | +--rw forwarding?
| | | | | +--rw (vrf-domain)?
| | | | | | +--:(management)
| | | | | | | +--rw management? empty
| | | | | | +--:(user-defined-vrf)
| | | | | | | +--rw name? vrf-name
| | +--rw address?
| | | +--rw primary?

```

```

| | | +--rw address? dell:ipv4-address-with-prefix
| | +--rw dhcp?
| | | +--rw relay?
| | | | +--rw information-option?
| | | | | +--rw remote-id?
| | | | | +--rw (remote-id-string)?
| | | | | | +--:(hostname)
| | | | | | | +--rw hostname? boolean
| | | | | | | +--:(mac)
| | | | | | | | +--rw mac? boolean
| | | | | | | | +--:(user-string)
| | | | | | | | | +--rw user-string? string
| | | | +--rw vendor-class-identifier?
| | | | | +--rw distinct-string? string
| | +--rw secondary [address]
| | | +--rw address dell:ipv4-address-with-prefix
+--rw access-group?
+--rw in?
| | +--rw acl-name access-list-name
| | +--rw implicit-permit? boolean
| | +--rw (ip-action)?
| | | +--:(vlan)
| | | | +--rw vlan* vlan-range
| | | | +--:(optimized)
| | | | | +--rw optimized? boolean
| | | | +--:(vrf)
| | | | | +--rw vrf? dell:vrf-name
+--rw out?
| | +--rw acl-name access-list-name
| | +--rw implicit-permit? boolean
| | +--rw (ip-action)?
| | | +--:(vlan)
| | | | +--rw vlan* vlan-range
| | | | +--:(optimized)
| | | | | +--rw optimized? boolean
| | | | +--:(vrf)
| | | | | +--rw vrf? dell:vrf-name
+--rw redirect-group [name]
| | +--rw name dell:redirect-list-name
+--rw helper-address [ip-address]
+--rw ip-address dell:ip-address
+--rw (vrf-type)?
+--:(default-vrf)
| | +--rw default-vrf? boolean
+--:(user-defined-vrf)
| | +--rw vrf? vrf-name
+--rw ipv6?
+--rw address [ipv6-address]
| | +--rw ipv6-address dell:ipv6-address
| | +--rw eui64? boolean
+--rw nd?
| | +--rw dad?
| | | +--rw attempts? uint32
+--rw hop-limit? uint32
+--rw managed-config-flag? boolean
+--rw ra-interval?
| | +--rw max-ra-interval uint32
| | +--rw min-ra-interval uint32
+--rw mtu? uint32
+--rw dhcp-non-address? boolean
+--rw prefix [prefix-name]
| | +--rw prefix-name dell:ipv6-nd-prefix-name
| | +--rw advertise? boolean
| | +--rw autoconfig? boolean

```

```

|         | | +--rw rtr-address?    boolean
|         | | +--rw off-link?      boolean
|         | | +--rw lifetime?
|         | |   +--rw valid        ipv6-nd-prefix-lifetime
|         | |   +--rw preferred    ipv6-nd-prefix-lifetime
|         | +--rw ra-lifetime?     uint32
|         | +--rw reachable-time?  uint32
|         | +--rw retrans-timer?   uint32
|         | +--rw suppress-ra?     boolean
| +--rw access-group?
|   +--rw in?
|     | +--rw WORD                access-list-name
|     | +--rw implicit-permit?    boolean
|     | +--rw (ipv6-action)?
|     |   +--:(vlan)
|     |   | +--rw vlan*           vlan-range
|     |   +--:(optimized)
|     |   +--rw optimized?       boolean
|   +--rw out?
|     | +--rw WORD                access-list-name
|     | +--rw implicit-permit?    boolean
|     | +--rw (ipv6-action)?
|     |   +--:(vlan)
|     |   | +--rw vlan*           vlan-range
|     |   +--:(optimized)
|     |   +--rw optimized?       boolean

```

IPv6 Global Configurations

The following definition displays the entries of IPv6 global configurations:

Module: IPv6
URI: /api/operational/dell/ipv6
Methods: GET, POST, PUT, PATCH, and DELETE
Supported:

IPv6 Global Configuration Model

```

+--rw ipv6
  +--rw route* [ipv6-address-with-prefix exit-interface next-hop]
  | +--rw ipv6-address-with-prefix  dell:ipv6-address
  | +--rw exit-interface            union
  | +--rw next-hop                  union
  | +--rw metric?                   uint32
  | +--rw permanent?                boolean
  | +--rw tag?                       uint32
  +--rw unicast-routing?            boolean
  +--rw unknown-unicast?            boolean
  +--rw flowlabel-zero?             boolean
  +--rw host* [hostname]
  | +--rw hostname                  dell:host-name
  | +--rw ipv6-address?             dell:ipv6-address
  +--rw neighbor* [ipv6-address interface]
  | +--rw ipv6-address              dell:ipv6-address
  | +--rw interface                 dell:interface-name
  | +--rw mac-address               dell:mac-address
  +--rw icmp!
  | +--rw source-interface?         dell:interface-name-icmp
  +--rw prefix-list* [name]
  |   +--rw name                    dell:prefix-list-name

```

```

+--rw description?  string
+--rw seq* [seq-num]
    +--rw seq-num      uint32
    +--rw action        ipv6-prefix-list-action-types
    +--rw ipv6-prefix   union
    +--rw prefix-min-len?  uint32
    +--rw prefix-max-len?  uint32

```

Management-Route

The following definition configures and displays the properties of a Management-Route:

```

Module:      MANAGEMENT-ROUTE
URI:         /api/running/dell/management/route
Methods     GET, POST, PUT, PATCH, and DELETE
Supported:

```

Management Route Configuration Model

```

+--rw management
  +--rw route* [ip-address-prefix]
    +--rw ip-address-prefix  union
    +--rw (exit-value)?
      +---:(exit-interface)
        | +--rw exit-interface?  dell:mgmt-route-intf-types
        +---:(next-hop)
          +--rw next-hop?        inet:ip-address

```

Policy Based Routing (PBR)

Use the following definition to configure and display the properties of PBR:

```

Module:      PBR
URI:         /api/running/dell/redirect-list
Methods     GET, POST, PUT, PATCH, and DELETE
Supported:

```

PBR Configuration Model

```

+--rw dell
  +--rw system
  +--rw interfaces
  +--rw router
  +--rw management
  +--rw ip
    | +--rw redirect-list [name]
    |   +--rw name          string
    |   +--rw description?  string
    |   +--rw rules-list [seq]
    |     +--rw seq          uint32
    |     +--rw output-action  enumeration
    |     +--rw next-hop      union
    |     +--rw track-num?    uint32
    |     +--rw protocol      ip-protocols-type
    |     +--rw source         ipv4-address-any
    |     +--rw source-mask    dell:ipv4-mask-or-prefix
    |     +--rw destination    ipv4-address-any

```

```

|         +---rw destination-mask          dell:ipv4-mask-or-prefix
|         +---rw tcp-options*             enumeration
|         +---rw tcp-udp-port-oper-list [port-type]
|         +---rw port-type                tcp-udp-port-type
|         +---rw oper-type                 tcp-udp-port-oper-type
|         +---rw port                      uint32
|         +---rw start-port                uint32
|         +---rw end-port                  uint32
+---rw ipv6
+---ro stats
  +---ro access-list
    +---ro mac [name]
      | +---ro name                        access-list-name
      | +---ro type?                       access-list-types
      | +---ro mac-acl-filter [seq-no]
      |   +---ro seq-no                    uint32
      |   +---ro filter-type?              acl-rule-types
      |   +---ro src-mac-filter?           acl-incoming-address-filter-
types
      |   +---ro src-mac-address?          yang:phys-address
      |   +---ro src-mac-mask?             yang:phys-address
      |   +---ro dest-mac-filter?          acl-incoming-address-filter-
types
      |   +---ro dest-mac-address?         yang:phys-address
      |   +---ro dest-mac-mask?            yang:phys-address
      |   +---ro encapsulation-type?       mac-acl-encap-types
      |   +---ro encapsulation-operator?   mac-acl-encap-operators
      |   +---ro encapsulation-protocol?   string
      |   +---ro count-type?               acl-rule-count-types
      |   +---ro egress-count-value?       uint32
      |   +---ro ingress-count-value?     uint32
      |   +---ro log?                      boolean
      |   +---ro monitor?                  boolean
      |   +---ro order?                    uint32
    +---ro ip [name]
      +---ro name                          access-list-name
      +---ro type?                          access-list-types
      +---ro ip-acl-filter [seq-no]
        +---ro seq-no                       uint32
        +---ro filter-type?                 acl-rule-types
        +---ro protocol?                    ip-protocols
        +---ro ip-protocol-number?          uint32
        +---ro src-ip-filter?               acl-incoming-address-filter-types
        +---ro source-ip-address?           inet:ipv4-address
        +---ro source-mask?                 inet:ipv4-address
        +---ro src-operator?                ip-acl-operators
        +---ro srcport?                     uint32
        +---ro srcportrange?                layer4-port-range
        +---ro dest-ip-filter?              acl-incoming-address-filter-types
        +---ro dest-ip-address?             inet:ipv4-address
        +---ro destination-mask?            inet:ipv4-address
        +---ro dst-operator?                ip-acl-operators
        +---ro dstport?                     uint32
        +---ro dstportrange?                layer4-port-range
        +---ro ack?                         boolean
        +---ro fin?                         boolean
        +---ro psh?                         boolean
        +---ro rst?                         boolean
        +---ro syn?                         boolean
        +---ro urg?                         boolean
        +---ro count-type?                  acl-rule-count-types
        +---ro egress-count-value?          uint32
        +---ro ingress-count-value?        uint32
        +---ro log?                         boolean

```

```

+---ro monitor?          boolean
+---ro order?           uint32
+---ro dscp?            uint32
+---ro ecn?             uint32
+---ro fragments?      boolean

```

Port Mirroring

The following definition is for configuring and displaying the properties of port mirroring:

Module: Port Mirroring
URI: /api/running/dell/monitor-session
Methods GET, POST, PUT, PATCH, and DELETE
Supported:

Port Mirroring Configuration Model

```

+---rw dell
  +---rw system
  +---rw interfaces
  +---rw router
  +---rw management
  +---rw ip
  +---rw ipv6
  +---rw monitor-session [session-id]
    +---rw session-id      uint32
    +---rw type?           enumeration
    +---rw description?    string
    +---rw source-list [interface-name]
      | +---rw interface-name      dell:interface-name
      | +---rw direction?          enumeration
      | +---rw destination-interface? dell:interface-name
      | +---rw rspan-intf-type?     enumeration
      | +---rw rspan-intf-tagged?   boolean
      | +---rw rspan-remote-vlan?   dell:vlan-id-num
    +---rw erspan-params?
      | +---rw source-ip?          dell:ipv4-no-prefix
      | +---rw destination-ip?     dell:ipv4-no-prefix
    +---rw disable?          boolean
    +---rw flow-based?       boolean

```

Route Maps

The following definition configures and displays the properties of a Route Map:

Module: ROUTE MAP
URI: /api/running/dell/interfaces/route-map
Methods GET, POST, PUT, PATCH, and DELETE
Supported:

Route Map Configuration Model

```

+---rw route-map* [map-name seq]
  +---rw map-name      string
  +---rw seq           union
  +---rw (entry)?
  | +---: (permit)

```

```

| | +--rw permit?      boolean
| | +--:(deny)
| | +--rw deny?       boolean
+--rw set
    +--rw ext-community!
    +--rw bandwidth?   uint32

```

Static Route

The following definition configures and displays the properties of a Static Route configuration:

Module: IP STATIC ROUTE
URI: /api/running/dell/ip/route
Methods: GET, POST, PUT, PATCH, and DELETE
Supported:

Static Route Configuration Model

```

+--rw ip
  +--rw route* [ip-address-prefix interface-name next-hop]
  | +--rw ip-address-prefix    dell:ipv4-address-with-prefix
  | +--rw interface-name      union
  | +--rw next-hop            union
  | +--rw weight?             uint32
  | +--rw metric?             uint32
  | +--rw permanent?         boolean
  | +--rw tag?                 uint32
+--rw ecmp!
  +--rw weighted?            boolean

```

Weighted ECMP

The following definition configures and displays the properties of Weighted ECMP:

Module: WEIGHTED ECMP
URI: /api/running/dell/ip/ecmp
Methods: GET, PATCH, PUT, POST, and DELETE
Supported:

Weighted ECMP Model

