CONTENTS

PREFACE

Preface xi
   Audience xi
   Document Conventions xi
   Related Documentation for Cisco Nexus 9000 Series Switches xii
   Documentation Feedback xii
   Communications, Services, and Additional Information xii

CHAPTER 1

New and Changed Information 1
   New and Changed Information 1

CHAPTER 2

Platform Support for Fundamentals Features 3
   Platform Support for Fundamentals Features 3

CHAPTER 3

Overview 5
   Software Image 5
   Software Compatibility 5
      Spine/Leaf Topology 5
      Modular Software Design 6
   Serviceability 6
      Switched Port Analyzer 6
      Ethalyzer 6
      Smart Call Home 6
      Online Diagnostics 7
      Embedded Event Manager 7
   Manageability 7
      Simple Network Management Protocol 7
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Special Characters</td>
<td>64</td>
</tr>
<tr>
<td>Keystroke Shortcuts</td>
<td>64</td>
</tr>
<tr>
<td>Abbreviating Commands</td>
<td>66</td>
</tr>
<tr>
<td>Completing a Partial Command Name</td>
<td>67</td>
</tr>
<tr>
<td>Identifying Your Location in the Command Hierarchy</td>
<td>67</td>
</tr>
<tr>
<td>Using the no Form of a Command</td>
<td>68</td>
</tr>
<tr>
<td>Configuring CLI Variables</td>
<td>69</td>
</tr>
<tr>
<td>About CLI Variables</td>
<td>69</td>
</tr>
<tr>
<td>Configuring CLI Session-Only Variables</td>
<td>70</td>
</tr>
<tr>
<td>Configuring Persistent CLI Variables</td>
<td>70</td>
</tr>
<tr>
<td>Command Aliases</td>
<td>71</td>
</tr>
<tr>
<td>About Command Aliases</td>
<td>71</td>
</tr>
<tr>
<td>Defining Command Aliases</td>
<td>72</td>
</tr>
<tr>
<td>Configuring Command Aliases for a User Session</td>
<td>73</td>
</tr>
<tr>
<td>Command Scripts</td>
<td>73</td>
</tr>
<tr>
<td>Running a Command Script</td>
<td>73</td>
</tr>
<tr>
<td>Echoing Information to the Terminal</td>
<td>74</td>
</tr>
<tr>
<td>Delaying Command Action</td>
<td>74</td>
</tr>
<tr>
<td>Context-Sensitive Help</td>
<td>75</td>
</tr>
<tr>
<td>Understanding Regular Expressions</td>
<td>76</td>
</tr>
<tr>
<td>Special Characters</td>
<td>76</td>
</tr>
<tr>
<td>Multiple-Character Patterns</td>
<td>77</td>
</tr>
<tr>
<td>Anchoring</td>
<td>77</td>
</tr>
<tr>
<td>Searching and Filtering show Command Output</td>
<td>78</td>
</tr>
<tr>
<td>Filtering and Searching Keywords</td>
<td>78</td>
</tr>
<tr>
<td>diff Utility</td>
<td>80</td>
</tr>
<tr>
<td>grep and egrep Utilities</td>
<td>81</td>
</tr>
<tr>
<td>less Utility</td>
<td>81</td>
</tr>
<tr>
<td>Mini AWK Utility</td>
<td>81</td>
</tr>
<tr>
<td>sed Utility</td>
<td>82</td>
</tr>
<tr>
<td>sort Utility</td>
<td>82</td>
</tr>
<tr>
<td>Searching and Filtering from the --More-- Prompt</td>
<td>83</td>
</tr>
<tr>
<td>Using the Command History</td>
<td>84</td>
</tr>
<tr>
<td>Recalling a Command</td>
<td>84</td>
</tr>
</tbody>
</table>
Contents

About Basic Device Management 101
  Device Hostname 101
  Message-of-the-Day Banner 101
  Device Clock 101
  Clock Manager 102
  Time Zone and Summer Time (Daylight Saving Time) 102
  User Sessions 102
Licensing Requirements for Basic Device Management 102
Default Settings for Basic Device Parameters 102
Changing the Device Hostname 103
Configuring the MOTD Banner 103
Configuring the Time Zone 104
Configuring Summer Time (Daylight Saving Time) 105
Manually Setting the Device Clock 106
Setting the Clock Manager 107
Managing Users 108
  Displaying Information about the User Sessions 108
  Sending a Message to Users 108
Verifying the Device Configuration 109
Additional References for Basic Device Management 109
  Related Documents for Basic Device Management 109

CHAPTER 10
Using the Device File Systems, Directories, and Files 111
About the Device File Systems, Directories, and Files 111
  File Systems 111
  Directories 112
  Files 112
Licensing Requirements for File Systems, Directories, and Files 113
Guidelines and Limitations 113
Default Settings for File System Parameters 113
Configuring the FTP, HTTP, or TFTP Source Interface 113
Working with Directories 114
  Identifying the Current Directory 114
  Changing the Current Directory 114
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About Configuration Files</td>
<td>129</td>
</tr>
<tr>
<td>Types of Configuration Files</td>
<td>129</td>
</tr>
<tr>
<td>Licensing Requirements for Configuration Files</td>
<td>130</td>
</tr>
<tr>
<td>Guidelines and Limitations for Configuration Files</td>
<td>130</td>
</tr>
<tr>
<td>Managing Configuration Files</td>
<td>130</td>
</tr>
<tr>
<td>Saving the Running Configuration to the Startup Configuration</td>
<td>130</td>
</tr>
<tr>
<td>Copying a Configuration File to a Remote Server</td>
<td>131</td>
</tr>
<tr>
<td>Downloading the Running Configuration From a Remote Server</td>
<td>132</td>
</tr>
<tr>
<td>Downloading the Startup Configuration From a Remote Server</td>
<td>133</td>
</tr>
<tr>
<td>Copying Configuration Files to an External Flash Memory Device</td>
<td>134</td>
</tr>
<tr>
<td>Copying the Running Configuration from an External Flash Memory Device</td>
<td>135</td>
</tr>
<tr>
<td>Copying the Startup Configuration From an External Flash Memory Device</td>
<td>136</td>
</tr>
<tr>
<td>Copying Configuration Files to an Internal File System</td>
<td>136</td>
</tr>
<tr>
<td>Rolling Back to a Previous Configuration</td>
<td>137</td>
</tr>
<tr>
<td>Removing the Configuration for a Missing Module</td>
<td>138</td>
</tr>
<tr>
<td>Erasing a Configuration</td>
<td>139</td>
</tr>
<tr>
<td>Clearing Inactive Configurations</td>
<td>140</td>
</tr>
<tr>
<td>Configuration Archive and Configuration Log</td>
<td>141</td>
</tr>
<tr>
<td>Information About Configuration Archive</td>
<td>141</td>
</tr>
<tr>
<td>Configuring the Characteristics of the Configuration Archive</td>
<td>141</td>
</tr>
<tr>
<td>Information About Configuration Log</td>
<td>143</td>
</tr>
<tr>
<td>Displaying Configuration Log Entries</td>
<td>144</td>
</tr>
<tr>
<td>Verifying the Device Configuration</td>
<td>145</td>
</tr>
<tr>
<td>Examples of Working with Configuration Files</td>
<td>145</td>
</tr>
<tr>
<td>Copying Configuration Files</td>
<td>145</td>
</tr>
<tr>
<td>Backing Up Configuration Files</td>
<td>146</td>
</tr>
<tr>
<td>Rolling Back to a Previous Configuration</td>
<td>146</td>
</tr>
<tr>
<td>Additional References for Configuration Files</td>
<td>146</td>
</tr>
<tr>
<td>Related Documents for Configuration Files</td>
<td>146</td>
</tr>
</tbody>
</table>
Preface

This preface includes the following sections:

- Audience, on page xi
- Document Conventions, on page xi
- Related Documentation for Cisco Nexus 9000 Series Switches, on page xii
- Documentation Feedback, on page xii
- Communications, Services, and Additional Information, on page xii

Audience

This publication is for network administrators who install, configure, and maintain Cisco Nexus switches.

Document Conventions

Command descriptions use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bold</td>
<td>Bold text indicates the commands and keywords that you enter literally as shown.</td>
</tr>
<tr>
<td>Italic</td>
<td>Italic text indicates arguments for which you supply the values.</td>
</tr>
<tr>
<td>[x]</td>
<td>Square brackets enclose an optional element (keyword or argument).</td>
</tr>
<tr>
<td>[x</td>
<td>y]</td>
</tr>
<tr>
<td>{x</td>
<td>y}</td>
</tr>
<tr>
<td>[x {y</td>
<td>z}]</td>
</tr>
</tbody>
</table>
**Description Convention**

Indicates a variable for which you supply values, in context where italics cannot be used.

<table>
<thead>
<tr>
<th>variable</th>
<th>Indicates a variable for which you supply values, in context where italics cannot be used.</th>
</tr>
</thead>
</table>

A nonquoted set of characters. Do not use quotation marks around the string or the string includes the quotation marks.

<table>
<thead>
<tr>
<th>string</th>
<th>A nonquoted set of characters. Do not use quotation marks around the string or the string includes the quotation marks.</th>
</tr>
</thead>
</table>

Examples use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>screen font</strong></td>
<td>Terminal sessions and information the switch displays are in screen font.</td>
</tr>
<tr>
<td><strong>boldface screen font</strong></td>
<td>Information that you must enter is in boldface screen font.</td>
</tr>
<tr>
<td>italic screen font</td>
<td>Arguments for which you supply values are in italic screen font.</td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Nonprinting characters, such as passwords, are in angle brackets.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Default responses to system prompts are in square brackets.</td>
</tr>
<tr>
<td>!, #</td>
<td>An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.</td>
</tr>
</tbody>
</table>

**Related Documentation for Cisco Nexus 9000 Series Switches**

The entire Cisco Nexus 9000 Series switch documentation set is available at the following URL:


**Documentation Feedback**

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus9k-docfeedback@cisco.com. We appreciate your feedback.

**Communications, Services, and Additional Information**

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you’re looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.
**Cisco Bug Search Tool**

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.
CHAPTER 1

New and Changed Information

This chapter provides release-specific information for each new and changed feature in the Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide Release 9.3(x).

- New and Changed Information, on page 1

New and Changed Information

Table 1: New and Changed Features for Cisco NX-OS Release 9.3(x)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Changed in Release</th>
<th>Where Documented</th>
</tr>
</thead>
<tbody>
<tr>
<td>Erase Configuration</td>
<td>Added support for preserving the IPv6 address on mgmt 0 interface and the IPv6 Default Gateway in Management VRF after write erase, reload operation.</td>
<td>9.3(2)</td>
<td>Erasing a Configuration, on page 139</td>
</tr>
<tr>
<td>No new features for this release</td>
<td></td>
<td>9.3(1)</td>
<td></td>
</tr>
</tbody>
</table>
Platform Support for Fundamentals Features

This chapter defines platform support for features that are not supported across the entire suite of Cisco Platforms.

- Platform Support for Fundamentals Features, on page 3

Platform Support for Fundamentals Features

The following tables list the supported platforms for each feature and the release in which they were first introduced. See the Release Notes for details about the platforms supported in the initial product release.

PowerOn Auto Provisioning

Return to Using PowerOn Auto Provisioning, on page 25.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Supported Platform(s) or Line Cards</th>
<th>First Supported Release</th>
<th>Platform Exceptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>POAP over IPv6</td>
<td>Cisco Nexus 9504 platform switches</td>
<td>9.2(2)</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>Cisco Nexus 9508 platform switches</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Network Plug and Play

Return to Using Network Plug and Play, on page 45.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Supported Platform(s) or Line Cards</th>
<th>First Supported Release</th>
<th>Platform Exceptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>PnP on Management Port</td>
<td>Cisco Nexus 9500 switches</td>
<td>9.2(3)</td>
<td>None</td>
</tr>
<tr>
<td>PnP connectivity to the Cisco DNA Center</td>
<td>Cisco Nexus 9504 platform switches</td>
<td>9.2(3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cisco Nexus 9508 platform switches</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cisco Nexus 9516 platform switches</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Overview

This chapter contains the following sections:

- Software Image, on page 5
- Software Compatibility, on page 5
- Serviceability, on page 6
- Manageability, on page 7
- Programmability, on page 8
- Traffic Routing, Forwarding, and Management, on page 9
- Quality of Service, on page 11
- Network Security Features, on page 11
- Licensing, on page 11
- Supported Standards, on page 12

Software Image

The Cisco NX-OS software consists of one NXOS software image. This image runs on all Cisco Nexus 3400 Series switches.

Software Compatibility

The Cisco NX-OS software interoperates with Cisco products that run any variant of the Cisco IOS software. The Cisco NX-OS software also interoperates with any networking operating system that conforms to the IEEE and RFC compliance standards.

Spine/Leaf Topology

The Cisco Nexus 9000 Series switches support a two-tier spine/leaf topology.
Modular Software Design

The Cisco NX-OS software supports distributed multithreaded processing on symmetric multiprocessors (SMPs), multi-core CPUs, and distributed data module processors. The Cisco NX-OS software offloads computationally intensive tasks, such as hardware table programming, to dedicated processors distributed across the data modules. The modular processes are created on demand, each in a separate protected memory space. Processes are started and system resources are allocated only when you enable a feature. A real-time preemptive scheduler helps to ensure the timely processing of critical functions.

Serviceability

The Cisco NX-OS software has serviceability functions that allow the device to respond to network trends and events. These features help you with network planning and improving response times.

Switched Port Analyzer

The Switched Port Analyzer (SPAN) feature allows you to analyze all traffic between ports (called the SPAN source ports) by nonintrusively directing the SPAN session traffic to a SPAN destination port that has an external analyzer attached to it. For more information about SPAN, see the Cisco Nexus 9000 Series NX-OS System Management Configuration Guide.

Ethanalyzer

Ethanalyzer is a Cisco NX-OS protocol analyzer tool based on the Wireshark (formerly Ethereal) open source code. Ethanalyzer is a command-line version of Wireshark for capturing and decoding packets. You can use Ethanalyzer to troubleshoot your network and analyze the control-plane traffic. For more information about Ethanalyzer, see the Cisco Nexus 9000 Series NX-OS Troubleshooting Guide.

Smart Call Home

The Call Home feature continuously monitors hardware and software components to provide e-mail-based notification of critical system events. A versatile range of message formats is available for optimal compatibility with pager services, standard e-mail, and XML-based automated parsing applications. It offers alert grouping...
capabilities and customizable destination profiles. You can use this feature, for example, to directly page a
network support engineer, send an e-mail message to a network operations center (NOC), and employ Cisco
AutoNotify services to directly generate a case with the Cisco Technical Assistance Center (TAC). For more
information about Smart Call Home, see the Cisco Nexus 9000 Series NX-OS System Management
Configuration Guide.

Online Diagnostics

Cisco generic online diagnostics (GOLD) verify that hardware and internal data paths are operating as designed.
Boot-time diagnostics, continuous monitoring, and on-demand and scheduled tests are part of the Cisco GOLD
feature set. GOLD allows rapid fault isolation and continuous system monitoring. For information about
configuring GOLD, see the Cisco Nexus 9000 Series NX-OS System Management Configuration Guide.

Embedded Event Manager

Cisco Embedded Event Manager (EEM) is a device and system management feature that helps you to customize
behavior based on network events as they happen. For information about configuring EEM, see the Cisco
Nexus 9000 Series NX-OS System Management Configuration Guide.

Manageability

This section describes the manageability features for the Cisco Nexus 9000 Series switches.

Simple Network Management Protocol

The Cisco NX-OS software is compliant with Simple Network Management Protocol (SNMP) version 1,
version 2, and version 3. A large number of MIBs is supported. For more information about SNMP, see the Cisco
Nexus 9000 Series NX-OS System Management Configuration Guide.

Configuration Verification and Rollback

The Cisco NX-OS software allows you to verify the consistency of a configuration and the availability of
necessary hardware resources prior to committing the configuration. You can preconfigure a device and apply
the verified configuration at a later time. Configurations also include checkpoints that allow you to roll back
to a known good configuration as needed. For more information about rollbacks, see the Cisco Nexus 9000
Series NX-OS System Management Configuration Guide.

Role-Based Access Control

With role-based access control (RBAC), you can limit access to device operations by assigning roles to users.
You can customize access and restrict it to the users who require it. For more information about RBAC, see
the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Cisco NX-OS Device Configuration Methods

You can use these methods to configure Cisco NX-OS devices:
Programmability

This section describes the programmability features for the Cisco Nexus 9000 Series switches.

Python API

Python is an easy-to-learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python website: http://www.python.org/. The Python scripting capability gives programmatic access to the CLI to perform various tasks and Power-On Auto Provisioning (POAP) or Embedded Event Manager (EEM) actions. For more information about the Python API and Python scripting, see the Cisco Nexus 9000 Series NX-OS Programmability Guide.

Tcl

Tool Command Language (Tcl) is a scripting language. With Tcl, you gain more flexibility in your use of the CLI commands on the device. You can use Tcl to extract certain values in the output of a `show` command, perform switch configurations, run Cisco NX-OS commands in a loop, or define EEM policies in a script.

Cisco NX-API

The Cisco NX-API provides web-based programmatic access to the Cisco Nexus 9000 Series switches. This support is delivered through the NX-API open-source web server. The Cisco NX-API exposes the complete configuration and management capabilities of the command-line interface (CLI) through web-based APIs. You can configure the switch to publish the output of the API calls in either XML or JSON format. For more information about the Cisco NX-API, see the Cisco Nexus 9000 Series NX-OS Programmability Guide.

Note

NX-API performs authentication through a programmable authentication module (PAM) on the switch. Use cookies to reduce the number of PAM authentications and thus reduce the load on PAM.
Bash Shell

The Cisco Nexus 9000 Series switches support direct Linux shell access. With Linux shell support, you can access the Linux system on the switch in order to use Linux commands and manage the underlying system. For more information about Bash shell support, see the Cisco Nexus 9000 Series NX-OS Programmability Guide.

Broadcom Shell

The Cisco Nexus 9000 Series switch front-panel and fabric module line cards contain several Broadcom ASICs. You can use the CLI to access the command-line shell (bcm shell) for these ASICs. The benefit of using this method to access the bcm shell is that you can use Cisco NX-OS command extensions such as pipe, include, and redirect output to file to manage the output. In addition, the activity is recorded in the system accounting log for audit purposes, unlike commands entered directly from the bcm shell, which are not recorded in the accounting log. For more information about Broadcom shell support, see the Cisco Nexus 9000 Series NX-OS Programmability Guide.

⚠️ Caution

Use Broadcom shell commands with caution and only under the direct supervision or request of Cisco Support personnel.

Traffic Routing, Forwarding, and Management

This section describes the traffic routing, forwarding, and management features supported by the Cisco NX-OS software.

Ethernet Switching

The Cisco NX-OS software supports high-density, high-performance Ethernet systems and provides the following Ethernet switching features:

- IEEE 802.1D-2004 Rapid and Multiple Spanning Tree Protocols (802.1w and 802.1s)
- IEEE 802.1Q VLANs and trunks
- IEEE 802.3ad link aggregation
- Unidirectional Link Detection (UDLD) in aggressive and standard modes

For more information, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide and the Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide.

IP Routing

The Cisco NX-OS software supports IP version 4 (IPv4) and IP version 6 (IPv6) and the following routing protocols:

- Open Shortest Path First (OSPF) Protocol Versions 2 (IPv4) and 3 (IPv6)
• Border Gateway Protocol (BGP) (IPv4 and IPv6)
• Enhanced Interior Gateway Routing Protocol (EIGRP) (IPv4 only)
• Routing Information Protocol Version 2 (RIPv2) (IPv4 only)

The Cisco NX-OS software implementations of these protocols are fully compliant with the latest standards and include 4-byte autonomous system numbers (ASNs) and incremental shortest path first (SPF). All unicast protocols support Non-Stop Forwarding Graceful Restart (NSF-GR). All protocols support all interface types, including Ethernet interfaces, VLAN interfaces, subinterfaces, port channels, and loopback interfaces.

For more information, see the Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide.