```

+--rw ip
  | +--rw ecmp!
  | +--rw weighted?    boolean

```

Operational

This section describes the properties of all the IPv4 operational data.

BGP

Use the following definition to get the BGP statistics:

Module: BGP

URI: /api/operational/dell/stats/ip/bgp

Methods GET

Supported:

BGP Model

```
+--ro bgp
  +--ro table-version?    uint32
  +--ro local-router-id?  string
  +--ro routes
    | +--ro prefix-list* [network-prefix network-prefix-len next-hop-
address seq-num]
    |   +--ro network-prefix      inet:ip-address
    |   +--ro network-prefix-len  inet-
address:InetAddressPrefixLength
    |   +--ro next-hop-address    inet:ip-address
    |   +--ro seq-num             uint32
    |   +--ro metric?             uint32
    |   +--ro local-pref?         uint32
    |   +--ro weight?             uint32
    |   +--ro next-hop-cost-index? uint32
    |   +--ro as-path-string?    string
    |   +--ro path-source?       enumeration
    |   +--ro origin-code?       enumeration
    |   +--ro is-nlre-loc-aggtd?  snmpv2-tc:TruthValue
    |   +--ro is-stale?           snmpv2-tc:TruthValue
    |   +--ro status-code?       enumeration
    |   +--ro best-route?        snmpv2-tc:TruthValue
    |   +--ro link-bandwidth?    string
  +--ro neighbors
    +--ro neighbor* [neighbor-address]
    +--ro neighbor-address    inet:ipv4-address
    +--ro advertised-routes
    | +--ro prefix-list* [network-prefix network-prefix-len next-
hop-address seq-num]
    |   +--ro network-prefix      inet:ip-address
    |   +--ro network-prefix-len  inet-
address:InetAddressPrefixLength
    |   +--ro next-hop-address    inet:ip-address
    |   +--ro seq-num             uint32
    |   +--ro metric?             uint32
    |   +--ro local-pref?         uint32
    |   +--ro weight?             uint32
    |   +--ro next-hop-cost-index? uint32
    |   +--ro as-path-string?    string
    |   +--ro path-source?       enumeration
    |   +--ro origin-code?       enumeration
    |   +--ro is-nlre-loc-aggtd?  snmpv2-tc:TruthValue
    |   +--ro is-stale?           snmpv2-tc:TruthValue
    |   +--ro status-code?       enumeration
    |   +--ro best-route?        snmpv2-tc:TruthValue
    |   +--ro link-bandwidth?    string
  +--ro received-routes
    +--ro prefix-list* [network-prefix network-prefix-len next-
hop-address seq-num]
    +--ro network-prefix      inet:ip-address
    +--ro network-prefix-len  inet-
address:InetAddressPrefixLength
    +--ro next-hop-address    inet:ip-address
    +--ro seq-num             uint32
    +--ro metric?             uint32
    +--ro local-pref?         uint32
```

```

+--ro weight?                uint32
+--ro next-hop-cost-index?   uint32
+--ro as-path-string?       string
+--ro path-source?          enumeration
+--ro origin-code?          enumeration
+--ro is-nlre-loc-aggtd?    snmpv2-tc:TruthValue
+--ro is-stale?              snmpv2-tc:TruthValue
+--ro status-code?          enumeration
+--ro best-route?           snmpv2-tc:TruthValue
+--ro link-bandwidth?       string

```

BGPv6

Use the following definition to get the BGPv6 operational data:

```

Module:      BGPv6
URI:         /api/operational/dell/stats/ip/bgpv6
Methods     GET
Supported:

```

BGPv6 Operational Model

```

+--rw dell
  +--ro stats
    +--ro ip
      +--ro bgpv6
        +--ro neighbors [neighbor-address vrf-id]
          | +--ro neighbor-address                dell:ipv6-address-
          | +--ro vrf-id                          uint32
          | +--ro remote-link?                    enumeration
          | +--ro router-id?                      dell:ipv6-address-
          | +--ro state?                          enumeration
          | +--ro state-duration?                 hour-minute-second
          | +--ro last-read?                      hour-minute-second
          | +--ro last-write?                    hour-minute-second
          | +--ro hold-time?                      uint32
          | +--ro keep-alive-interval?            uint32
          | +--ro stale-path-time?               uint32
          | +--ro config-hold-time?              uint32
          | +--ro config-keep-alive-interval?    uint32
          | +--ro recv-messages?                 uint32
          | +--ro recv-in-queue?                 uint32
          | +--ro recv-opens?                    uint32
          | +--ro recv-notifications?            uint32
          | +--ro recv-updates?                  uint32
          | +--ro recv-keepalives?              uint32
          | +--ro recv-route-refresh-requests?   uint32
          | +--ro sent-messages?                 uint32
          | +--ro sent-in-queue?                uint32
          | +--ro sent-opens?                    uint32
          | +--ro sent-notifications?           uint32
          | +--ro sent-updates?                  uint32
          | +--ro sent-keepalives?              uint32
          | +--ro sent-route-refresh-requests?   uint32
          | +--ro route-refresh-rcvd?            uint32
          | +--ro route-refresh-sent-messages?   uint32
          | +--ro min-time-bw-advrt-runs?        uint32
          | +--ro min-time-b4-advrt-start?       uint32

```

| | | |
|--|---|--------------------|
| | +--ro conn-established? | uint32 |
| | +--ro conn-dropped? | uint32 |
| | +--ro last-connstr? | string |
| | +--ro last-reset? | hour-minute-second |
| | +--ro last-reset-never? | boolean |
| | +--ro notification-history? | string |
| | +--ro addr-type-receive-nbr? | string |
| | +--ro recv-multi-proto-ext? | string |
| | +--ro recv-route-refresh? | string |
| | +--ro recv-graceful-restart? | string |
| | +--ro recv-cisco-route-refresh? | string |
| | +--ro recv-four-byte-octet-as? | string |
| | +--ro recv-add-path? | string |
| | +--ro addr-type-adv-nbr? | string |
| | +--ro adv-multi-proto-ext? | string |
| | +--ro adv-route-refresh? | string |
| | +--ro adv-graceful-restart? | string |
| | +--ro adv-cisco-route-refresh? | string |
| | +--ro adv-four-byte-octet-as? | string |
| | +--ro adv-add-path? | string |
| | +--ro member-of-peer-grp? | string |
| | +--ro peer-admin-status? | enumeration |
| | +--ro peer-fall-over? | enumeration |
| | +--ro bfd-enable-mode? | enumeration |
| | +--ro update-sourceset? | string |
| | +--ro peer-allow-aslimit? | uint32 |
| | +--ro peer-act-in-grp-out-bound-opt? | string |
| | +--ro max-prefix-set? | uint32 |
| | +--ro with-threshold? | uint32 |
| | +--ro config-drop-warn? | enumeration |
| | +--ro ttl? | uint32 |
| | +--ro ebgp-multi-hop-enable? | boolean |
| | +--ro next-hop-set-self? | string |
| | +--ro localpref? | uint32 |
| | +--ro num-stale-paths-left? | uint32 |
| | +--ro remote-as? | string |
| | +--ro local-as? | string |
| | +--ro noprepnd? | uint32 |
| | +--ro local-as-prepend? | uint32 |
| | +--ro neighbor-under-common-admin? | string |
| | +--ro version? | uint32 |
| | +--ro description? | string |
| | +--ro address-family? | string |
| | +--ro local-added-list? | uint32 |
| | +--ro local-replaced-list? | uint32 |
| | +--ro local-removed-list? | uint32 |
| | +--ro inq-added-list? | uint32 |
| | +--ro inq-replaced-list? | uint32 |
| | +--ro inq-removed-list? | uint32 |
| | +--ro outq-added-list? | uint32 |
| | +--ro outq-removed-list? | uint32 |
| | +--ro accept-prefix? | uint32 |
| | +--ro advertised-prefix? | uint32 |
| | +--ro prefix-denied? | uint32 |
| | +--ro addr-withdrawn-from-peer? | uint32 |
| | +--ro addr-withdrawn-by-peer? | uint32 |
| | +--ro martian-addr-ignored? | boolean |
| | +--ro restart-time-adver-peer? | uint32 |
| | +--ro restart-time-rece-peer? | uint32 |
| | +--ro stale-path-time-config-peer? | uint32 |
| | +--ro graceful-restart-role-recv-only? | string |
| | +--ro graceful-restart-role-adver-only? | string |
| | +--ro restart-time-adv? | uint32 |
| | +--ro restart-time-rcv? | uint32 |

```

| +--ro route-reflector-client?          string
| +--ro soft-reconfig-inbound?          boolean
| +--ro session-md5-auth?                string
| +--ro default-org-routemap?            string
| +--ro default-org-config?              string
| +--ro no-active-tcp-connection?        string
| +--ro pdu-cap-dir?                      string
| +--ro pdu-cap-stats?                   string
| +--ro local-host?                       dell:ipv6-address-
with-prefix
| +--ro local-port?                       uint32
| +--ro foreign-host?                     dell:ipv6-address-
with-prefix
| +--ro foreign-port?                     uint32
+--ro summary
| +--ro local-router-id?                  dell:ipv6-address-
with-prefix
| +--ro local-as?                         string
| +--ro neighbor-count?                   uint32
| +--ro nlri-count?                       uint32
| +--ro path-count?                       uint32
| +--ro penalized-paths?                  uint32
| +--ro denied-paths?                     uint32
| +--ro denied-paths-memory?              uint32
| +--ro nlri-memory?                       uint32
| +--ro path-memory?                       uint32
| +--ro total-memory?                      uint32
| +--ro path-attribute?                   uint32
| +--ro path-attribute-memory?            uint32
| +--ro community?                        uint32
| +--ro community-memory?                 uint32
| +--ro external-community?               uint32
| +--ro external-community-memory?        uint32
| +--ro as-path?                          uint32
| +--ro as-path-memory?                   uint32
| +--ro rrcluster?                        uint32
| +--ro rrcluster-memory?                 uint32
| +--ro optional-transattribute?          uint32
| +--ro optional-transattribute-memory?   uint32
| +--ro history-paths?                    uint32
| +--ro dampened-paths?                   uint32
| +--ro memory?                           uint32
| +--ro dampening-enabled?                uint32
| +--ro bfd-enabled?                      uint32
| +--ro bfd-interval?                     uint32
| +--ro bfd-minimum-rx?                   uint32
| +--ro bfd-multiplier?                   uint32
| +--ro bfd-role?                         enumeration
| +--ro local-rib
| | +--ro routes-added?                    uint32
| | +--ro routes-replaced?                 uint32
| | +--ro routes-withdrawn?               uint32
| | +--ro router-id?                       dell:ipv6-address-with-prefix
+--ro neighbor [neighbor-address]
|   +--ro neighbor-address                 dell:ipv6-address-with-prefix
|   +--ro remote-as?                       string
|   +--ro msg-rcvd?                         uint32
|   +--ro msg-sent?                         uint32
|   +--ro table-version?                   uint32
|   +--ro in-queue?                         string
|   +--ro out-queue?                       string
|   +--ro up-time?                         string
|   +--ro state-prefix?                    string
+--ro unicast

```

```

|   +---ro local-rib
|   |   +---ro routes-added?      uint32
|   |   +---ro routes-replaced?   uint32
|   |   +---ro routes-withdrawn?  uint32
|   |   +---ro router-id?         dell:ipv6-address-with-prefix
| +---ro prefix-list [network-prefix next-hop-addr]
|   +---ro network-prefix         dell:ipv6-address-with-prefix
|   +---ro next-hop-addr         dell:ipv6-address-with-prefix
|   +---ro status-code?          bgp-status-code
|   +---ro best-route?           boolean
|   +---ro path-source?          bgp-path-source
|   +---ro metric?               string
|   +---ro local-pref?           string
|   +---ro weight?               uint32
|   +---ro as-path-string?       string
|   +---ro origin-code?          bgp-origin-code
+---ro peer-group [peer-group-name]
|   +---ro peer-group-name        string
|   +---ro version?               uint32
|   +---ro minimum-time?         uint32
|   +---ro address-family?       enumeration
|   +---ro as4-support?          boolean
|   +---ro peer-count?           uint32
|   +---ro maximum-connections?  uint32
|   +---ro peer-group-members [neighbor-address]
|     +---ro neighbor-address     dell:ipv6-address-with-prefix
|     +---ro outbound-optimized?  boolean
+---x filter-list
|   +---ro input
|   |   +---ro filter-list-name?   bgp-list-name
|   +---ro output
|     +---ro filter-list [filter-list-name]
|       +---ro filter-list-name    bgp-list-name
|       +---ro local-rib
|         |   +---ro routes-added?      uint32
|         |   +---ro routes-replaced?   uint32
|         |   +---ro routes-withdrawn?  uint32
|         |   +---ro router-id?         dell:ipv6-address-with-
prefix
|         +---ro prefix-list [network-prefix next-hop-addr]
|           +---ro network-prefix     dell:ipv6-address-with-prefix
|           +---ro next-hop-addr     dell:ipv6-address-with-prefix
|           +---ro status-code?      bgp-status-code
|           +---ro best-route?       boolean
|           +---ro path-source?      bgp-path-source
|           +---ro metric?           string
|           +---ro local-pref?       string
|           +---ro weight?           uint32
|           +---ro as-path-string?   string
|           +---ro origin-code?      bgp-origin-code
+---x community
|   +---ro input
|   |   +---ro local-as?            boolean
|   |   +---ro no-advertise?       boolean
|   |   +---ro no-export?          boolean
|   |   +---ro community-number?    string
|   +---ro output
|     +---ro community
|       +---ro local-rib
|         |   +---ro routes-added?      uint32
|         |   +---ro routes-replaced?   uint32
|         |   +---ro routes-withdrawn?  uint32
|         |   +---ro router-id?         dell:ipv6-address-with-
prefix

```

```

|         +--ro prefix-list [network-prefix next-hop-addr]
|         +--ro network-prefix    dell:ipv6-address-with-prefix
|         +--ro next-hop-addr     dell:ipv6-address-with-prefix
|         +--ro status-code?     bgp-status-code
|         +--ro best-route?      boolean
|         +--ro path-source?     bgp-path-source
|         +--ro metric?          string
|         +--ro local-pref?      string
|         +--ro weight?          uint32
|         +--ro as-path-string?  string
|         +--ro origin-code?     bgp-origin-code
+----x community-list
| +--ro input
| | +--ro community-list-name?   bgp-list-name
| +--ro output
|   +--ro community-list [community-list-name]
|   +--ro community-list-name    bgp-list-name
|   +--ro local-rib
|   | +--ro routes-added?       uint32
|   | +--ro routes-replaced?   uint32
|   | +--ro routes-withdrawn?  uint32
|   | +--ro router-id?         dell:ipv6-address-with-
prefix
|
|         +--ro prefix-list [network-prefix next-hop-addr]
|         +--ro network-prefix    dell:ipv6-address-with-prefix
|         +--ro next-hop-addr     dell:ipv6-address-with-prefix
|         +--ro status-code?     bgp-status-code
|         +--ro best-route?      boolean
|         +--ro path-source?     bgp-path-source
|         +--ro metric?          string
|         +--ro local-pref?      string
|         +--ro weight?          uint32
|         +--ro as-path-string?  string
|         +--ro origin-code?     bgp-origin-code
+----x ext-community-list
| +--ro input
| | +--ro ext-community-list-name? bgp-list-name
| +--ro output
|   +--ro ext-community-list [ext-community-list-name]
|   +--ro ext-community-list-name  bgp-list-name
|   +--ro local-rib
|   | +--ro routes-added?       uint32
|   | +--ro routes-replaced?   uint32
|   | +--ro routes-withdrawn?  uint32
|   | +--ro router-id?         dell:ipv6-address-with-
prefix
|
|         +--ro prefix-list [network-prefix next-hop-addr]
|         +--ro network-prefix    dell:ipv6-address-with-prefix
|         +--ro next-hop-addr     dell:ipv6-address-with-prefix
|         +--ro status-code?     bgp-status-code
|         +--ro best-route?      boolean
|         +--ro path-source?     bgp-path-source
|         +--ro metric?          string
|         +--ro local-pref?      string
|         +--ro weight?          uint32
|         +--ro as-path-string?  string
|         +--ro origin-code?     bgp-origin-code
+----x cluster-list
| +--ro input
| | +--ro cluster-list-id?       bgp-list-name
| +--ro output
|   +--ro cluster-list [cluster-list-id]
|   +--ro cluster-list-id       bgp-list-name
|   +--ro local-rib

```

```

|         | +--ro routes-added?      uint32
|         | +--ro routes-replaced?     uint32
|         | +--ro routes-withdrawn?  uint32
|         | +--ro router-id?      dell:ipv6-address-with-
prefix
|
|         +--ro prefix-list [network-prefix next-hop-addr]
|         | +--ro network-prefix  dell:ipv6-address-with-prefix
|         | +--ro next-hop-addr  dell:ipv6-address-with-prefix
|         | +--ro status-code?   bgp-status-code
|         | +--ro best-route?    boolean
|         | +--ro path-source?   bgp-path-source
|         | +--ro metric?        string
|         | +--ro local-pref?    string
|         | +--ro weight?        uint32
|         | +--ro as-path-string? string
|         | +--ro origin-code?   bgp-origin-code
+----x flap-statistics
| +--ro input
| | +--ro filter-list?  string
| +--ro output
| | +--ro flap-statistics
| | | +--ro local-rib
| | | | +--ro routes-added?      uint32
| | | | +--ro routes-replaced?   uint32
| | | | +--ro routes-withdrawn?  uint32
| | | | +--ro router-id?        dell:ipv6-address-with-
prefix
|
|         +--ro routes [network from]
|         | +--ro network          dell:ipv6-address-with-prefix
|         | +--ro from             dell:ipv6-address
|         | +--ro status-code?     bgpv6-status-code
|         | +--ro best-route?      boolean
|         | +--ro flaps?           uint8
|         | +--ro duration?        hour-minute-second
|         | +--ro reuse?           hour-minute-second
|         | +--ro as-path?         string
|         | +--ro path-source?     bgp-path-source
|         | +--ro origin-code?     bgp-origin-code
+----x dampened-paths
| +--ro input
| +--ro output
| | +--ro dampened-paths
| | | +--ro local-rib
| | | | +--ro routes-added?      uint32
| | | | +--ro routes-replaced?   uint32
| | | | +--ro routes-withdrawn?  uint32
| | | | +--ro router-id?        dell:ipv6-address-with-
prefix
|
|         +--ro routes [network from]
|         | +--ro network          dell:ipv6-address-with-prefix
|         | +--ro from             dell:ipv6-address
|         | +--ro status-code?     bgpv6-status-code
|         | +--ro best-route?      boolean
|         | +--ro flaps?           uint8
|         | +--ro duration?        hour-minute-second
|         | +--ro reuse?           hour-minute-second
|         | +--ro as-path?         string
|         | +--ro path-source?     bgp-path-source
|         | +--ro origin-code?     bgp-origin-code

```

Interface

Use the following definition to get the static IPv4 operational data:

Module: STATS
URI: /api/operational/dell/stats
Methods GET
Supported:

Interface Model for IPv4

```
+--ro stats
  +--ro interfaces* [name]
    +--ro name                dell:interface-name
    +--ro description?        string
    +--ro type?               ianaiftype-mib:IANAifType
    +--ro if-index?           uint32
    +--ro admin-status?       enumeration
    +--ro oper-status?        enumeration
    +--ro oper-down-reason?   string
    +--ro mtu?                int32
    +--ro speed?              yang:gauge32
    +--ro queueing-strategy?  string
    +--ro phys-address?       yang:phys-address
    +--ro tunnel-mode?        encapsulation-modes
    +--ro decapsulate-any?    boolean
    +--ro source?             string
    +--ro source-ip-address?  dell:ip-address-any
    +--ro destination-ip-address? dell:ip-address
    +--ro dscp-mapped?        boolean
    +--ro dscp?               uint32
    +--ro flow-label?         uint32
    +--ro hop-limit?          uint32
    +--ro keep-alive?         boolean
    +--ro ka-destination-address? dell:ip-address
    +--ro ka-interval?        uint32
    +--ro ka-attempts-type?   keep-alive-attempt-types
    +--ro ka-attempts?        uint32
    +--ro ka-state?           keep-alive-states
    +--ro ka-state-last-modified? string
    +--ro allow-remote-enabled? boolean
    +--ro allow-remote* [remote-address remote-prefix-len]
      | +--ro remote-address    dell:ip-address
      | +--ro remote-prefix-len  dell:ip-address-prefix-length
    +--ro lag-name?           lag-intf-name
    +--ro current-intf-address? string
    +--ro ipv4-address?       string
    +--ro ipv4-address-mode?  enumeration
    +--ro dhcp-client?        string
    +--ro link-local-ipv6-address? dell:ipv6-address-with-prefix
    +--ro global-ipv6-address* [address]
      | +--ro address           dell:ipv6-address-with-prefix
    +--ro secondary-ip-address* [address]
      | +--ro address           dell:ipv4-address-with-prefix
    +--ro line-speed?         enumeration
    +--ro ip-mtu?             uint32
    +--ro arp-type?           enumeration
    +--ro arp-timeout?        uint32
    +--ro last-clear-intf-cters? uint32
```

```

+---ro last-intf-status-changed? uint32
+---ro in-octets? yang:counter64
+---ro in-unicast-pkts? yang:counter64
+---ro in-multicast-pkts? yang:counter64
+---ro in-broadcast-pkts? yang:counter64
+---ro in-64bytes-pkts? yang:counter64
+---ro in-over-64bytes-pkts? yang:counter64
+---ro in-over-127bytes-pkts? yang:counter64
+---ro in-over-255bytes-pkts? yang:counter64
+---ro in-over-511bytes-pkts? yang:counter64
+---ro in-over-1023bytes-pkts? yang:counter64
+---ro in-symbol-errors? yang:counter64
+---ro in-runts? yang:counter64
+---ro in-giants? yang:counter64
+---ro in-throttles? yang:counter64
+---ro in-crc? yang:counter64
+---ro in-over-run? yang:counter64
+---ro in-discards? yang:counter64
+---ro in-unknown-protocol? yang:counter64
+---ro out-octets? yang:counter64
+---ro out-unicast-pkts? yang:counter64
+---ro out-multicast-pkts? yang:counter64
+---ro out-broadcast-pkts? yang:counter64
+---ro out-64bytes-pkts? yang:counter64
+---ro out-over-64bytes-pkts? yang:counter64
+---ro out-over-127bytes-pkts? yang:counter64
+---ro out-over-255bytes-pkts? yang:counter64
+---ro out-over-511bytes-pkts? yang:counter64
+---ro out-over-1023bytes-pkts? yang:counter64
+---ro out-under-runs? yang:counter64
+---ro out-throttles? yang:counter64
+---ro out-collisions? yang:counter64
+---ro out-wred-drops? yang:counter64
+---ro out-discards? yang:counter64
+---ro out-symbol-errors? yang:counter64
+---ro rate-interval? yang:counter32
+---ro in-pkts-in-mbps? yang:counter32
+---ro in-pkts-in-pkts-per-sec? yang:counter32
+---ro in-cent-line-rate? yang:counter32
+---ro out-pkts-in-mbps? yang:counter32
+---ro out-pkts-in-pkts-per-sec? yang:counter32
+---ro out-cent-line-rate? yang:counter32
+---ro discontinuity-time? yang:timestamp
+---ro owner? enumeration
+---ro member-interfaces
| +---ro interface* [name]
|   +---ro name dell:phy-intf-name
|   +---ro oper-status? enumeration
+---ro min-links? uint32
+---ro vlan-name? vlan-name

```

IP Route

Use the following definition to get the Route statistics:

```

Module:      IP ROUTE
URI:        /api/operational/dell/stats/ip/route
Methods     GET
Supported:

```

Route Model