**IP Services**

The following IP services are available in the Cisco NX-OS software:

• Virtual routing and forwarding (VRF)
• Dynamic Host Configuration Protocol (DHCP) helper
• Hot Standby Router Protocol (HSRP)
• Enhanced object tracking
• Policy-based routing (PBR)
• Unicast graceful restart for all protocols in IPv4 unicast graceful restart for OPSFv3 in IPv6

For more information, see the Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide.

**IP Multicast**

The Cisco NX-OS software includes the following multicast protocols and functions:

• Protocol Independent Multicast (PIM) Version 2 (PIMv2)
• PIM sparse mode (Any-Source Multicast [ASM] for IPv4)
• Anycast rendezvous point (Anycast-RP)
• Multicast NSF for IPv4
• RP-Discovery using bootstrap router (BSR) (Auto-RP and static)
• Internet Group Management Protocol (IGMP) Versions 1, 2, and 3 router role
• IGMPv2 host mode
• IGMP snooping
• Multicast Source Discovery Protocol (MSDP) (for IPv4)
The Cisco NX-OS software does not support PIM dense mode.

For more information, see the Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide.

Quality of Service

The Cisco NX-OS software supports quality of service (QoS) functions for classification, marking, queuing, policing, and scheduling. Modular QoS CLI (MQC) supports all QoS features. You can use MQC to provide uniform configurations across various Cisco platforms. For more information, see the Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide.

Network Security Features

The Cisco NX-OS software includes the following security features:

- Control Plane Policing (CoPP)
- Message-digest algorithm 5 (MD5) routing protocol authentication
- Authentication, authorization, and accounting (AAA)
- RADIUS and TACACS+
- SSH Protocol Version 2
- SNMPv3
- Policies based on MAC and IPv4 addresses supported by named ACLs (port-based ACLs [PACLs], VLAN-based ACLs [VACLs], and router-based ACLs [RACLs])
- Traffic storm control (unicast, multicast, and broadcast)

For more information, see the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Licensing

The Cisco NX-OS software licensing feature allows you to access premium features on the device after you install the appropriate license for that feature. Any feature not included in a license package is bundled with the Cisco NX-OS software and is provided to you at no extra charge.

You must purchase and install a license for each device.

For detailed information about Cisco NX-OS software licensing, see the Cisco NX-OS Licensing Guide.

For information about troubleshooting licensing issues, see the Cisco Nexus 9000 Series NX-OS Troubleshooting Guide.
Supported Standards

This table lists the IEEE compliance standards.

Table 2: IEEE Compliance Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.1D</td>
<td>MAC Bridges</td>
</tr>
<tr>
<td>802.1p</td>
<td>Class of Service Tagging for Ethernet frames</td>
</tr>
<tr>
<td>802.1Q</td>
<td>VLAN Tagging</td>
</tr>
<tr>
<td>802.1s</td>
<td>Multiple Spanning Tree Protocol</td>
</tr>
<tr>
<td>802.1w</td>
<td>Rapid Spanning Tree Protocol</td>
</tr>
<tr>
<td>802.3ab</td>
<td>1000Base-T (10/100/1000 Ethernet over copper)</td>
</tr>
<tr>
<td>802.3ad</td>
<td>Link aggregation with LACP</td>
</tr>
<tr>
<td>802.3ae</td>
<td>10-Gigabit Ethernet</td>
</tr>
</tbody>
</table>

This table lists the RFC compliance standards. For information on each RFC, see [www.ietf.org](http://www.ietf.org).

Table 3: RFC Compliance Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BGP</td>
<td></td>
</tr>
<tr>
<td>RFC 1997</td>
<td>BGP Communities Attribute</td>
</tr>
<tr>
<td>RFC 2385</td>
<td>Protection of BGP Sessions via the TCP MD5 Signature Option</td>
</tr>
<tr>
<td>RFC 2439</td>
<td>BGP Route flap damping</td>
</tr>
<tr>
<td>RFC 2519</td>
<td>A Framework for Inter-Domain Route Aggregation</td>
</tr>
<tr>
<td>RFC 2545</td>
<td>Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing</td>
</tr>
<tr>
<td>RFC 2858</td>
<td>Multiprotocol Extensions for BGP-4</td>
</tr>
<tr>
<td>RFC 2918</td>
<td>Route Refresh Capability for BGP-4</td>
</tr>
<tr>
<td>RFC 3065</td>
<td>Autonomous System Confederations for BGP</td>
</tr>
<tr>
<td>Standard</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RFC 3392</td>
<td>Capabilities Advertisement with BGP-4</td>
</tr>
<tr>
<td>RFC 4271</td>
<td>BGP version 4</td>
</tr>
<tr>
<td>RFC 4273</td>
<td>BGP4 MIB - Definitions of Managed Objects for BGP-4</td>
</tr>
<tr>
<td>RFC 4456</td>
<td>BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)</td>
</tr>
<tr>
<td>RFC 4486</td>
<td>Subcodes for BGP cease notification message</td>
</tr>
<tr>
<td>RFC 4724</td>
<td>Graceful Restart Mechanism for BGP</td>
</tr>
<tr>
<td>RFC 4893</td>
<td>BGP Support for Four-octet AS Number Space</td>
</tr>
<tr>
<td>RFC 5004</td>
<td>Avoid BGP Best Path Transitions from One External to Another</td>
</tr>
<tr>
<td>RFC 5396</td>
<td>Textual Representation of Autonomous System (AS) Numbers</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> RFC 5396 is partially supported. The asplain and asdot notations are supported, but the asdot+ notation is not.</td>
</tr>
<tr>
<td>RFC 5549</td>
<td>Advertising IPv4 Network Layer Reachability Information with an IPv6 Next Hop</td>
</tr>
<tr>
<td>RFC 5668</td>
<td>4-Octet AS Specific BGP Extended Community</td>
</tr>
<tr>
<td>ietf-draft</td>
<td>Bestpath transition avoidance (draft-ietf-idr-avoid-transition-05.txt)</td>
</tr>
<tr>
<td>ietf-draft</td>
<td>Peer table objects (draft-ietf-idr-bgp4-mib-15.txt)</td>
</tr>
<tr>
<td>ietf-draft</td>
<td>Dynamic Capability (draft-ietf-idr-dynamic-cap-03.txt)</td>
</tr>
<tr>
<td>IP Multicast</td>
<td></td>
</tr>
</tbody>
</table>
## Supported Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2236</td>
<td>Internet Group Management Protocol, Version 2</td>
</tr>
<tr>
<td>RFC 3376</td>
<td>Internet Group Management Protocol, Version 3</td>
</tr>
<tr>
<td>RFC 3446</td>
<td>Anycast Rendezvous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)</td>
</tr>
<tr>
<td>RFC 3569</td>
<td>An Overview of Source-Specific Multicast (SSM)</td>
</tr>
<tr>
<td>RFC 3618</td>
<td>Multicast Source Discovery Protocol (MSDP)</td>
</tr>
<tr>
<td>RFC 4607</td>
<td>Source-Specific Multicast for IP</td>
</tr>
<tr>
<td>RFC 4610</td>
<td>Anycast-RP Using Protocol Independent Multicast (PIM)</td>
</tr>
<tr>
<td>RFC 6187</td>
<td>X.509v3 Certificates for Secure Shell Authentication</td>
</tr>
<tr>
<td>ietf-draft</td>
<td>Mtrace server functionality, to process mtrace-requests, draft-ietf-idmr-traceroute-ipm-07.txt</td>
</tr>
</tbody>
</table>

### IP Services

| RFC 768 | UDP |
| RFC 783 | TFTP |
| RFC 791 | IP  |
| RFC 792 | ICMP|
| RFC 793 | TCP |
| RFC 826 | ARP |
| RFC 854 | Telnet |
| RFC 959 | FTP |
| RFC 1027 | Proxy ARP |
## Supported Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 1305</td>
<td>NTP v3</td>
</tr>
<tr>
<td>RFC 1519</td>
<td>CIDR</td>
</tr>
<tr>
<td>RFC 1542</td>
<td>BootP relay</td>
</tr>
<tr>
<td>RFC 1591</td>
<td>DNS client</td>
</tr>
<tr>
<td>RFC 1812</td>
<td>IPv4 routers</td>
</tr>
<tr>
<td>RFC 2131</td>
<td>DHCP Helper</td>
</tr>
<tr>
<td>RFC 2338</td>
<td>VRRP</td>
</tr>
<tr>
<td>IS-IS</td>
<td></td>
</tr>
<tr>
<td>RFC 1142</td>
<td>OSI 10589 Intermediate system to intermediate system intra-domain routing exchange protocol</td>
</tr>
<tr>
<td>RFC 1195</td>
<td>Use of OSI IS-IS for routing in TCP/IP and dual environment</td>
</tr>
<tr>
<td>RFC 2763</td>
<td>Dynamic Hostname Exchange Mechanism for IS-IS</td>
</tr>
<tr>
<td>RFC 2966</td>
<td>Domain-wide Prefix Distribution with Two-Level IS-IS</td>
</tr>
<tr>
<td>RFC 2973</td>
<td>IS-IS Mesh Groups</td>
</tr>
<tr>
<td>RFC 3277</td>
<td>IS-IS Transient Blackhole Avoidance</td>
</tr>
<tr>
<td>RFC 3373</td>
<td>Three-Way Handshake for IS-IS Point-to-Point Adjacencies</td>
</tr>
<tr>
<td>RFC 3567</td>
<td>IS-IS Cryptographic Authentication</td>
</tr>
<tr>
<td>RFC 3847</td>
<td>Restart Signaling for IS-IS</td>
</tr>
<tr>
<td>ietf-draft</td>
<td>Internet Draft Point-to-point operation over LAN in link-state routing protocols (draft-ietf-isis-igp-p2p-over-lan-06.txt)</td>
</tr>
<tr>
<td>OSPF</td>
<td></td>
</tr>
<tr>
<td>RFC 2328</td>
<td>OSPF Version 2</td>
</tr>
<tr>
<td>RFC 2370</td>
<td>OSPF Opaque LSA Option</td>
</tr>
<tr>
<td>RFC 2740</td>
<td>OSPF for IPv6 (OSPF version 3)</td>
</tr>
<tr>
<td>Standard</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>RFC 3101</td>
<td>OSPF Not-So-Stubby-Area (NSSA) Option</td>
</tr>
<tr>
<td>RFC 3137</td>
<td>OSPF Stub Router Advertisement</td>
</tr>
<tr>
<td>RFC 3509</td>
<td>Alternative Implementations of OSPF Area Border Routers</td>
</tr>
<tr>
<td>RFC 3623</td>
<td>Graceful OSPF Restart</td>
</tr>
<tr>
<td>RFC 4750</td>
<td>OSPF Version 2 MIB</td>
</tr>
<tr>
<td>Per-Hop Behavior (PHB)</td>
<td></td>
</tr>
<tr>
<td>RFC 2597</td>
<td>Assured Forwarding PHB Group</td>
</tr>
<tr>
<td>RFC 3246</td>
<td>An Expedited Forwarding PHB</td>
</tr>
<tr>
<td>RIP</td>
<td></td>
</tr>
<tr>
<td>RFC 1724</td>
<td>RIPv2 MIB extension</td>
</tr>
<tr>
<td>RFC 2082</td>
<td>RIPv2 MD5 Authentication</td>
</tr>
<tr>
<td>RFC 2453</td>
<td>RIPv2 Version 2</td>
</tr>
<tr>
<td>SNMP</td>
<td></td>
</tr>
<tr>
<td>RFC 2579</td>
<td>Textual Conventions for SMIv2</td>
</tr>
<tr>
<td>RFC 2819</td>
<td>Remote Network Monitoring Management Information Base</td>
</tr>
<tr>
<td>RFC 2863</td>
<td>The Interfaces Group MIB</td>
</tr>
<tr>
<td>RFC 3164</td>
<td>The BSD syslog Protocol</td>
</tr>
<tr>
<td>RFC 3176</td>
<td>InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks</td>
</tr>
<tr>
<td>RFC 3413</td>
<td>Simple Network Management Protocol (SNMP) Applications</td>
</tr>
<tr>
<td>RFC 3417</td>
<td>Transport Mappings for the Simple Network Management Protocol (SNMP)</td>
</tr>
</tbody>
</table>
Using the Cisco NX-OS Setup Utility