```
+--ro route
  +--ro gateway-of-last-resort?  string
  +--ro route-entry* [destination]
  | +--ro destination            inet:ip-prefix
  | +--ro next-hop* [address]
  | | +--ro address             inet:ip-address
  | | +--ro interface?         dell:interface-name
  | | +--ro weight?            uint32
  | | +--ro route-type?       enumeration
  | +--ro route-owner?         enumeration
  | +--ro metric?              uint32
  | +--ro age?                  yang:timestamp
  | +--ro route-tag?           string
  | +--ro distance?            uint32
  | +--ro default-route?       boolean
  | +--ro summary?             boolean
  +--ro summary
    +--ro connected-active?     uint32
    +--ro connected-in-active?  uint32
    +--ro dynamic-active?       uint32
    +--ro dynamic-in-active?    uint32
    +--ro static-active?        uint32
    +--ro static-in-active?     uint32
    +--ro route-size-active?    uint32
    +--ro route-size-in-active? uint32
```

IP Interface

The following definition displays the statistics of an IP interface:

Module: IP INTERFACE
URI: /api/operational/dell/stats/ip/interface
Methods GET
Supported:

IP Interface Operational Model

```
+--ro ip
  +--ro interface* [name]
  +--ro name                string
  +--ro description?        string
  +--ro type?                ianaiftype-mib:IANAifType
  +--ro if-admin-status?    enumeration
  +--ro if-oper-status?     enumeration
  +--ro ip
  | +--ro address!
  | | +--ro primary
  | | | +--ro address?     dell:ipv4-address-with-prefix
  | | +--ro secondary* [address]
  | | | +--ro address      dell:ipv4-address-with-prefix
  +--ro if-virtual-address?  dell:ipv4-address-with-prefix
  +--ro if-broadcast-address? inet:ip-address
  +--ro if-address-input?    enumeration
  +--ro if-ip-mtu?           int32
  +--ro protocols
  | +--ro udp-helper* [helper]
  | | +--ro helper         inet:ip-address
```

```

+--ro udp-broadcast?      inet:ip-address
+--ro directed-broadcast? boolean
+--ro proxy-arp?         boolean
+--ro split-horizon?     boolean
+--ro poison-reverse?    boolean
+--ro icmp-redirects?    boolean
+--ro icmp-unreachables? boolean

```

Management – Route

Use the following definition to get the Management–Route statistics:

```

Module:      MANAGEMENT-ROUTE
URI:         /api/operational/dell/stats/ip/management-route
Methods     GET
Supported:

```

Management-Route Model

```

+--ro management-route
  +--ro route-entry* [destination]
  | +--ro destination      dell:ipv4-address-with-prefix
  | +--ro gateway?        string
  | +--ro state?          enumeration
  | +--ro route-source?   enumeration
  +--ro summary
    +--ro connected-active? uint32
    +--ro connected-inactive? uint32
    +--ro dynamic-active?   uint32
    +--ro dynamic-inactive? uint32
    +--ro static-active?    uint32
    +--ro static-inactive?  uint32
    +--ro mgmt-route-size?  uint32

```

MAC Address Table

Use the following definition to get the entries of mac-address-table:

```

Module:      MAC-ADDRESS-TABLE
URI:         /api/operational/dell/stats/mac-address-table
Methods     GET
Supported:

```

MAC Address Table Model

```

+--ro mac-address-table
  +--ro count
  | +--ro dynamic-mac-count? uint32
  | +--ro static-mac-count?  uint32
  | +--ro sticky-mac-count?  uint32
  | +--ro total-mac-in-use?   uint32
  +--ro aging-time?          uint32
  +--ro mac-vlan* [vlan-id]
  | +--ro vlan-id            vlan-id
  | +--ro count
  | | +--ro dynamic-mac-count? uint32
  | | +--ro static-mac-count?  uint32

```

```

| | +--ro sticky-mac-count?   uint32
| | +--ro total-mac-in-use?   uint32
| +--ro mac-address-list* [mac-address]
|   +--ro mac-address       yang:phys-address
|   +--ro type?             enumeration
|   +--ro interface?        string
|   +--ro state?            enumeration
+--ro multicast
  +--ro count?               uint32
  +--ro mac-vlan* [vlan-id]
    +--ro vlan-id           vlan-id
    +--ro static-mac-count? uint32
    +--ro mac-address-list* [mac-address]
      +--ro mac-address     yang:phys-address
      +--ro state?          enumeration
      +--ro l2-mc-index?    uint32
      +--ro interface-list? string

```

Policy Based Routing (PBR)

Use the following model to get the PBR operational data:

```

Module:          STATS
URI:             /api/operational/dell/stats/ip/redirect-list
Methods         GET
Supported:

```

PBR Operational Model

```

+--rw dell
  +--ro stats
    +--ro ip
      +--ro redirect-list [name]
        +--ro name          string
        +--ro filter [seq-no]
          +--ro seq-no      uint32
          +--ro output-action? pbr-rule-types
          +--ro next-hop-ip? dell:ipv4-no-prefix
          +--ro redirect-interface? dell:tunnel-intf-name
          +--ro track-num?   uint32
          +--ro track-status? enumeration
          +--ro protocol?    dell:ip-protocols-type
          +--ro source-type?  pbr-address-types
          +--ro source-address? dell:ipv4-no-prefix
          +--ro source-mask?   dell:ipv4-mask-or-prefix
          +--ro source-operator? ip-acl-operators
          +--ro source-port?   uint32
          +--ro source-start-port? uint32
          +--ro source-end-port?  uint32
          +--ro destination-type? pbr-address-types
          +--ro destination-address? dell:ipv4-no-prefix
          +--ro destination-mask?   dell:ipv4-mask-or-prefix
          +--ro destination-operator? ip-acl-operators
          +--ro destination-port?   uint32
          +--ro destination-start-port? uint32
          +--ro destination-end-port?  uint32
          +--ro destination-bits*    tcp-bits
          +--ro next-hop-status?     enumeration
          +--ro next-hop-interface?  dell:interface-name
          +--ro arp-status?          enumeration

```

```

|      +--ro applied-interface
|      +--ro interfaces*   dell:interface-name
+--ro cam
  +--ro pbr
    +--ro unit [unit-num port-set-num]
      | +--ro unit-num           dell:stack-unit
      | +--ro port-set-num       enumeration
      | +--ro rule-entry [cam-index]
      | | +--ro cam-index         uint32
      | | +--ro port?             uint32
      | | +--ro vlan-id?          dell:vlan-id-num-with-default
      | | +--ro protocol?         dell:ip-protocols-type
      | | +--ro tcp-flag*         tcp-bits
      | | +--ro source-port?      uint32
      | | +--ro destination-port? uint32
      | | +--ro source-ip?        string
      | | +--ro destination-ip?   string
      | | +--ro nexthop-mac?      string
      | | +--ro egress-port?      string
      | +--ro rule-entry-count?   uint32
    +--ro interface [name]
      +--ro name                 dell:interface-name
      +--ro rule-entry [cam-index]
        +--ro cam-index         uint32
        +--ro port?             uint32
        +--ro vlan-id?          dell:vlan-id-num-with-default
        +--ro protocol?         dell:ip-protocols-type
        +--ro tcp-flag*         tcp-bits
        +--ro source-port?      uint32
        +--ro destination-port? uint32
        +--ro source-ip?        string
        +--ro destination-ip?   string
        +--ro nexthop-mac?      string
        +--ro egress-port?      string

```

Port Mirroring

Use the following model to get the Port Mirroring operational data:

```

Module:          STATS
URI:             /api/operational/dell/stats/monitor-session
Methods         GET
Supported:

```

Port Mirroring Operational Model

```

+--rw dell
  +--rw stats
    +--ro monitor-session [session-id source]
      +--ro session-id         uint32
      +--ro source              dell:interface-name
      +--ro session-type?      enumeration
      +--ro destination?       dell:interface-name
      +--ro direction?         enumeration
      +--ro mode?              enumeration
      +--ro remote-vlan?       dell:vlan-id-num
      +--ro rspan-session-type? enumeration
      +--ro source-ip?         dell:ipv4-no-prefix
      +--ro destination-ip?    dell:ipv4-no-prefix

```

Static Access-list

The following definition describes how to retrieve all the Access-lists (MAC and Layer 3):

Module: STATIC ACCESS-LIST
URI: /api/operational/dell/stats/access-list
Methods GET
Supported:

Static Access-list Model

```
+--ro access-list
  +--ro mac* [name]
    | +--ro name          access-list-name
    | +--ro type?        access-list-types
    | +--ro mac-acl-filter* [seq-no]
    |   +--ro seq-no          uint32
    |   +--ro filter-type?   acl-rule-types
    |   +--ro src-mac-filter? acl-incoming-address-filter-types
    |   +--ro src-mac-address? yang:phys-address
    |   +--ro src-mac-mask?   yang:phys-address
    |   +--ro dest-mac-filter? acl-incoming-address-filter-types
    |   +--ro dest-mac-address? yang:phys-address
    |   +--ro dest-mac-mask?  yang:phys-address
    |   +--ro encapsulation-type? mac-acl-encap-types
    |   +--ro encapsulation-operator? mac-acl-encap-operators
    |   +--ro encapsulation-protocol? string
    |   +--ro count-type?    acl-rule-count-types
    |   +--ro egress-count-value? uint32
    |   +--ro ingress-count-value? uint32
    |   +--ro log?           boolean
    |   +--ro monitor?      boolean
    |   +--ro order?        uint32
  +--ro ip* [name]
    +--ro name          access-list-name
    +--ro type?        access-list-types
    +--ro ip-acl-filter* [seq-no]
      +--ro seq-no          uint32
      +--ro filter-type?   acl-rule-types
      +--ro protocol?      ip-protocols
      +--ro ip-protocol-number? uint32
      +--ro src-ip-filter?  acl-incoming-address-filter-types
      +--ro source-ip-address? inet:ipv4-address
      +--ro source-mask?    inet:ipv4-address
      +--ro src-operator?   ip-acl-operators
      +--ro srcport?       uint32
      +--ro srcportrange?  layer4-port-range
      +--ro dest-ip-filter?  acl-incoming-address-filter-types
      +--ro dest-ip-address? inet:ipv4-address
      +--ro destination-mask? inet:ipv4-address
      +--ro dst-operator?   ip-acl-operators
      +--ro dstport?       uint32
      +--ro dstportrange?  layer4-port-range
      +--ro ack?           boolean
      +--ro fin?           boolean
      +--ro psh?           boolean
      +--ro rst?           boolean
      +--ro syn?           boolean
      +--ro urg?           boolean
```

```

+--ro count-type?          acl-rule-count-types
+--ro egress-count-value?  uint32
+--ro ingress-count-value? uint32
+--ro log?                 boolean
+--ro monitor?            boolean
+--ro order?              uint32
+--ro dscp?               uint32
+--ro ecn?                uint32
+--ro fragments?         boolean

```

System

Use the following definition to display the System entries:

```

Module:          STATS
URI:            /api/operational/dell/stats/system
Methods        GET
Supported:

```

System Model

```

+--ro mac-address?          yang:phys-address
+--ro reload-type?         reload-type
+--ro next-boot?           reload-type
+--ro units* [unit-id]
| +--ro unit-id            uint32
| +--ro unit-role?        string
| +--ro unit-status?     string
| +--ro unit-next-boot-status? string
| +--ro required-type?   string
| +--ro current-type?    string
| +--ro master-priority? int32
| +--ro hardware-revision? string
| +--ro os-version?      string
| +--ro num-ports?       uint32
| +--ro up-time?         string
| +--ro jumbo-capable?   boolean
| +--ro poe-capable?     boolean
| +--ro fips-mode?       boolean
| +--ro boot-flash?      string
| +--ro memory-size?     uint32
| +--ro temperature?     int32
| +--ro voltage?         voltage-status
| +--ro serial-number?   string
| +--ro part-number?     string
| +--ro vendor-id?       string
| +--ro date-code?       string
| +--ro country-code?    string
| +--ro piece-part-id?   string
| +--ro ppid-revision?   string
| +--ro service-tag?     string
| +--ro express-service-code? string
| +--ro auto-reboot?     boolean
| +--ro burned-in-mac?   yang:phys-address
| +--ro no-of-macs?      uint32
+--ro power-supplies* [index]
| +--ro index            uint32
| +--ro unit-id?        uint32
| +--ro bay-id?         uint32
| +--ro bay-status?     string

```

```

| +--ro ps-type?      string
| +--ro fan-status?  string
| +--ro fan-speed?   uint32
+--ro fans* [index]
| +--ro index        uint32
| +--ro unit-id?     uint32
| +--ro bay-id?      uint32
| +--ro bay-status?  string
| +--ro fan-id?      uint32
| +--ro fan-status?  string
| +--ro fan-speed?   uint32
+--ro environment-status* [unit-id]
| +--ro unit-id      uint32
| +--ro unit-status? string
| +--ro temperature? int32
| +--ro voltage?     voltage-status
| +--ro temp-status? uint32
+--ro thermal-sensor-reading* [unit-id]
  +--ro unit-id      uint32
  +--ro sensor-class* [sensor-type]
    +--ro sensor-type  sensor-types
    +--ro sensor* [sensor-id]
      +--ro sensor-id  uint32
      +--ro temperature? int32

```

System Alarm

Use the following definition to display the Alarm entries:

```

Module:          STATS
URI:             /api/operational/dell/stats/alarms
Methods         GET
Supported:

```

System Alarm Model

```

+--ro alarms
  +--ro major-alarms [index]
    | +--ro index        uint32
    | +--ro alarm-description? string
    | +--ro duration?    yang:timestamp
  +--ro minor-alarms [index]
    | +--ro index        uint32
    | +--ro alarm-description? string
    | +--ro duration?    yang:timestamp
  +--ro alarm-thresholds [unit-number]
    +--ro unit-number    uint32
    +--ro minor?         uint32
    +--ro minor-off?     uint32
    +--ro major?         uint32
    +--ro major-off?     uint32
    +--ro shutdown?     uint32

```

System Inventory

The following definition displays the Inventory details:

```

Module:          STATS

```

URI: /api/operational/dell/stats/inventory
Methods GET
Supported:

System Inventory Model

```
+--ro inventory
  +--ro system-type?  string
  +--ro system-mode?  string
  +--ro sw-version?   string
  +--ro parts [index]
    | +--ro index          uint32
    | +--ro stack-unit?   uint32
    | +--ro part-type?    string
    | +--ro serial-number? string
    | +--ro part-number?  string
    | +--ro revision?    string
    | +--ro piece-part-id? string
    | +--ro ppid-revision? string
    | +--ro service-tag?  string
    | +--ro express-service-code? string
  +--ro protocols?    string
```

System Version

The following definition displays version-info:

Module: STATS
URI: /api/operational/dell/stats/version-info
Methods GET
Supported:

System Version Model

```
+--ro version-info
  +--ro os-ver?      string
  +--ro sw-ver?     string
  +--ro build-time?  string
  +--ro build-path?  string
  +--ro up-time?    string
  +--ro image-name?  string
  +--ro chassis-type? string
  +--ro processor-type? string
  +--ro flash-info?  string
  +--ro card-info [index]
    | +--ro index          uint32
    | +--ro card-info-detail? string
  +--ro card-if-info [index]
    +--ro index          uint32
    +--ro card-if-info-detail? string
```

VLAN

The following definition is for displaying VLAN operations:

Module: STATS

URI: /api/operational/dell/stats/vlan
Methods GET
Supported:

VLAN Model

```
+--ro vlan* [vlan-id]
  +--ro vlan-id          vlan-id-num
  +--ro vlan-name?      vlan-name
  +--ro default-vlan?   boolean
  +--ro vlan-codes?     vlan-codes
  +--ro vlan-status?    vlan-status
  +--ro vlan-description? string
  +--ro vlan-port-list* [vlan-port-list-num]
    +--ro vlan-port-list-num      int32
    +--ro vlan-port-list-port-mode? vlan-port-mode
    +--ro vlan-port-list-intf-type? vlan-intf-type
    +--ro vlan-port-list-data?    string
```

Operational Data for IPv6

This section describes the method of fetching the IPv6 operational data.

Interface

Use the following definition to get the statistics of IPv6 Interface operational data.

Module: STATS IPv6 INTERFACE
URI: /api/operational/dell/stats/ipv6/interface
Methods GET
Supported:

Interface Model for IPv6

```
+--ro interface* [interface-name]
  +--ro interface-name          dell:interface-name
  +--ro description?           string
  +--ro if-admin-status?       dell:admin-status
  +--ro if-oper-status?        dell:oper-status
  +--ro port-down-cause?       string
  +--ro stateless-address-autoconfig? boolean
  +--ro ipv6-status?           boolean
  +--ro link-local-ipv6-address? dell:ipv6-address
  +--ro global-unicast-ipv6-address* [address]
    | +--ro address          dell:ipv6-address-with-prefix
    | +--ro configuration-type? enumeration
    | +--ro remaining-life-time? string
    | +--ro address-flag?    string
  +--ro virtual-ipv6-address?   dell:ipv6-address-with-prefix
  +--ro received-prefixes* [prefix-count]
    | +--ro prefix-count      uint32
    | +--ro ipv6-address?     dell:ipv6-address-with-prefix
    | +--ro onlink?           boolean
    | +--ro autoconfig?       boolean
    | +--ro valid-lifetime?   string
    | +--ro preferred-lifetime? string
    | +--ro advertised-by* [address]
```

```

|      +--ro address      dell:ipv6-address
+--ro global-anycast-ipv6-address?      dell:ipv6-address-with-prefix
|   +--ro joined-group-ipv6-address* [address]
|   |   +--ro address      dell:ipv6-address
+--ro icmp-redirects?                  boolean
+--ro nd-mtu?                          uint32
+--ro dad?                              boolean
+--ro dad-attempts?                    uint32
+--ro nd-reachable-time?               uint32
+--ro nd-base-reachable-time?          uint32
+--ro nd-advertised-reachable-time?    uint32
+--ro nd-advertised-retransmit-interval? uint32
+--ro nd-router-advertisement-interval-min? uint32
+--ro nd-router-advertisement-interval-max? uint32
+--ro nd-router-advertisement-live-for? uint32
+--ro nd-router-advertised-hop-limit?  uint32
+--ro ipv6-hop-limitfor-orig-pkts?     uint32
+--ro flowlabel-zero
|   +--ro label_value?    boolean
+--ro mld-host
|   +--ro elapsed-time?    dell:hour-minute-second
+--ro valid-mld-pkts-received? yang:counter32
+--ro valid-mld-pkts-sent?    yang:counter32
+--ro reports-received?      yang:counter32
+--ro reports-sent?          yang:counter32
+--ro leaves-sent?           yang:counter32
+--ro mldv2-queries-received? yang:counter32
|   +--ro mldv1-queries-received? yang:counter32
|   +--ro errors-malformed-pkts?  yang:counter32
+--ro neighbors* [ipv6-address]
|   +--ro ipv6-address      dell:ipv6-address-with-prefix
|   +--ro intf-type?        string
|   +--ro age-expires-in-mins? dell:hour-minute
|   +--ro vlan?             dell:vlan-id-num
+--ro state?                enumeration
|   +--ro cpu?               enumeration
|   +--ro hardware-address?  string
+--ro error-stats
|   +--ro errs-total?        yang:counter32
|   +--ro errs-not-sent?     yang:counter32
+--ro errs-no-route?        yang:counter32
|   +--ro errs-admin-prohib? yang:counter32
|   +--ro errs-beyond-scope? yang:counter32
|   +--ro errs-addr-unreach? yang:counter32
|   +--ro errs-port-unreach? yang:counter32
|   +--ro errs-pkt-too-big?  yang:counter32
|   +--ro errs-time-ex-trans? yang:counter32
|   +--ro errs-time-ex-frags? yang:counter32
|   +--ro errs-hdr-field?    yang:counter32
|   +--ro errs-unrec-nxt-hdr? yang:counter32
|   +--ro errs-unrec-opt?    yang:counter32
|   +--ro errs-redirect?     yang:counter32
|   +--ro errs-unknown?     yang:counter32

```

Management-Route

Use the following definition to get the statistics of IPv6 Management-Route:

Module: MANAGEMENT-ROUTE
URI: /api/running/dell/stats/ipv6/management-route

Methods GET
Supported:

IPv6 Management-Route Model

```
+--ro ipv6
  +--ro management-route
    +--ro route-entry* [destination]
      | +--ro destination      dell:ipv6-address-with-prefix
      | +--ro gateway?        string
      | +--ro state?          enumeration
      | +--ro default-route?  boolean
    +--ro summary
      +--ro connected-active?    uint32
      +--ro connected-inactive?  uint32
      +--ro static-active?       uint32
      +--ro static-inactive?     uint32
      +--ro route-size-active?   uint32
      +--ro route-size-inactive? uint32
      +--ro total-active-routes?  uint32
      +--ro total-inactive-routes? uint32
```

Prefix-list

Use the following definition to get the IPv6 Prefix- list data:

Module: PREFIX-LIST
URI: /api/operational/dell/stats/ipv6/prefix-list
Methods GET
Supported:

Prefix-List Model for IPv6

```
+--ro prefix-list
  +--ro last-deletion-insertion? dell:prefix-list-name
  +--ro prefix-list-name* [name]
    +--ro name      dell:prefix-list-name
    +--ro count?    uint32
    +--ro range-entries? uint32
    +--ro sequence-min? uint32
    +--ro sequence-max? uint32
    +--ro seq* [seq-num]
      +--ro seq-num      uint32
      +--ro action?     enumeration
      +--ro ipv6-prefix? dell:ipv6-address-with-prefix
      +--ro ipv6-mask?  dell:ipv6-address-with-prefix
      +--ro ge?         uint32
      +--ro le?         uint32
      +--ro hit-count?  uint32
```

Route

Use the following definition to get the IPv6 Route statistics:

Module: ROUTE
URI: /api/operational/dell/stats/ipv6/route

Methods GET
Supported:

Route Model for IPv6

```
+--ro route
  +--ro gateway-of-last-resort? string
  +--ro route-entry* [destination]
    | +--ro destination      dell:ipv6-address-with-prefix
    | +--ro gateway?        dell:ipv6-address-with-prefix
    | +--ro route-type?     enumeration
    | +--ro route-owner?    string
    | +--ro metric?         string
    | +--ro intf-type?      string
    | +--ro last-change?    yang:timestamp
    | +--ro state?          enumeration
    | +--ro default-route?  boolean
    | +--ro summary?       boolean
  +--ro summary
    +--ro connected-active?  uint32
    +--ro connected-inactive? uint32
    +--ro dynamic-active?    uint32
    +--ro dynamic-in-active? uint32
    +--ro static-active?     uint32
    +--ro static-inactive?   uint32
    +--ro route-size-active? uint32
    +--ro route-size-in-active? uint32
```

Management Information Base (MIB)

Management Information Base (MIB) is a database of objects that can be monitored by a network management system. Dell Networking OS management information system (MIB) supports many new simple network management protocol (SNMP) objects and notifications. The following MIBS are translated and exposed via the REST API.

BGP V2

The following definition displays the properties of Interfaces using BGP V2:

Module: BGP – V2
URI: /api/operational/mib/bgp-v2
Methods GET
Supported:

BGP V2 Model

```
+--ro mib
  +--ro bgp4-v2
    +--ro f10BgpM2
      | +--ro f10BgpM2CapabilitySupportAvailable? snmpv2-tc:TruthValue
      | +--ro f10BgpM2AsSize? enumeration
      | +--ro f10BgpM2LocalAs? inet-
address:InetAutonomousSystemNumber
      | +--ro f10BgpM2LocalIdentifier? inet:ip-address
      | +--ro f10BgpM2RouteReflector? snmpv2-tc:TruthValue
      | +--ro f10BgpM2ClusterId? mib:F10BgpM2Identifier
      | +--ro f10BgpM2ConfederationRouter? snmpv2-tc:TruthValue
```