This chapter contains the following sections:

- About the Cisco NX-OS Setup Utility, on page 17
- Prerequisites for the Setup Utility, on page 18
- Setting Up Your Cisco NX-OS Device, on page 19
- Additional References for the Setup Utility, on page 23

About the Cisco NX-OS Setup Utility

The Cisco NX-OS setup utility is an interactive command-line interface (CLI) mode that guides you through a basic (also called a startup) configuration of the system. The setup utility allows you to configure only enough connectivity for system management.

The setup utility allows you to build an initial configuration file using the System Configuration Dialog. The setup starts automatically when a device has no configuration file in NVRAM. The dialog guides you through initial configuration. After the file is created, you can use the CLI to perform additional configuration.

You can press Ctrl-C at any prompt to skip the remaining configuration options and proceed with what you have configured up to that point, except for the administrator password. If you want to skip answers to any questions, press Enter. If a default answer is not available (for example, the device hostname), the device uses what was previously configured and skips to the next question.
This figure shows how to enter and exit the setup script.

You use the setup utility mainly for configuring the system initially, when no configuration is present. However, you can use the setup utility at any time for basic device configuration. The setup utility keeps the configured values when you skip steps in the script. For example, if you have already configured the mgmt0 interface, the setup utility does not change that configuration if you skip that step. However, if there is a default value for the step, the setup utility changes to the configuration using that default, not the configured value. Be sure to carefully check the configuration changes before you save the configuration.

---

**Note**  
Be sure to configure the IPv4 route, the default network IPv4 address, and the default gateway IPv4 address to enable SNMP access. If you enable IPv4 routing, the device uses the IPv4 route and the default network IPv4 address. If IPv4 routing is disabled, the device uses the default gateway IPv4 address.

**Note**  
The setup script only supports IPv4.

### Prerequisites for the Setup Utility

The setup utility has the following prerequisites:
• Have a password strategy for your network environment.
• Connect the console port on the supervisor module to the network. If you have dual supervisor modules, connect the console ports on both supervisor modules to the network.
• Connect the Ethernet management port on the supervisor module to the network. If you have dual supervisor modules, connect the Ethernet management ports on both supervisor modules to the network.

Setting Up Your Cisco NX-OS Device

To configure basic management of the Cisco NX-OS device using the setup utility, follow these steps:

Step 1  Power on the device.
Step 2  Enable or disable password-strength checking.

A strong password has the following characteristics:
• At least eight characters long
• Does not contain many consecutive characters (such as "abcd")
• Does not contain many repeating characters (such as "aaabbb")
• Does not contain dictionary words
• Does not contain proper names
• Contains both uppercase and lowercase characters
• Contains numbers

Example:

---- System Admin Account Setup ----

Do you want to enforce secure password standard (yes/no) [y]: y

Step 3  Enter the new password for the administrator.

Note  If a password is trivial (such as a short, easy-to-decipher password), your password configuration is rejected. Passwords are case sensitive. Be sure to configure a strong password that has at least eight characters, both uppercase and lowercase letters, and numbers.

Example:

Enter the password for "admin": <password>

Confirm the password for "admin": <password>

---- Basic System Configuration Dialog ----

This setup utility will guide you through the basic configuration of the system. Setup configures only enough connectivity for management of the system.
Please register Cisco Nexus 9000 Family devices promptly with your supplier. Failure to register may affect response times for initial service calls. Nexus devices must be registered to receive entitled support services.

Press Enter at anytime to skip a dialog. Use ctrl-c at anytime to skip the remaining dialogs.

**Step 4**  
Enter the setup mode by entering **yes**.

*Example:*  

Would you like to enter the basic configuration dialog (yes/no): **yes**

**Step 5**  
Create additional accounts by entering **yes** (*no* is the default).

*Example:*  

Create another login account (yes/no) [n]: **yes**

a) Enter the user login ID.

*Example:*  

Enter the User login Id : **user_login**

*Caution*  
Usernames must begin with an alphanumeric character and can contain only these special characters: (+ = . _ \ -). The # and ! symbols are not supported. If the username contains characters that are not allowed, the specified user is unable to log in.

b) Enter the user password.

*Example:*  

Enter the password for "user1": **user_password**  
Confirm the password for "user1": **user_password**

c) Enter the default user role.

*Example:*  

Enter the user role (network-operator|network-admin) [network-operator]: **default_user_role**

For information on the default user roles, see the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

**Step 6**  
Configure an SNMP community string by entering **yes**.

*Example:*  

Configure read-only SNMP community string (yes/no) [n]: **yes**  
SNMP community string : **snmp_community_string**

For information on SNMP, see the Cisco Nexus 9000 Series NX-OS System Management Configuration Guide.
Step 7  Enter a name for the device (the default name is switch).

**Example:**

Enter the switch name: switch_name

Step 8  Configure out-of-band management by entering yes. You can then enter the mgmt0 IPv4 address and subnet mask.

**Note** You can only configure IPv4 address in the setup utility. For information on configuring IPv6, see the Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide.

**Example:**

Continue with Out-of-band (mgmt0) management configuration? (yes/no): yes
Mgmt0 IPv4 address: mgmt0_ip_address
Mgmt0 IPv4 netmask: mgmt0_subnet_mask

Step 9  Configure the IPv4 default gateway (recommended) by entering yes. You can then enter its IP address.

**Example:**

Configure the default-gateway: (yes/no) [y]: yes
IPv4 address of the default-gateway: default_gateway

Step 10 Configure advanced IP options such as the static routes, default network, DNS, and domain name by entering yes.

**Example:**

Configure Advanced IP options (yes/no)? [n]: yes

Step 11 Configure a static route (recommended) by entering yes. You can then enter its destination prefix, destination prefix mask, and next hop IP address.

**Example:**

Configure static route: (yes/no) [y]: yes
Destination prefix: dest_prefix
Destination prefix mask: dest_mask
Next hop ip address: next_hop_address

Step 12 Configure the default network (recommended) by entering yes. You can then enter its IPv4 address.

**Note** The default network IPv4 address is the same as the destination prefix in the static route configuration.

**Example:**

Configure the default network: (yes/no) [y]: yes
Default network IP address [dest_prefix]: dest_prefix

Step 13 Configure the DNS IPv4 address by entering yes. You can then enter the address.

**Example:**
Configure the DNS IP address? (yes/no) [y]: yes
DNS IP address: ipv4_address

Step 14 Configure the default domain name by entering yes. You can then enter the name.

Example:

Configure the DNS IP address? (yes/no) [y]: yes
DNS IP address: ipv4_address

Step 15 Enable the Telnet service by entering yes.

Example:

Enable the telnet service? (yes/no) [y]: yes

Step 16 Enable the SSH service by entering yes. You can then enter the key type and number of key bits. For more information, see the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Example:

Enable the ssh service? (yes/no) [y]: yes
Type of ssh key you would like to generate (dsa/rsa) : key_type
Number of key bits <768-2048> : number_of_bits

Step 17 Configure the NTP server by entering yes. You can then enter its IP address. For more information, see the Cisco Nexus 9000 Series NX-OS System Management Configuration Guide.

Example:

Configure NTP server? (yes/no) [n]: yes
NTP server IP address: ntp_server_IP_address

Step 18 Specify a default interface layer (L2 or L3).

Example:

Configure default interface layer (L3/L2) [L3]: interface_layer

Step 19 Enter the default switchport interface state (shutdown or no shutdown). A shutdown interface is in an administratively down state. For more information, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

Example:

Configure default switchport interface state (shut/noshut) [shut]: default_state

Step 20 Enter the best practices profile for control plane policing (CoPP). For more information, see the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Example:
Configure best practices CoPP profile (strict/moderate/lenient/none) [strict]: policy

The system now summarizes the complete configuration and asks if you want to edit it.

**Step 21** Continue to the next step by entering **no**. If you enter **yes**, the setup utility returns to the beginning of the setup and repeats each step.

**Example:**

Would you like to edit the configuration? (yes/no) [y]: yes

**Step 22** Use and save this configuration by entering **yes**. If you do not save the configuration at this point, none of your changes are part of the configuration the next time the device reboots. Enter **yes** to save the new configuration. This step ensures that the boot variables for the nx-os image are also automatically configured.

**Example:**

Use this configuration and save it? (yes/no) [y]: yes

**Caution** If you do not save the configuration at this point, none of your changes are part of the configuration the next time that the device reboots. Enter **yes** to save the new configuration to ensure that the boot variables for the nx-os image are also automatically configured.

---

**Additional References for the Setup Utility**

This section includes additional information related to using the setup utility.

**Related Documents for the Setup Utility**

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensing</td>
<td>Cisco NX-OS Licensing Guide</td>
</tr>
<tr>
<td>SSH and Telnet</td>
<td>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</td>
</tr>
<tr>
<td>User roles</td>
<td>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</td>
</tr>
<tr>
<td>IPv4 and IPv6</td>
<td>Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide</td>
</tr>
<tr>
<td>SNMP and NTP</td>
<td>Cisco Nexus 9000 Series NX-OS System Management Configuration Guide</td>
</tr>
</tbody>
</table>
Using PowerOn Auto Provisioning

This chapter contains the following sections:

- About PowerOn Auto Provisioning, on page 25
- Guidelines and Limitations for POAP, on page 35
- Setting Up the Network Environment to Use POAP, on page 36
- Configuring a Switch Using POAP, on page 37
- Creating md5 Files, on page 37
- Verifying the Device Configuration, on page 39
- Troubleshooting for POAP, on page 39
- Managing the POAP Personality, on page 40

About PowerOn Auto Provisioning

PowerOn Auto Provisioning (POAP) automates the process of upgrading software images and installing configuration files on devices that are being deployed in the network for the first time.

When a device with the POAP feature boots and does not find the startup configuration, the device enters POAP mode, locates a DHCP server, and bootstrap itself with its interface IP address, gateway, and DNS server IP addresses. The device also obtains the IP address of a TFTP server and downloads a configuration script that enables the switch to download and install the appropriate software image and configuration file.

Note

The DHCP information is used only during the POAP process.

Network Requirements for POAP

POAP requires the following network infrastructure:

- A DHCP server to bootstrap the interface IP address, gateway address, and Domain Name System (DNS) server.

- A TFTP server that contains the configuration script used to automate the software image installation and configuration process.

- One or more servers that contain the desired software images and configuration files.
• If you use USB, then no DHCP server or TFTP server are required for POAP.

Figure 3: POAP Network Infrastructure

Disabling POAP

POAP is enabled when there is no configuration in the system. It runs as a part of bootup. However, you can bypass POAP enablement during initial setup. If you want to disable POAP permanently (even when there is no configuration in the system), you can use the 'system no poap' command. This command ensures that POAP is not started during the next boot (even if there is no configuration). To enable POAP, use the 'system poap' command or the 'write erase poap' command. The 'write erase poap' command erases the POAP flag and enables POAP.

• Example: Disabling POAP

```
switch# system no poap
switch# sh boot
Current Boot Variables:
sup-1
NXOS variable = bootflash:/nxos.9.2.1.125.bin
Boot POAP Disabled

POAP permanently disabled using 'system no poap'

Boot Variables on next reload:
sup-1
NXOS variable = bootflash:/nxos.9.2.1.125.bin
Boot POAP Disabled

POAP permanently disabled using 'system no poap'
```
switch# sh system poap
System-wide POAP is disabled using exec command 'system no poap'
POAP will be bypassed on write-erase reload.
(Perpetual POAP cannot be enabled when system-wide POAP is disabled)

• Example: Enabling POAP

switch# system poap

switch# sh system poap
System-wide POAP is enabled

• Example: Erase POAP

switch# write erase poap
This command will erase the system wide POAP disable flag only if it is set.
Do you wish to proceed anyway? (y/n) [n] y
System wide POAP disable flag erased.

switch# sh system poap
System-wide POAP is enabled

POAP Configuration Script

We provide a sample configuration script that is developed using the Python programming language. We recommend using the provided script and modifying it to meet the requirements of your network environment.

The POAP script can be found at https://github.com/datacenter/nexus9000/blob/master/nx-os/poap/poap.py.

To modify the script using Python, see the Cisco NX-OS Python API Reference Guide for your platform.

Using the POAP Script and POAP Script Options

Before using the POAP script, perform the following actions:

1. Edit the options dictionary at the top of the script to ensure that all relevant options for your setup are included in the script. Do not change the defaults (in the default options function) directly.

2. Update the MD5 checksum of the POAP script as shown using shell commands.

   ```
   f=poap_nexus_script.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*$/#md5sum="$(md5sum $f.md5 | sed 's/ .*/' '/')"" $f
   ```

3. If the device has a startup configuration, perform a write erase and reload the device.

The following POAP script options can be specified to alter the POAP script behavior. When you download files from a server, the hostname, username, and password options are required. For every mode except personality, the target_system_image is also required. Required parameters are enforced by the script, and the script aborts if the required parameters are not present. Every option except hostname, username, and password has a default option. If you do not specify the option in the options dictionary, the default is used.

• **username**
The username to use when downloading files from the server.

- **password**
  The password to use when downloading files from the server.

- **hostname**
  The name or address of the server from which to download files.

- **mode**
  The default is `serial_number`.
  Use one of the following options:

  - **personality**
    A method to restore the switch from a tarball.

  - **serial_number**
    The serial number of the switch to determine the configuration filename. The format for the serial number in the configuration file is `conf.serialnumber`. Example: `conf.FOC123456`

  - **hostname**
    The hostname as received in the DHCP options to determine the configuration filename. The format for the hostname in the configuration file is `conf_hostname.cfg`. Example: `conf_3164-RS.cfg`

  - **mac**
    The interface MAC address to determine the configuration filename. The format for the hostname in the configuration file is `conf_macaddress.cfg`. Example: `conf_7426CC5C9180.cfg`

  - **raw**
    The configuration filename is used exactly as provided in the options. The filename is not altered in any way.

  - **location**
    The CDP neighbors are used to determine the configuration filename. The format for the location in the configuration file is `conf_host_intf.cfg`, where host is the host connected to the device over the POAP interface, and intf is the remote interface to which the POAP interface is connected. Example: `conf_remote-switch_Eth1_8.cfg`

  - **required_space**
    The required space in KB for that particular iteration of POAP. The default is 100,000. For multi-step upgrades, specify the size of the last image in the upgrade path of the target image.

  - **transfer_protocol**
    Any transfer protocol such as http, https, ftp, scp, sftp, or tftp that is supported by VSH. The default is scp.

  - **config_path**
    The path to the configuration file on the server. Example: `/tftpboot`. The default is `/var/lib/tftpboot`.

  - **target_system_image**
The name of the image to download from the remote server. This is the image you get after POAP completes. This option is a required parameter for every mode except personality. The default is "".

- **target_image_path**
  The path to the image on the server. Example: /tftpboot. The default is /var/lib/tftpboot.

- **destination_path**
  The path to which to download images and MD5 sums. The default is /bootflash.

- **destination_system_image**
  The name for the destination image filename. If not specified, the default will be the target_system_image name.

- **user_app_path**
  The path on the server where the user scripts, agents, and user data are located. The default is /var/lib/tftpboot.

- **disable_md5**
  This is True if MD5 checking should be disabled. The default is False.

- **midway_system_image**
  The name of the image to use for the midway system upgrade. By default, the POAP script finds the name of any required midway images in the upgrade path and uses them. Set this option if you prefer to pick a different midway image for a two-step upgrade. The default is "".

- **source_config_file**
  The name of the configuration file when raw mode is used. The default is poap.cfg.

- **vrf**
  The VRF to use for downloads and so on. The VRF is automatically set by the POAP process. The default is the POAP_VRF environment variable.

- **destination_config**
  The name to use for the downloaded configuration. The default is poap_replay.cfg.

- **split_config_first**
  The name to use for the first configuration portion if the configuration needs to be split. It is applicable only when the configuration requires a reload to take effect. The default is poap_1.cfg.

- **split_config_second**
  The name to use for the second configuration portion if the configuration is split. The default is poap_2.cfg.

- **timeout_config**
  The timeout in seconds for copying the configuration file. The default is 120. For non-legacy images, this option is not used, and the POAP process times out. For legacy images, FTP uses this timeout for the login process and not for the copy process, while scp and other protocols use this timeout for the copy process.

- **timeout_copy_system**
The timeout in seconds for copying the system image. The default is 2100. For non-legacy images, this option is not used, and the POAP process times out. For legacy images, FTP uses this timeout for the login process and not for the copy process, while scp and other protocols use this timeout for the copy process.

- **timeout_copy_personality**

  The timeout in seconds for copying the personality tarball. The default is 900. For non-legacy images, this option is not used, and the POAP process times out. For legacy images, FTP uses this timeout for the login process and not for the copy process, while scp and other protocols use this timeout for the copy process.

- **timeout_copy_user**

  The timeout in seconds for copying any user scripts and agents. The default is 900. For non-legacy images, this option is not used, and the POAP process times out. For legacy images, FTP uses this timeout for the login process and not for the copy process, while scp and other protocols use this timeout for the copy process.

- **personality_path**

  The remote path from which to download the personality tarball. Once the tarball is downloaded and the personality process is started, the personality will download all files in the future from locations specified inside the tarball configuration. The default is /var/lib/tftpboot.

- **source_tarball**

  The name of the personality tarball to download. The default is personality.tar.

- **destination_tarball**

  The name for the downloaded personality tarball after it is downloaded. The default is personality.tar.

### Setting up the DHCP Server without DNS for POAP

Beginning with Cisco NX-OS Release 7.0(3)I6(1), the tftp-server-name can be used without the DNS option. To enable POAP functionality without DNS on earlier releases, a custom option of 150 must be used to specify the tftp-server-address.

To use the tftp-server-address option, specify the following at the start of your dhcpd.conf file.

```
option tftp-server-address code 150 = ip-address;
```

For example:

```
host MyDevice {
    option dhcp-client-identifier "\000SAL12345678";
    fixed-address 2.1.1.10;
    option routers 2.1.1.1;
    option host-name "MyDevice";
    option bootfile-name "poap_nexus_script.py";
    option tftp-server-address 2.1.1.1;
}
```
Downloading and Using User Data, Agents, and Scripts as part of POAP

Under the options dictionary, you can find the `download_scripts_and_agents` function. If you choose to download user scripts and data, uncomment the first `poap_log` line and then use a series of `download_user_app` function calls to download each application. Since older Cisco NX-OS versions do not support recursive copy of directories, such directories must be put into a tarball (TAR archive) and then unpacked once on the switch. The parameters for the `download_scripts_and_agents` function are as follows:

- **source_path** - The path to where the file or tarball is located. This is a required parameter. Example: 
  `/var/lib/tftpboot`

- **source_file** - The name of the file to download. This is a required parameter. Example: `agents.tar`, `script.py`, and so on.