```

| +--ro f10BgpM2ConfederationId?          inet-
address: InetAutonomousSystemNumber
| +--ro f10BgpM2CfgBaseScalarStorageType? snmpv2-tc:StorageType
| +--ro f10BgpM2CfgLocalAs?              inet-
address: InetAutonomousSystemNumber
| +--ro f10BgpM2CfgLocalIdentifier?      inet:ip-address
| +--ro f10BgpM2CfgRouteReflector?      snmpv2-tc:TruthValue
| +--ro f10BgpM2CfgClusterId?           mib:F10BgpM2Identifier
| +--ro f10BgpM2CfgConfederationRouter? snmpv2-tc:TruthValue
| +--ro f10BgpM2CfgConfederationId?     inet-
address: InetAutonomousSystemNumber
| +--ro f10BgpM2CfgPeerNextIndex?       int32
| +--ro f10BgpM2PathAttrCount?          yang:counter32
+--ro f10BgpM2VersionTable* [f10BgpM2VersionIndex]
| +--ro f10BgpM2VersionIndex            uint32
| +--ro f10BgpM2VersionSupported?      snmpv2-tc:TruthValue
+--ro f10BgpM2SupportedCapabilitiesTable*
[f10BgpM2SupportedCapabilityCode]
| +--ro f10BgpM2SupportedCapabilityCode  uint32
| +--ro f10BgpM2SupportedCapability?    snmpv2-tc:TruthValue
+--ro f10BgpM2PeerTable* [f10BgpM2PeerInstance f10BgpM2PeerLocalAddr
f10BgpM2PeerRemoteAddr]
| +--ro f10BgpM2PeerInstance            uint32
| +--ro f10BgpM2PeerIdentifier?         inet:ip-address
| +--ro f10BgpM2PeerState?             enumeration
| +--ro f10BgpM2PeerStatus?            enumeration
| +--ro f10BgpM2PeerConfiguredVersion? uint32
| +--ro f10BgpM2PeerNegotiatedVersion? uint32
| +--ro f10BgpM2PeerLocalAddr          inet:ip-address
| +--ro f10BgpM2PeerLocalPort?         inet-
address: InetPortNumber
| +--ro f10BgpM2PeerLocalAs?           inet-
address: InetAutonomousSystemNumber
| +--ro f10BgpM2PeerRemoteAddr          inet:ip-address
| +--ro f10BgpM2PeerRemotePort?        inet-
address: InetPortNumber
| +--ro f10BgpM2PeerRemoteAs?          inet-
address: InetAutonomousSystemNumber
| +--ro f10BgpM2PeerIndex?             uint32
| +--ro f10BgpM2PeerLastErrorReceived?  binary
| +--ro f10BgpM2PeerLastErrorSent?     binary
| +--ro f10BgpM2PeerLastErrorReceivedTime? yang:timeticks
| +--ro f10BgpM2PeerLastErrorSentTime? yang:timeticks
| +--ro f10BgpM2PeerLastErrorReceivedText? snmp-
framework: SnmpAdminString
| +--ro f10BgpM2PeerLastErrorSentText?  snmp-
framework: SnmpAdminString
| +--ro f10BgpM2PeerLastErrorReceivedData? binary
| +--ro f10BgpM2PeerLastErrorSentData?  binary
| +--ro f10BgpM2PeerFsmEstablishedTime? yang:gauge32
| +--ro f10BgpM2PeerInUpdatesElapsedTime? yang:gauge32
| +--ro f10BgpM2PeerConnectRetryInterval? uint32
| +--ro f10BgpM2PeerHoldTimeConfigured? uint32
| +--ro f10BgpM2PeerKeepAliveConfigured? uint32
| +--ro f10BgpM2PeerMinASOrigInterval?  uint32
| +--ro f10BgpM2PeerMinRouteAdverInterval? uint32
| +--ro f10BgpM2PeerHoldTime?           uint32
| +--ro f10BgpM2PeerKeepAlive?         uint32
| +--ro f10BgpM2PeerInUpdates?         yang:counter32
| +--ro f10BgpM2PeerOutUpdates?        yang:counter32
| +--ro f10BgpM2PeerInTotalMessages?    yang:counter32
| +--ro f10BgpM2PeerOutTotalMessages?   yang:counter32
| +--ro f10BgpM2PeerFsmEstablishedTrans? yang:counter32
| +--ro f10BgpM2PeerInKeepalives?      yang:counter32

```

```

|   +--ro f10BgpM2PeerOutKeepalives?          yang:counter32
|   +--ro f10BgpM2PeerInOpen?                yang:counter32
|   +--ro f10BgpM2PeerOutOpen?               yang:counter32
|   +--ro f10BgpM2PeerInRteRefresh?          yang:counter32
|   +--ro f10BgpM2PeerOutRteRefresh?         yang:counter32
|   +--ro f10BgpM2PeerReflectorClient?       enumeration
|   +--ro f10BgpM2PeerConfedMember?          snmpv2-tc:TruthValue
|   +--ro f10BgpM2CfgPeerConfedMember?      snmpv2-tc:TruthValue
|   +--ro f10BgpM2PeerGroupName?             string
+--ro f10BgpM2PeerCapsAnnouncedTable* [f10BgpM2PeerIndex
f10BgpM2PeerCapAnnouncedCode f10BgpM2PeerCapAnnouncedIndex]
|   +--ro f10BgpM2PeerIndex                   uint32
|   +--ro f10BgpM2PeerCapAnnouncedCode      uint32
|   +--ro f10BgpM2PeerCapAnnouncedIndex     uint32
|   +--ro f10BgpM2PeerCapAnnouncedValue?    binary
+--ro f10BgpM2PeerCapsReceivedTable* [f10BgpM2PeerIndex
f10BgpM2PeerCapReceivedCode f10BgpM2PeerCapReceivedIndex]
|   +--ro f10BgpM2PeerIndex                   uint32
|   +--ro f10BgpM2PeerCapReceivedCode        uint32
|   +--ro f10BgpM2PeerCapReceivedIndex       uint32
|   +--ro f10BgpM2PeerCapReceivedValue?      binary
+--ro f10BgpM2PrefixCountersTable* [f10BgpM2PeerIndex
f10BgpM2PrefixCountersAfi f10BgpM2PrefixCountersSafi]
|   +--ro f10BgpM2PeerIndex                   uint32
|   +--ro f10BgpM2PrefixCountersAfi          mib:F10BgpM2Afi
|   +--ro f10BgpM2PrefixCountersSafi         mib:F10BgpM2Safi
|   +--ro f10BgpM2PrefixInPrefixes?         yang:gauge32
|   +--ro f10BgpM2PrefixInPrefixesAccepted? yang:gauge32
|   +--ro f10BgpM2PrefixInPrefixesRejected? yang:gauge32
|   +--ro f10BgpM2PrefixOutPrefixes?        yang:gauge32
|   +--ro f10BgpM2PrefixWdrawnByPeer?       yang:gauge32
|   +--ro f10BgpM2PrefixWdrawnFromPeer?     yang:gauge32
+--ro f10BgpM2CfgPeerAdminStatusTable* [f10BgpM2PeerIndex]
|   +--ro f10BgpM2PeerIndex                   uint32
|   +--ro f10BgpM2CfgPeerAdminStatus?        enumeration
+--ro f10BgpM2CfgPeerTable* [f10BgpM2CfgPeerIndex]
|   +--ro f10BgpM2CfgPeerIndex                 uint32
|   +--ro f10BgpM2CfgPeerConfiguredVersion?  uint32
|   +--ro f10BgpM2CfgAllowVersionNegotiation? snmpv2-tc:TruthValue
|   +--ro f10BgpM2CfgPeerLocalAddr?           inet:ip-address
|   +--ro f10BgpM2CfgPeerLocalAs?            inet-
address:InetAutonomousSystemNumber
|   +--ro f10BgpM2CfgPeerRemoteAddr?          inet:ip-address
|   +--ro f10BgpM2CfgPeerRemoteAs?           inet-
address:InetAutonomousSystemNumber
|   +--ro f10BgpM2CfgPeerEntryStorageType?    snmpv2-tc:StorageType
|   +--ro f10BgpM2CfgPeerError?              enumeration
|   +--ro f10BgpM2CfgPeerBgpPeerEntry?       snmpv2-tc:RowPointer
|   +--ro f10BgpM2CfgPeerRowEntryStatus?     snmpv2-tc:RowStatus
|   +--ro f10BgpM2CfgPeerStatus?             enumeration
|   +--ro f10BgpM2CfgPeerConnectRetryInterval? uint32
|   +--ro f10BgpM2CfgPeerHoldTimeConfigured? uint32
|   +--ro f10BgpM2CfgPeerKeepAliveConfigured? uint32
|   +--ro f10BgpM2CfgPeerMinASOrigInterval?  uint32
|   +--ro f10BgpM2CfgPeerMinRouteAdverInter? uint32
|   +--ro f10BgpM2CfgPeerReflectorClient?    enumeration
+--ro f10BgpM2NlriTable* [f10BgpM2PeerIndex f10BgpM2NlriAfi
f10BgpM2NlriSafi f10BgpM2NlriPrefix f10BgpM2NlriPrefixLen f10BgpM2NlriIndex]
|   +--ro f10BgpM2PeerIndex                   uint32
|   +--ro f10BgpM2NlriIndex                   uint32
|   +--ro f10BgpM2NlriAfi                     mib:F10BgpM2Afi
|   +--ro f10BgpM2NlriSafi                     mib:F10BgpM2Safi
|   +--ro f10BgpM2NlriPrefix                   inet:ip-address
|   +--ro f10BgpM2NlriPrefixLen               inet-

```

```

address:InetAddressPrefixLength
| +---ro f10BgpM2NlriBest?          snmpv2-tc:TruthValue
| +---ro f10BgpM2NlriCalcLocalPref? uint32
| +---ro f10BgpM2PathAttrIndex?    uint32
| +---ro f10BgpM2NlriOpaqueType?   enumeration
| +---ro f10BgpM2NlriOpaquePointer? snmpv2-tc:RowPointer
| +---ro f10BgpM2RouteFlag?        enumeration
+---ro f10BgpM2AdjRibsOutTable* [f10BgpM2PeerIndex f10BgpM2NlriAfi
f10BgpM2NlriSafi f10BgpM2NlriPrefix f10BgpM2NlriPrefixLen
f10BgpM2AdjRibsOutIndex]
| +---ro f10BgpM2PeerIndex          uint32
| +---ro f10BgpM2NlriAfi           mib:F10BgpM2Afi
| +---ro f10BgpM2NlriSafi          mib:F10BgpM2Safi
| +---ro f10BgpM2NlriPrefix        inet:ip-address
| +---ro f10BgpM2NlriPrefixLen     inet-address:InetAddressPrefixLength
| +---ro f10BgpM2AdjRibsOutIndex   uint32
| +---ro f10BgpM2AdjRibsOutRoute?  snmpv2-tc:RowPointer
+---ro f10BgpM2PathAttrTable* [f10BgpM2PathAttrIndex]
| +---ro f10BgpM2PathAttrIndex     uint32
| +---ro f10BgpM2PathAttrOrigin?   enumeration
| +---ro f10BgpM2PathAttrNextHop?  inet:ip-address
| +---ro f10BgpM2PathAttrMedPresent? snmpv2-tc:TruthValue
| +---ro f10BgpM2PathAttrMed?      uint32
| +---ro f10BgpM2PathAttrLocalPrefPresent? snmpv2-tc:TruthValue
| +---ro f10BgpM2PathAttrLocalPref? uint32
| +---ro f10BgpM2PathAttrAtomicAggregate? enumeration
| +---ro f10BgpM2PathAttrAggregatorAS? inet-
address:InetAddressAutonomousSystemNumber
| +---ro f10BgpM2PathAttrAggregatorAddr? mib:F10BgpM2Identifier
| +---ro f10BgpM2AsPathCalcLength?    uint32
| +---ro f10BgpM2AsPathString?        snmp-
framework:SnmpAdminString
| +---ro f10BgpM2AsPathIndex?          uint32
| +---ro f10BgpM2AsPath4bytePathPresent? snmpv2-tc:TruthValue
| +---ro f10BgpM2AsPath4byteAggregatorAS? inet-
address:InetAddressAutonomousSystemNumber
| +---ro f10BgpM2AsPath4byteCalcLength? uint32
| +---ro f10BgpM2AsPath4byteString?    snmp-
framework:SnmpAdminString
| +---ro f10BgpM2AsPath4byteIndex?     uint32
+---ro f10BgpM2AsPathTable* [f10BgpM2PathAttrIndex
f10BgpM2AsPathSegmentIndex f10BgpM2AsPathElementIndex]
| +---ro f10BgpM2PathAttrIndex        uint32
| +---ro f10BgpM2AsPathSegmentIndex   uint32
| +---ro f10BgpM2AsPathElementIndex   uint32
| +---ro f10BgpM2AsPathType?          enumeration
| +---ro f10BgpM2AsPathElementValue?  inet-
address:InetAddressAutonomousSystemNumber
+---ro f10BgpM2PathAttrUnknownTable* [f10BgpM2PathAttrIndex
f10BgpM2PathAttrUnknownIndex]
| +---ro f10BgpM2PathAttrIndex        uint32
| +---ro f10BgpM2PathAttrUnknownIndex uint32
| +---ro f10BgpM2PathAttrUnknownType? uint32
| +---ro f10BgpM2PathAttrUnknownValue? binary
+---ro f10BgpM2PathAttrCommTable* [f10BgpM2PathAttrIndex
f10BgpM2PathAttrCommIndex]
| +---ro f10BgpM2PathAttrIndex        uint32
| +---ro f10BgpM2PathAttrCommIndex    uint32
| +---ro f10BgpM2PathAttrCommValue?   mib:F10BgpM2Community
+---ro f10BgpM2LinkLocalNextHopTable* [f10BgpM2PathAttrIndex]
| +---ro f10BgpM2PathAttrIndex        uint32
| +---ro f10BgpM2LinkLocalNextHopPresent? snmpv2-tc:TruthValue
| +---ro f10BgpM2LinkLocalNextHop?   inet-address:InetAddress
+---ro f10BgpM2PathAttrOriginatorIdTable* [f10BgpM2PathAttrIndex]

```

```

    | +--ro f10BgpM2PathAttrIndex          uint32
    | +--ro f10BgpM2PathAttrOriginatorId? inet:ip-address
    +--ro f10BgpM2PathAttrClusterTable* [f10BgpM2PathAttrIndex
f10BgpM2PathAttrClusterIndex]
    | +--ro f10BgpM2PathAttrIndex          uint32
    | +--ro f10BgpM2PathAttrClusterIndex  uint32
    | +--ro f10BgpM2PathAttrClusterValue? mib:F10BgpM2Identifier
    +--ro f10BgpM2PathAttrExtCommTable* [f10BgpM2PathAttrIndex
f10BgpM2PathAttrExtCommIndex]
    | +--ro f10BgpM2PathAttrIndex          uint32
    | +--ro f10BgpM2PathAttrExtCommIndex  uint32
    | +--ro f10BgpM2PathAttrExtCommValue? mib:F10BgpM2ExtendedCommunity
    +--ro f10BgpM2FlapStatisticsTable* [NetworkPrefix NetworkPrefixLen
NextHopAddress]
    +--ro NetworkPrefix                    inet:ip-address
    +--ro NetworkPrefixLen                  inet-address:InetAddressPrefixLength
    +--ro NextHopAddress                    inet:ip-address
    +--ro StatusCode?                       enumeration
    +--ro BestRoute?                        snmpv2-tc:TruthValue
    +--ro PathSource?                       enumeration
    +--ro OriginCode?                       enumeration
    +--ro RouteFlapCount?                   uint32
    +--ro RouteFlapDuration?                yang:timeticks
    +--ro ReuseDuration?                    yang:timeticks
    +--ro AsPathString?                     snmp-framework:SnmpAdminString

```

Forwarding Information Base (FIB)

The following definition displays the properties of interfaces using MIBs FIB:

```

Module:          FIB
URI:             /api/operational/mib/fib
Methods         GET
Supported:

```

MIBs FIB Model

```

+--ro mib
  +--ro fib
    +--ro f10IpForwardMibObjects
      | +--ro f10InetCidrIpv4RouteNumber?  int32
      | +--ro f10InetCidrIpv6RouteNumber?  int32
      +--ro f10IpForwardVersionTable* [chSysCardNumber
f10IpForwardAddrFamily]
      | +--ro chSysCardNumber                uint32
      | +--ro f10IpForwardAddrFamily         enumeration
      | +--ro f10IpForwardVersion?           uint64
      +--ro f10InetCidrRouteTable* [chSysCardNumber f10InetCidrRouteDest
f10InetCidrRouteNextHop f10InetCidrRouteFirstHop]
      +--ro chSysCardNumber                  uint32
      +--ro f10InetCidrRouteDest              string
      +--ro f10InetCidrRouteNextHop           string
      +--ro f10InetCidrRouteFirstHop          string
      +--ro f10InetCidrRouteIfIndex?         uint32
      +--ro f10InetCidrRouteMacAddress?      string
      +--ro f10InetCidrRouteEgressPort?     string
      +--ro f10InetCidrRouteCamIndex?        uint32

```

Forwarding Plane Statistics

The following definition displays the Forwarding Plane statistics:

Module: FP-STATS
URI: /api/operational/mib/fp-stats
Methods Supported: GET

Forwarding Plane Statistics Model

```
+--ro mib
  +--ro fp-stats
    +--ro fp-stats-table* [stackUnitId]
      | +--ro stackUnitId      int32
      | +--ro rxHandle?       int32
      | +--ro numMsgHdr?      int32
      | +--ro numMsgBuf?      int32
      | +--ro numCluster?     int32
      | +--ro received?       int32
      | +--ro dropped?        int32
      | +--ro rxToNetwork?    int32
      | +--ro rxError?        int32
      | +--ro rxDatapathError? int32
      | +--ro rxPktCOS0?      int32
      | +--ro rxPktCOS1?      int32
      | +--ro rxPktCOS2?      int32
      | +--ro rxPktCOS3?      int32
      | +--ro rxPktCOS4?      int32
      | +--ro rxPktCOS5?      int32
      | +--ro rxPktCOS6?      int32
      | +--ro rxPktCOS7?      int32
      | +--ro rxPktUnit0?     int32
      | +--ro rxPktUnit1?     int32
      | +--ro rxPktUnit2?     int32
      | +--ro rxPktUnit3?     int32
      | +--ro transmitted?    int32
      | +--ro txRequested?    int32
      | +--ro txDescriptor?   int32
      | +--ro txError?        int32
      | +--ro txReqTooLarge?  int32
      | +--ro txInternalError? int32
      | +--ro txDatapathError? int32
      | +--ro txPktCOS0?      int32
      | +--ro txPktCOS1?      int32
      | +--ro txPktCOS2?      int32
      | +--ro txPktCOS3?      int32
      | +--ro txPktCOS4?      int32
      | +--ro txPktCOS5?      int32
      | +--ro txPktCOS6?      int32
      | +--ro txPktCOS7?      int32
      | +--ro txPktUnit0?     int32
      | +--ro txPktUnit1?     int32
      | +--ro txPktUnit2?     int32
      | +--ro txPktUnit3?     int32
    +--ro fp-cpu-party-bus-stats-table* [stackUnitId]
      | +--ro stackUnitId      int32
      | +--ro inputPackets?    int32
      | +--ro inputBytes?      int32
```

```

|   +--ro inputDropped?      int32
|   +--ro inputError?       int32
|   +--ro outputPackets?    int32
|   +--ro outputBytes?     int32
|   +--ro outputError?     int32
+--ro fp-drops-table* [stackUnitId stackPortId]
|   +--ro stackUnitId       int32
|   +--ro stackPortId      int32
|   +--ro inDrops?         yang:counter64
|   +--ro inIBPCBPFFullDrops? yang:counter64
|   +--ro inPortSTPnotFwdDrops? yang:counter64
|   +--ro inIPv4L3Discards? yang:counter64
|   +--ro inPolicyDiscards? yang:counter64
|   +--ro inPktDroppedByFP? yang:counter64
|   +--ro inL2L3Drops?     yang:counter64
|   +--ro inPortBitMapZeroDrops? yang:counter64
|   +--ro inRxFVLANDrops?  yang:counter64
|   +--ro inFCSDDrops?     yang:counter64
|   +--ro inMTUExceeds?    yang:counter64
|   +--ro mmuHoldDrops?    yang:counter64
|   +--ro mmuTxPurgeCellErr? yang:counter64
|   +--ro mmuAgedDrops?    yang:counter64
|   +--ro egressFCSDDrops? yang:counter64
|   +--ro egIPv4L3UCAgedDrops? yang:counter64
|   +--ro egTTLThresholdDrops? yang:counter64
|   +--ro egInvalidVLANCounterDrops? yang:counter64
|   +--ro egL2MCDrops?     yang:counter64
|   +--ro egPktDropsOfAnyCondition? yang:counter64
|   +--ro egHgMacUnderFlow? yang:counter64
|   +--ro egTxErrPktCounter? yang:counter64
+--ro fp-packet-buffer-table* [stackUnitId portPipe]
|   +--ro stackUnitId      int32
|   +--ro portPipe         int32
|   +--ro totalPktBuffer?  int32
|   +--ro currentAvailableBuffer? int32
|   +--ro packetBufferAlloc? int32
+--ro fp-port-stats-table* [stackUnitId stackPortId]
|   +--ro stackUnitId      int32
|   +--ro stackPortId      int32
|   +--ro currentUsagePerPort? int32
|   +--ro defaultPktBufferAlloc? int32
|   +--ro maxLimitPerPort? int32
+--ro fp-cos-stats-table* [stackUnitId stackPortId stackPortCOSId]
|   +--ro stackUnitId      int32
|   +--ro stackPortId      int32
|   +--ro stackPortCOSId   int32
|   +--ro currentUsagePerCOS? int32
|   +--ro defaultPktBufferAlloc? int32
|   +--ro maxLimitPerCOS? int32

```

IF MIB Operations

The following definition displays the properties of Interfaces using the IF-MIB:

```

Module:      IF-MIB
URI:         /api/operational/mib/if-mib
Methods     GET
Supported:

```

IF MIB Model

```
+--ro mib
  +--ro if-mib
    +--ro ifNumber? int32
    +--ro ifTable* [ifIndex]
      +--ro ifIndex mib:InterfaceIndex
      +--ro ifDescr? smiv2:DisplayString
      +--ro ifType? ianaiftype-mib:IANAifType
      +--ro ifMtu? int32
      +--ro ifSpeed? yang:gauge32
      +--ro ifPhysAddress? yang:phys-address
      +--ro ifAdminStatus? enumeration
      +--ro ifOperStatus? enumeration
      +--ro ifLastChange? yang:timeticks
      +--ro ifInOctets? yang:counter32
      +--ro ifInUcastPkts? yang:counter32
      x--ro ifInNUcastPkts? yang:counter32
      +--ro ifInDiscards? yang:counter32
      +--ro ifInErrors? yang:counter32
      +--ro ifInUnknownProtos? yang:counter32
      +--ro ifOutOctets? yang:counter32
      +--ro ifOutUcastPkts? yang:counter32
      x--ro ifOutNUcastPkts? yang:counter32
      +--ro ifOutDiscards? yang:counter32
      +--ro ifOutErrors? yang:counter32
      x--ro ifOutQLen? yang:gauge32
      x--ro ifSpecific? yang:object-identifier
      +--ro ifName? smiv2:DisplayString
      +--ro ifInMulticastPkts? yang:counter32
      +--ro ifInBroadcastPkts? yang:counter32
      +--ro ifOutMulticastPkts? yang:counter32
      +--ro ifOutBroadcastPkts? yang:counter32
      +--ro ifHCInOctets? yang:counter64
      +--ro ifHCInUcastPkts? yang:counter64
      +--ro ifHCInMulticastPkts? yang:counter64
      +--ro ifHCInBroadcastPkts? yang:counter64
      +--ro ifHCOutOctets? yang:counter64
      +--ro ifHCOutUcastPkts? yang:counter64
      +--ro ifHCOutMulticastPkts? yang:counter64
      +--ro ifHCOutBroadcastPkts? yang:counter64
      +--ro ifLinkUpDownTrapEnable? enumeration
      +--ro ifHighSpeed? yang:gauge32
      +--ro ifPromiscuousMode? smiv2:TruthValue
      +--ro ifConnectorPresent? smiv2:TruthValue
      +--ro ifAlias? smiv2:DisplayString
      +--ro ifCounterDiscontinuityTime? yang:timestamp
      x--ro ifTestId? smiv2:TestAndIncr
      x--ro ifTestStatus? enumeration
      x--ro ifTestType? smiv2:AutonomousType
      x--ro ifTestResult? enumeration
      x--ro ifTestCode? yang:object-identifier
      x--ro ifTestOwner? mib:OwnerString
      +--ro ifStackTable* [ifStackHigherLayer ifStackLowerLayer]
      | +--ro ifStackHigherLayer mib:InterfaceIndexOrZero
      | +--ro ifStackLowerLayer mib:InterfaceIndexOrZero
      | +--ro ifStackStatus? smiv2:RowStatus
    +--ro ifRcvAddressTable* [ifIndex ifRcvAddressAddress]
      | +--ro ifIndex leafref
      | +--ro ifRcvAddressAddress yang:phys-address
      | +--ro ifRcvAddressStatus? smiv2:RowStatus
      | +--ro ifRcvAddressType? enumeration
```