- **dest_path** - The location to download the file on the switch. Any directories that do not exist earlier will be created. This is an optional parameter. The default is `/bootflash`.

- **dest_file** - The name to give the downloaded file. This is an optional parameter. The default is unchanged `source_file`.

- **unpack** - Indicates whether a tarball exists for unpacking. Unpacking is done with `tar -xf tarfile -C /bootflash`. This is an optional parameter. The default is `False`.

- **delete_after_unpack** - Indicates whether to delete the downloaded tarball after unpack is successful. There is no effect if `unpack` is `False`. The default is `False`.

Using the download functionality, you can download all the agents and files needed to run POAP. To start the agents, you should have the configuration present in the running configuration downloaded by POAP. Then the agents, scheduler, and cron entry, along with EEM, can be used.

**POAP Process**

The POAP process has the following phases:

1. Power up
2. USB discovery
3. DHCP discovery
4. Script execution
5. Post-installation reload

Within these phases, other process and decision points occur. The following illustration shows a flow diagram of the POAP process.
Power-Up Phase

When you powerup the device for the first time, it loads the software image that is installed at manufacturing and tries to find a configuration file from which to boot. When a configuration file is not found, POAP mode starts.

During startup, a prompt appears asking if you want to abort POAP and continue with a normal setup. You can choose to exit or continue with POAP.

Note

No user intervention is required for POAP to continue. The prompt that asks if you want to abort POAP remains available until the POAP process is complete.

If you exit POAP mode, you enter the normal interactive setup script. If you continue in POAP mode, all the front-panel interfaces are set up in the default configuration.
**USB Discovery Phase**

When POAP starts, the process searches the root directory of all accessible USB devices for the POAP script file (the Python script file, poap_script.py), configuration files, and system and kickstart images.

If the script file is found on a USB device, POAP begins running the script. If the script file is not found on the USB device, POAP executes DHCP discovery. (When failures occur, the POAP process alternates between USB discovery and DHCP discovery, until POAP succeeds or you manually abort the POAP process.)

If the software image and switch configuration files specified in the configuration script are present, POAP uses those files to install the software and configure the switch. If the software image and switch configuration files are not on the USB device, POAP does some cleanup and starts DHCP phase from the beginning.

**DHCP Discovery Phase**

The switch sends out DHCP discover messages on the front-panel interfaces or the MGMT interface that solicit DHCP offers from the DHCP server or servers. (See the following figure.) The DHCP client on the Cisco Nexus switch uses the switch serial number in the client-identifier option to identify itself to the DHCP server. The DHCP server can use this identifier to send information, such as the IP address and script filename, back to the DHCP client.

POAP requires a minimum DHCP lease period of 3600 seconds (1 hour). POAP checks the DHCP lease period. If the DHCP lease period is set to less than 3600 seconds (1 hour), POAP does not complete the DHCP negotiation.

The DHCP discover message also solicits the following options from the DHCP server:

- TFTP server name or TFTP server address—The DHCP server relays the TFTP server name or TFTP server address to the DHCP client. The DHCP client uses this information to contact the TFTP server to obtain the script file.

- Bootfile name—The DHCP server relays the bootfile name to the DHCP client. The bootfile name includes the complete path to the bootfile on the TFTP server. The DHCP client uses this information to download the script file.

When multiple DHCP offers that meet the requirement are received, the one arriving first is honored and the POAP process moves to next stage. The device completes the DHCP negotiation (request and acknowledgment) with the selected DHCP server, and the DHCP server assigns an IP address to the switch. If a failure occurs in any of the subsequent steps in the POAP process, the IP address is released back to the DHCP server.

If no DHCP offers meet the requirements, the switch does not complete the DHCP negotiation (request and acknowledgment) and an IP address is not assigned.
Beginning with Cisco NX-OS Release 7.0(3)I4(1), POAP dynamically breaks out ports in an effort to detect a DHCP server behind one of the broken-out ports. Previously, the DHCP server used for POAP had to be directly connected to a normal cable because breakout cables were not supported.

POAP determines which breakout map (for example, 10gx4, 50gx2, 25gx4, or 10gx2) will bring up the link connected to the DHCP server. If breakout is not supported on any of the ports, POAP skips the dynamic breakout process. After the breakout loop completes, POAP proceeds with the DHCP discovery phase as normal.

**Note**

For more information on dynamic breakout, see the interfaces configuration guide for your device.
Script Execution Phase

After the device bootstraps itself using the information in the DHCP acknowledgement, the script file is downloaded from the TFTP server.

The switch runs the configuration script, which downloads and installs the software image and downloads a switch-specific configuration file.

However, the configuration file is not applied to the switch at this point, because the software image that currently runs on the switch might not support all of the commands in the configuration file. After the switch reboots, it begins running the new software image, if an image was installed. At that point, the configuration is applied to the switch.

Note
If the switch loses connectivity, the script stops, and the switch reloads its original software images and bootup variables.

Post-Installation Reload Phase

The switch restarts and applies (replays) the configuration on the upgraded software image. Afterward, the switch copies the running configuration to the startup configuration.

Guidelines and Limitations for POAP

POAP configuration guidelines and limitations are as follows:

- The switch software image must support POAP for this feature to function.

- POAP does not support provisioning of the switch after it has been configured and is operational. Only auto-provisioning of a switch with no startup configuration is supported.

- If you use POAP to bootstrap a Cisco Nexus device that is a part of a virtual port channel (vPC) pair using static port channels on the vPC links, the Cisco Nexus device activates all of its links when POAP starts up. The dually connected device at the end of the vPC links might start sending some or all of its traffic to the port-channel member links that are connected to the Cisco Nexus device, which causes traffic to get lost.

  To work around this issue, you can configure Link Aggregation Control Protocol (LACP) on the vPC links so that the links do not incorrectly start forwarding traffic to the Cisco Nexus device that is being bootstrapped using POAP.

- If you use POAP to bootstrap a Cisco Nexus device that is connected downstream to a Cisco Nexus 9000 Series switch through a LACP port channel, the Cisco Nexus 9000 Series switch defaults to suspend its member port if it cannot bundle it as a part of a port channel. To work around this issue, configure the Cisco Nexus 9000 Series switch to not suspend its member ports by using the no lacp suspend-individual command from interface configuration mode.

- Important POAP updates are logged in the syslog and are available from the serial console.

- Critical POAP errors are logged to the bootflash. The filename format is date-time_poap_PID_[init,1,2].log, where date-time is in the YYYYMMDD_hhmmss format and PID is the process ID.
• You can bypass the password and the basic POAP configuration by using the `skip` option at the POAP prompt. When you use the `skip` option, no password is configured for the admin user. The `copy running-config startup-config` command is blocked until a valid password is set for the admin user.

• If the `boot poap enable` command (perpetual POAP) is enabled on the switch, on a reload, a POAP boot is triggered even if there is a startup configuration present. If you do not want to use POAP in this scenario, remove the `boot poap enable` configuration by using the `no boot poap enable` command.

• Script logs are saved in the bootflash directory. The filename format is `date-time_poap_PID_script.log`, where `date-time` is in the `YYYYMMDD_hhmmss` format and `PID` is the process ID.

You can configure the format of the script log file. Script file log formats are specified in the script. The template of the script log file has a default format; however, you can choose a different format for the script execution log file.

• The POAP feature does not require a license and is enabled by default. However, for the POAP feature to function, appropriate licenses must be installed on the devices in the network before the deployment of the network.

• Beginning with NX-OS 7.0(3)I7(4), RFC 3004 (User Class Option for DHCP) is supported. This enables POAP to support user-class option 77 for DHCPv4 and user-class option 15 for DHCPv6. The text displayed for the user-class option for both DHCPv4 and DHCPv6 is "Cisco-POAP".
  
  • With RFC 3004 (User Class Option for DHCP) support, POAP over IPv6 is supported on Nexus 9000 switches.
  • Beginning with NX-OS 9.2(2), POAP over IPv6 is supported on Nexus 9504 and Nexus 9508 switches with –R line cards.

The POAP over IPv6 feature enables the POAP process to use IPv6 when IPv4 fails. The feature is designed to cycle between IPv4 and IPv6 protocols when a connection failure occurs.

• For secure POAP, ensure that DHCP snooping is enabled.

• To support POAP, set firewall rules to block unintended or malicious DHCP servers.

• To maintain system security and make POAP more secure, configure the following:
  
  • Enable DHCP snooping.
  • Set firewall rules to block unintended or malicious DHCP servers.

• POAP is supported on both MGMT ports and in-band ports.

---

**Setting Up the Network Environment to Use POAP**

**Step 1**  
Modify the basic configuration script provided by Cisco or create your own script. For information, see the Python Scripting and API Configuration Guide.

**Step 2**  
Every time you make a change to the configuration script, ensure that you recalculate the MD5 checksum by running:

```bash
# f=poap_nexus_script.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*$/\"md5sum="$(md5sum $f.md5 | sed 's/\*/\)/\"')/\" $f using a bash shell. For more information, see the Python API Reference Guide.
```
### Configuring a Switch Using POAP

#### Before you begin
Make sure that the network environment is set up to use POAP.

#### Step 1
Install the switch in the network.

#### Step 2
Power on the switch.

If no configuration file is found, the switch boots in POAP mode and displays a prompt that asks if you want to abort POAP and continue with a normal setup.

No entry is required to continue to boot in POAP mode.

#### Step 3
(Optional) If you want to exit POAP mode and enter the normal interactive setup script, enter `y` (yes).

The switch boots, and the POAP process begins.