```
+++ro ifTableLastChange? yang:timeticks
+++ro ifStackLastChange? yang:timeticks
```

IETF Interface Operations

The following definition configures and displays the properties of an Interface using IETF:

Module: IETF INTERFACE
URI: /api/operational/interfaces
Methods Supported: GET, PATCH, and PUT

IETF Interface Model


```
+-rw interfaces
  +-rw interface [name]
    +-rw name string
    +-rw description? string
    +-ro type? ianaift:iana-if-type
    +-ro location? string
    +-rw enabled? boolean
    +-ro if-index? int32
    +-rw mtu? uint32
    +-rw link-up-down-trap-enable? enumeration
    +-rw rate-interval? uint32
```

REST API Framework to Execute the CLIs

REST CLI is an alternative approach for Telnet and SSH to send the Dell Networking OS commands to the system.

The command is sent as an XML payload and the corresponding command output or error message returns in an unstructured format in the XML response. There are three sets of XML commands:

- **config-commands** – to send the `configuration` commands to the system. Commands with submodes are allowed within a single XML tag by using the separator `\r\n`.
- **show-command** – to send the `show` commands to the system. The keyword `show` is not required explicitly in the tag (for example, for `show version`, use `version`).
- **exec-command** – to configure the `exec` commands on the system.

 **NOTE:** Interactive CLIs (for example, `clear counters`, `reload`, and so on) are not supported via the REST API framework. Also, the commands sent in XML payload are restricted to a maximum of 1000 characters length.

To execute REST-CLI on the device, the HTTP username and password needs the appropriate privilege. The REST-CLI request returns `HTTP/1.1 200 OK` on a successful completion. All the other HTTP error codes are treated as failure. Multiple command tags are not allowed inside the `<input>` tag.

The tree structure of the REST CLI is as follows:

```
module: dell
+-rw dell?
  |
  +-ro cli
```

```

+--ro input
|   +--ro config-commands?   string
|   +--ro show-command?     dell:show-cmd-str
|   +--ro exec-command?     dell:cli-cmd-str
+--ro output
   +--ro command             string

```

Samples of the config-command

Following is the sample output of CONFIG command:

To configure an IP address:

Input.xml:

```

<input>
<config-commands>
interface vlan 100\r\n
ip address 1.2.3.4/24
</config-commands>
</input>

```

```

curl -u demo:demo -X POST -T Input.xml http://<IP>:8008/api/running/dell/
_operations/cli
<output xmlns='http://www.dell.com/ns/dell:0.1/root'>
<command>conf
Dell(conf)#interface vlan 100
Dell(conf-if-vl-100)#ip address 1.2.3.4/24
Dell(conf-if-vl-100)#end

```

To configure MTU:

Input.xml

```

<input>
<config-commands>
interface vlan 100\r\n
mtu 10000
</config-commands>
</input>

```

```

curl -u demo:demo -X POST -T Input.xml http://<IP>:8008/api/running/dell/
_operations/cli
<output xmlns='http://www.dell.com/ns/dell:0.1/root'>
<command>conf
Dell(conf)#interface vlan 100
Dell(conf-if-vl-100)#mtu 10000
Dell(conf-if-vl-100)#end

```

Samples of the EXEC command

Following is the sample output of EXEC command:

Input.xml:

```

<input>
  <exec-command>ping 10.16.127.35</exec-command>
</input>

```

```

curl -u demo:demo http://10.16.129.43:8008/api/running/dell/_operations/cli -X
POST -T ./input.xml
<output xmlns='http://www.dell.com/ns/dell:0.1/root'>
  <command>ping 10.16.127.35
Type Ctrl-C to abort.

```

```
Sending 5, 100-byte ICMP Echos to 10.16.127.35, timeout is 2 seconds:
!!!!
Success rate is 100.0 percent (5/5), round-trip min/avg/max = 0/4/20 (ms)
Dell# </command>
</output>
```

Samples of the show-command

Following is the sample output of SHOW command:

Input.xml:

```
<input>
  <show-command>version</show-command>
</input>
```

```
curl -u demo:demo -X POST -T Input.xml http://<IP>:8008/api/running/dell/
_operations/cli
```

```
<output xmlns='http://www.dell.com/ns/dell:0.1/root'>
  <command>show version
Dell Real Time Operating System Software
Dell Operating System Version: 2.0
Dell Application Software Version: 9-7(0-90)
Copyright (c) 1999-2014 by Dell Inc. All Rights Reserved.
<snip...>
Dell Networking OS uptime is 10 minute(s)
System image file is "DT-MAA-S4810-16"
System Type: S4810
Control Processor: Freescale QorIQ P2020 with 2 Gbytes (2147483648 bytes) of
memory, cores(s) 1.
128M bytes of boot flash memory.
1 52-port GE/TE/FG (SE)
48 Ten GigabitEthernet/IEEE 802.3 interface(s)
  4 Forty GigabitEthernet/IEEE 802.3 interface(s)
Dell#</command>
</output>
```

REST API CLI

The REST API CLI commands are supported on the S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, and Z9500 platforms.

rest-server

Enable the REST API service over a non-secure or secure HTTP.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, and Z9500

Syntax `rest-server {http | secure-http}`
 To disable the REST API over a non-secure or secure HTTP request, use the `no rest-server {http | secure-http}` command.

Parameters

| | |
|--------------------------|--|
| <code>http</code> | Enable the REST API on HTTP (Port: 8008). |
| <code>secure-http</code> | Enable the REST API on HTTPS (Port: 8888). |

Defaults REST API is disabled by default.

Command Modes CONFIGURATION

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S6000-ON. |
| 9.6(0.0) | Introduced on the S5000. |
| 9.5(0.1) | Introduced on the Z9500. |
| 9.4(0.0P1) | Introduced on the S4810, S4820T, S6000, and Z9000. |

Usage Information



NOTE: The `rest enable` command is deprecated; however, the support is maintained for backward compatibility in version 9.4(0.0P1) and is removed in the 9.5(0.1) release.

crypto cert generate

Generate a self-signed certificate and key files, if exists, and replace them with the newly generated files.

S4810, S4820T, S3048-ON, S5000, S6000, S6000-ON, and Z9500

Syntax

```
crypto cert generate {cert-file file-name key-file file-name}
[cname common-name] [country country-name] [email email-id]
[length key-length] [locality locality-name] [organization
organization-name] [orgunit organization-unit-name] [state
state-name] [validity days]
```

Parameters

| | |
|---------------------------------------|---|
| cert-file file-name | Enter the keyword <code>cert-file</code> then the filename of the certificate to generate with the full path. |
| key-file file-name | Enter the keyword <code>key-file</code> then the key filename to generate with the full path. |
| cname common-name | (OPTIONAL) Enter the keyword <code>cname</code> then the common name to include in the certificate. The default is Dell Networking. |
| country country-name | (OPTIONAL) Enter the keyword <code>country</code> then the name of the country to include in the certificate. The maximum number of characters allowed is two. The default is US. |
| email email-id | (OPTIONAL) Enter the keyword <code>email</code> then the email-id to include in the certificate. |
| length key-length | (OPTIONAL) Enter the keyword <code>length</code> then the key length in bits. The key length range is from 1024 to 4096 bits. The default is 2048 bits. |
| locality locality-name | (OPTIONAL) Enter the keyword <code>locality</code> then the name of the locality to include in the certificate. The default is San Jose. |
| organization organization-name | (OPTIONAL) Enter the keyword <code>organization</code> then the name of the organization to include in the certificate. The default is Dell. |
| orgunit organization-unit-name | (OPTIONAL) Enter the keyword <code>orgunit</code> then the organization unit name to include in the certificate. The default is Dell Networking. |
| state state-name | (OPTIONAL) Enter the keyword <code>state</code> then the name of the state to include in the certificate. The default is California. |
| validity days | (OPTIONAL) Enter the keyword <code>validity</code> then the number of days the certificate is valid. The validity range is from 1 to 10000 days. The default value is 3650 days (10 years). |


Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S Series and Z Series switches. |

Usage Information The maximum number of characters allowed for the parameters described (except country-name) is 64.

 **NOTE:** Any customized SSL certificate used for Secure REST API communication must be installed individually in each box in the stack setup.

crypto cert install rest-server

Install the certificate and key files to use in the REST service secure communication channel.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, and Z9500

Syntax `crypto cert install rest-server cert-file-name key-file-name`

Parameters

| | |
|------------------------------|--|
| <i>cert-file-name</i> | Enter the certificate filename with the full path. |
| <i>key- file-name</i> | Enter the key filename with the full path. |

Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S Series and Z Series switches. |

Usage Information If either certificate or key files are not present or not valid, the prompt displays a message *File not found/invalid*.

show crypto cert

Display the certificate information.

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, and Z9500

| | | |
|---------------------|--|---|
| Syntax | <code>show crypto cert {rest-server file-name}</code> | |
| Command Mode | EXEC | |
| Parameters | rest-server | Enter the keyword <code>rest-server</code> to display the certificate information installed for rest service. |
| | file-name | Enter the filename with full path to display the certificate information. |

Command Modes EXEC

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|---|
| 9.8(0.0P2) | Introduced on the S3048–ON. |
| 9.7(0.0) | Introduced on the S6000–ON. |
| 9.6(0.0) | Introduced on the S Series and Z Series switches. |

Example

```
Dell#show crypto cert rest-server
Certificate:
  Data:
    Version: 1 (0x0)
    Serial Number:
      b8:9c:00:51:b1:51:9c:3b
    Signature Algorithm: sha1WithRSAEncryption
    Issuer: C=US, ST=California, L=San Jose, O=Dell,
OU=Dell Networking,CN=S6000-NA
  Validity
    Not Before: Apr 28 19:05:16 2014 GMT
    Not After : Apr 30 19:05:16 2014 GMT
Dell#
```

show ip rest-server

Display the status of the configured rest-server (HTTP or Secure HTTP).

S4810, S4820T, S3048–ON, S5000, S6000, S6000–ON, and Z9500

Syntax `show ip rest-server`

Defaults none

Command Modes EXEC Privilege

Command History This guide is platform-specific. For command information about other platforms, see the relevant *Dell Networking OS Command Line Reference Guide*.

The following is a list of the Dell Networking OS version history for this command.

| Version | Description |
|------------|--|
| 9.8(0.0P2) | Introduced on the S3048-ON. |
| 9.7(0.0) | Introduced on the S4810, S4820T, S5000, S6000, S6000-ON, Z9000, and Z9500. |

Example of the rest service enabled over a non-secure HTTP

```
Dell#show ip rest-server
REST server status      : enabled
REST server protocol    : HTTP
REST server port        : 8008
Dell#
```

Example of the rest service enabled over a secure HTTP

```
Dell#show ip rest-server
REST server status      : enabled
REST server protocol    : HTTPS
REST server port        : 8888
Dell#
```

Web Server with HTTP Support

You download web server with HTTP support with the SmartScripts package. It is supported on the S4810, S4820T, S5000, and MXL switch platforms.

For more information, see the [Download the Smart Scripting Package](#).

Starting the Web Server

In the Open Automation package, the web server runs on a switch and handles HTTP and HTTPS requests. You can start the web server in a non-secure (HTTP) or secure (HTTPS) mode.

To start the web server in a non-secure (without SSL) mode for receiving HTTP requests and write the configuration to the running configuration, use the `http-server http` command.

To start the web server in a secure mode for receiving HTTPS requests and write the configuration to the running configuration, use the `http-server secure-http` command.

- Starts the web-server application in non-secure mode to receive HTTP requests.
CONFIGURATION mode

```
http-server http
```

- Starts the web-server application in secure mode using SSL to receive HTTPS requests.
CONFIGURATION mode

```
http-server secure-http
```

To stop the web server and remove the configuration from the running-configuration file, use the `no http-server {http | secure-http}` command.