#### What to do next
Verify the configuration.

### Creating md5 Files

Every time you make a change to the configuration script, ensure that you recalculate the MD5 checksum by running:

```
# f=poap_fabric.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*/#md5sum="$(md5sum $f.md5 | sed 's/.*\///g')"/"" $f using a bash shell.
```

This procedure replaces md5sum in `poap_fabric.py` with a new value if there was any change in that file.

#### Note
Steps 1-4 and 7-8 are needed only if you are using the BASH shell. If you have access to any other Linux server, these steps are not required.
### Creating md5 Files

**Before you begin**

Access to the BASH shell.

**Procedure**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>
| **Example:** | switch# configure terminal  
switch(config)# | |
| **Step 2** | feature bash-shell | Enable BASH shell feature. |
| **Example:** | switch(config)# feature bash-shell | |
| **Step 3** | exit | Exit configuration mode. |
| **Example:** | switch(config)# exit | |
| **Step 4** | run bash | Open Linux BASH. |
| **Example:** | switch# run bash | |
| **Step 5** | md5sum /bootflash/nxos.release_number.bin > /bootflash/nxos.release_number.bin.md5 | Creates md5sum for the .bin file. |
| **Example:** | bash-4.2$ md5sum /bootflash/nxos.7.0.3.I6.1.bin > /bootflash/nxos.7.0.3.I6.1.bin.md5 | |
| **Step 6** | md5sum /bootflash/poap.cfg > /bootflash/poap.cfg.md5 | Creates md5sum for the .cfg file. |
| **Example:** | bash-4.2$ md5sum /bootflash/poap.cfg > /bootflash/poap.cfg.md5 | |
| **Step 7** | exit | Exit the BASH shell. |
| **Example:** | switch(config)# exit | |
| **Step 8** | dir | Display the .md5 files. |
| **Example:** | switch# dir |
Verifying the Device Configuration

To verify the configuration, use one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show running-config</code></td>
<td>Displays the running configuration.</td>
</tr>
<tr>
<td><code>show startup-config</code></td>
<td>Displays the startup configuration.</td>
</tr>
<tr>
<td><code>show time-stamp running-config last-changed</code></td>
<td>Displays the timestamp when the running configuration was last changed.</td>
</tr>
</tbody>
</table>

Troubleshooting for POAP

The following is a list of known issues and suggestions while using POAP:

- Issue: POAP script execution fails immediately with no syslogs or output except for a "Script execution failed" statement.
  
  Suggestion: Use the `python script-name` command on the server and make sure there are no syntax errors. The options dictionary is a Python dictionary so each entry must be comma separated and have the key or option and the value separated by a colon.

- Issue: A `TypeError` exception occurs at various places depending on the incorrectly used option.
  
  Suggestion: Some options use integers (for example, timeouts and other numeric values). Check the options dictionary for numeric values that are enclosed in quotes. Refer to the options list for the correct usage.
Managing the POAP Personality

POAP Personality

The POAP personality feature, which is introduced in Cisco NX-OS Release 7.0(3)I4(1), enables user data, Cisco NX-OS and third-party patches, and configuration files to be backed up and restored. In previous releases, POAP can restore only the configuration.

The POAP personality is defined by tracked files on the switch. The configuration and package list in the personality file are ASCII files.

Binary versions are recorded in the personality file, but the actual binary files are not included. Because binary files are typically large, they are accessed from a specified repository.

The personality file is a .tar file, which would typically be extracted into a temporary folder. Here is an example:

```
switch# dir bootflash: 042516182843personality 
46985 Dec 06 23:12:56 2015 running-config Same as “show running-configuration” command.
20512 Dec 06 23:12:56 2015 host-package-list Package/Patches list
58056 Dec 06 23:12:56 2015 data.tar User Data
25 Dec 06 23:12:56 2015 IMAGEFILE Tracked image metadata
```

Backing Up the POAP Personality

You can create a backup of the POAP personality either locally on the switch or remotely on the server. The personality backup taken from the switch should be restored only on a switch of the same model.

---

**Note**

If you are using the Cisco scheduler feature for backups, you can configure it to also back up the POAP personality, as shown in the following example. For more information on the scheduler, see the Cisco Nexus 9000 Series NX-OS System Management Configuration Guide.

```
switch(config)# scheduler schedule name weeklybkup
switch(config-schedule)# time weekly mon:07:00
switch(config-schedule)# job name personalitybkup
switch(config-schedule)# exit
switch(config)# scheduler job name personalitybkup
switch(config-job)# personality backup bootflash:/personality-file ; copy bootflash:/personality-file tftp://10.1.1.1/ vrf management
```

---

**SUMMARY STEPS**

1. personality backup [bootflash: uri | scp: uri]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 Required: personality backup [bootflash: uri</td>
<td>scp: uri]</td>
</tr>
</tbody>
</table>
### Purpose

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch# personality backup bootflash:personality1.tar</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# personality backup scp://root@2.1.1.1/var/lib/tftpboot/backup.tar</td>
<td></td>
</tr>
</tbody>
</table>

### Configuring the POAP Personality

You can specify whether the POAP personality should be derived from the running state of the system or the committed (startup) state.

### SUMMARY STEPS

1. configure terminal
2. personality
3. track [running-state | startup-state | data local-directories-or-files]
4. binary-location source-uri-folder

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> Required: configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> Required: personality</td>
<td>Enters personality configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# personality switch(config-personality)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> Required: track [running-state</td>
<td>startup-state</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-personality)# track data bootflash:myfile1</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-personality)# track data bootflash:user_scripts/*.py</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-personality)# track data bootflash:basedir/*/backup_data</td>
<td></td>
</tr>
</tbody>
</table>

- running-state—Captures the following information: the running configuration (as shown in the show running-config command), active Cisco NX-OS patches and third-party packages in the host system, and the image name (as shown in the show version command). This is the default option.

- startup-state—Captures the following information: the startup configuration (as shown in the show startup-config command), committed Cisco NX-OS patches and third-party packages in the host system, and the image name (as shown in the show version command).
### Purpose

Command or Action | Purpose
--- | ---
- `data local-directories-or-files`—Specifies a directory or file to be backed up. You can enter this command multiple times to back up multiple directories and files. UNIX-style wildcards are supported. In the example, one folder and two directories are specified.

**Note** Do not use this command to backup binary files in the bootflash and do not point to the entire bootflash.

**Note** Guest Shell packages are not tracked.

**Note** Signed RPMs (which require a key) are not supported. The POAP personality feature does not work with signed RPMs.

### Step 4

<table>
<thead>
<tr>
<th>Required: binary-location source-uri-folder</th>
</tr>
</thead>
</table>

**Example:**

```bash
switch(config-personality)# binary-location scp://remote-dir1/nxos_patches/
```

**Example:**

```bash
switch(config-personality)# binary-location tftp://1.2.3.4/remote-dir2/packages/
```

Specifies the local or remote directory from which to pick up binary files when the POAP personality is restored. You can enter this command multiple times (in order of priority) to specify multiple locations.

### Restoring the POAP Personality

During the POAP script execution phase, the personality module in the script restores the POAP personality, provided that the currently booted switch image is Cisco NX-OS Release 7.0(3)I4(1) or later. If necessary, upgrade the switch to the correct software image.

**Note** A personality restore is done with the same software image used for the personality backup. Upgrading to a newer image is not supported through the POAP personality feature. To upgrade to a newer image, use the regular POAP script.

**Note** If the personality script fails to execute for any reason (such as not enough space in the bootflash or a script execution failure), the POAP process returns to the DHCP discovery phase.

The restore process performs the following actions:

1. Untars and unzips the personality file in the bootflash.
2. Validates the personality file.
3. Reads the configuration and package list files from the personality file to make a list of the binaries to be downloaded.

4. If the current image or patches are not the same as specified in the personality file, downloads the binaries to the bootflash (if not present) and reboots with the correct image and then applies the packages or patches.

5. Unzips or untars the user data files relative to "/".

6. Copies the configuration file in the POAP personality to the startup configuration.

7. Reboots the switch.

**POAP Personality Sample Script**

The following sample POAP script (poap.py) includes the personality feature:

```python
from poap.personality import POAPPersonality
import os

# Location to download system image files, checksums, etc.
download_path = "/var/lib/tftpboot"

# The path to the personality tarball used for restoration
personality_tarball = "/var/lib/tftpboot/foo.tar"

# The protocol to use to download images/config protocol = "scp"

# The username to download images, the personality tarball, and the patches and RPMs during restoration
username = "root"

# The password for the above username
password = "passwd754"

# The hostname or IP address of the file server
server = "2.1.1.1"

# The VRF to use for downloading and restoration
vrf = "default"

if os.environ.has_key('POAP_VRF'):
    vrf = os.environ['POAP_VRF']

# Initialize housekeeping stuff (logs, temp dirs, etc.)
p = POAPPersonality(download_path, personality_tarball, protocol, username, password, server, vrf)
p.get_personality()
p.apply_personality()
sys.exit(0)
```

CHAPTER 6

Using Network Plug and Play

This chapter contains the following sections:

- About Network Plug and Play, on page 45
- Troubleshooting Examples for Network Plug and Play, on page 53

About Network Plug and Play

Network Plug and Play (PnP) is a software application that runs on a Cisco Nexus 9500 Series Switch (specifically, N9K-C9504, N9K-C9508, and N9K-C9516). The PnP feature provides a simple, secure, unified, and integrated offering to make a new branch or campus rollouts much easier, or for provisioning updates to an existing or a new network. This feature provides a unified approach to provision networks comprising multiple devices with a near-zero-touch deployment experience.

Simplified deployment reduces the cost and complexity and increases the speed and security of the deployments. The PnP feature helps simplify the deployment of any Cisco device by automating the following deployment-related operational tasks:

- Establishing initial network connectivity for a device.
- Delivering device configuration to the controller.
- Delivering software and firmware images to the controller.
- Provisioning local credentials of a switch.
- Notifying other management systems about deployment-related events.

The PnP is a client-server based model. The client (agent) runs on a Cisco Nexus 9500 Series Switch and the server (controller) runs on the Cisco DNA Controller.

PnP uses a secure connection to communicate between the agent and the controller. This communication is encrypted.

For information on configuring and managing the needed security certificate(s) for PnP functionality, see the Cisco Digital Network Architecture Center Security Best Practices Guide.

The PnP agent converge solutions that exist in a network into a unified agent and adds additional functionality to enhance the current solutions. The main objectives of the PnP agent is to provide consistent Day 0 deployment solution for all the deployment scenarios.
Features Provided by the Network Plug and Play (PnP) Agent

Day 0 Provisioning

Day 0 bootstrapping includes the configuration, image, and other files. When a device is powered on for the first time, the PnP discovery process, which is embedded in the device, gets enabled in the absence of a startup configuration file and attempts to discover the address of the PnP controller or server. The PnP agent uses methods such as DHCP, Domain Name System (DNS), and others to acquire the desired IP address of the PnP server.

When the PnP agent successfully acquires the IP address, it initiates a long-term, bidirectional Layer 3 connection with the server and waits for a message from the server. The PnP server application sends messages to the corresponding agent, requesting for information about the devices and the services to be performed on the device.

The agent running on the Cisco Nexus 9500 Series switch then configures the IP address on receiving the DHCP acknowledgment and establishes a secure channel with the controller to provision the configurations. The switch then upgrades the image and applies the configurations.

Discovery Methods

A PnP agent discovers the PnP controller or server using one of the following methods:

- DHCP-based discovery
- DNS-based discovery
- PnP connect

After the discovery, the PnP agent writes the discovered information into a file, which is then used to handshake with the PnP server (DNA controller/DNA-C).

The following tasks are carried out by the agent in the PnP discovery phase:

- Brings up all the interfaces.
- Sends a DHCP request in parallel for all the interfaces.
- On receiving a DHCP reply, configures the IP address and mask, default route, DNS server, domain name, and writes the PnP server IP in a lease-parsing file. Note that there is no DHCP client in Cisco Nexus Switches and static configuration is required.
- Brings down all the interfaces.

Note

POAP is the first order of choice for Day 0 provisioning. Only when there is no valid POAP offer, PnP discovery is attempted. Also, PnP is supported only on Cisco Nexus 9000 EoR models N9K-C9504, N9K-C9508, and N9K-C9516. PnP is not supported on Cisco Nexus 9000 ToRs.

DHCP-Based Discovery

When the switch is powered on and if there is no startup configuration, the PnP starts with DHCP discovery. DHCP discovery obtains the PnP server connectivity details.

The PnP agent configures the following:

- IP address
- Netmask
- Default gateway
- DNS server
• Domain name

If the agent configuration fails, you should manually intervene and configure the switch.

DHCP discovery has the following flow:
• Power on the switch.
• Switch will boot up, the PnP process will be started, as there is no configuration present.
• Start DHCP discovery.
• DHCP Server replies with the PnP server configuration.
• PnP agent handshakes with the PnP server.
• Download the image, install, and reload.
• Download and apply the configuration from the controller.

A device with no startup configuration in the NV-RAM triggers the day 0 provisioning and goes through the POAP process (as detailed in m_using_poweron_auto_provisioning_92x.ditamap#id_70221). When there is no valid POAP offer, the PnP agent is initiated. The DHCP server can be configured to insert additional information using vendor-specific Option 43. Upon receiving Option 60 from the device with the string (cisco pnp), to pass on the IP address or hostname of the PnP server to the requesting device. When the DHCP response is received by the device, the PnP agent extracts the Option 43 from the response to get the IP address or the hostname of the PnP server. The PnP agent then uses this IP address or hostname to communicate with the PnP server.

**Figure 6: DHCP Discovery Process for PnP Server**

DNS-Based Discovery

When the DHCP discovery fails to get the PnP server, the agent falls back to DNS-based discovery. To start the DNS-based discovery, the following information is required from DHCP:
• IP address and netmask
• Default gateway
• DNS server IP
• Domain name

The agent obtains the domain name of the customer network from the DHCP response and constructs the fully qualified domain name (FQDN). The following FQDN is constructed by the PnP agent using a preset deployment server name and the domain name information for the DHCP response. The agent then looks up the local name server and tries to resolve the IP address for the above FQDN.

Figure 7: DNS Lookup for pnpserver.[domainname].com

The device reads domain name and creates predefined PnP server name as pnpserver.[domain name].com, for example; pnpserver.cisco.com.

Plug and Play Connect

When the DHCP and the DNS discovery fail, the PnP agent discovers and communicates with Cisco Cloud-based deployment service for initial deployment. The PnP agent directly opens an HTTPS channel using the Python library, which internally invokes OpenSSL to talk with cloud for configuration.

Cisco Power On Auto Provisioning

Cisco Power On Auto Provisioning (PoAP) communicates with the DHCP and TFTP servers to download the image and configurations. With the introduction of the PnP feature, PnP and PoAP coexist together in a Cisco Nexus 9500 Series Switch. PoAP and PnP interworking has the following processes:
• PoAP starts first when no start-up configuration is present in the system.
• PnP starts later if PoAP does not get provisioned.
• PoAP and PnP discover the controller alternatively.
• The controller discovery process continues until a controller or until the admin aborts auto provision.
• The process (POAP or PnP) that finds the controller continues provisioning and the other process that
do not find the controller is notified and eventually terminated.

Services and Capabilities of the Network Plug and Play Agent

The PnP agent performs the following tasks:
• Backoff
• Capability
• CLI execution
• Configuration upgrade
• Device information
• Certificate install
• Image install
• Redirection

Note

The PnP controller or server provides an optional checksum tag to be used in the image installation and
configuration upgrade service requests by the PnP agent. When the checksum is provided in a request, the
image install process compares the checksum against the current running image checksum.

If the checksums are same, the image being installed or upgraded is the same as the current image running
on the device. The image install process will not perform any other operation in this scenario.

If the checksums are not the same, the new image will be copied to the local file system, and the checksum
will be calculated again and compared with the checksum provided in the request. If they are the same, the
image install process continues to install the new image or upgrade the device to the new image. If the
checksums are not the same, the process exits with an error.

Backoff

A Cisco NX-OS device that supports PnP protocol, which uses HTTP transport, requires the PnP agent to
send the work request to the PnP server continuously. If the PnP server does not have any scheduled or
outstanding PnP service for the PnP agent to execute, the continuous no-operation work requests exhaust both
the network bandwidth and the device resources. This PnP backoff service allows the PnP server to inform
the PnP agent to rest for the specified time and call back later.

Capability

Capability service request is sent by the PnP server to the PnP agent on a device to query the supported services
by the agent. The server then sends an inventory service request to query the device's inventory information;
and then sends an image installation request to download an image and install it. After getting the response
from the agent, the list of supported PnP services and features are enlisted and returned back to the Server.

CLI Execution

Cisco NX-OS supports two modes of command execution, privileged EXEC mode and global configuration
mode. Most of the EXEC commands are one-time commands, such as show commands, which show the
current configuration status, and clear commands, which clear counters or interfaces. The EXEC commands
are not saved when a device reboots. Configuration mode commands allow user to make changes to the running configuration. If you save the configuration, these commands are saved when a device reboots.

Configuration Upgrade

Two types of configuration upgrades takes place in a Cisco device—copying new configuration files to the startup configuration and copying new configuration files to the running configuration.

Copying new configuration files to the startup configuration—A new configuration file is copied from the file server to the device using the copy command, and the file check task is performed to check the validity of the file. If the file is valid, the file is copied to the startup configuration. The previous configuration file is backed up if enough disk space is available. The new configuration comes into effect when the device reloads again.

Copying new configuration files to the running configuration—A new configuration file is copied from the file server to the device using the copy command or configure replace command. Replace and rollback of configuration files may leave the system in an unstable state if rollback is performed inefficiently. Therefore, configuration upgrade by copying the files is preferred.

Device Information

The PnP agent provides the capability to extract device inventory and other important information to the PnP server on request. The following device-profile request types are supported:

- all—Returns complete inventory information, which includes unique device identifier (UDI), image, hardware, and file system inventory data.
- filesystem—Returns file system inventory information, which includes file system name and type, local size (in bytes), free size (in bytes), read flag, and write flag.
- hardware—Returns hardware inventory information, which includes hostname, vendor string, platform name, processor type, hardware revision, main memory size, I/O memory size, board ID, board rework ID, processor revision, mid-plane revision, and location.
- image—Returns image inventory information, which includes version string, image name, boot variable, return to ROMMON reason, bootloader variable, configuration register, configuration register on next boot, and configuration variables.
- UDI—Returns the device UDI.

Certificate Install

Certificate install is a security service through which a PnP server requests the PnP agent on a device for trust pool or trust point certificate installation or uninstallation. This service also specifies the agent about the primary and backup servers for reconnection. The following prerequisites are required for a successful certificate installation:

- The server from which the certificate or trust pool bundle needs to be downloaded should be reachable.
- There should not be any permission issues to download the certificate or the bundle.
- The PKI API should be available and accessible for the PnP agent so that the agent could call to download and install the certificate or the bundle.
- There is enough memory on the device to save the downloaded certificate or bundle.

PnP Agent

The PnP agent is an embedded software component that is present in all Cisco network devices that support simplified deployment architecture. The PnP agent understands and interacts only with a PnP server. The PnP agent first tries to discover a PnP server, with which it can communicate. After a server is found and connection established, the agent performs deployment-related activities such as configuration, image and file updates.
by communicating with the server. It also notifies the server of all interesting deployment-related events such as out-of-band configuration changes and new device connections on an interface.

PnP Server

The PnP server is a central server that encodes the logic of managing and distributing deployment information (images, configurations, files, and licenses) for the devices being deployed. The server communicates with the agent on the device that supports the simplified deployment process using a specific deployment protocol.

Figure 8: Simplified Deployment Server

The PnP server also communicates with proxy servers such as deployment applications on smart phones and PCs, or other PnP agents acting as Neighbor Assisted Provisioning Protocol (NAPP) servers, and other types of proxy deployment servers such as VPN gateways.

The PnP server can redirect the PnP agent to another deployment server. A common example of redirection is a PnP server redirecting a device to communicate with it directly after sending the bootstrap configuration through a NAPP server. A PnP server can be hosted by an enterprise. This solution allows for a cloud-based deployment service provided by Cisco. In this case, a device discovers and communicates with Cisco cloud-based deployment service for initial deployment. After that, it can be redirected to the customer's deployment server.

In addition to communicating with the devices, the server interfaces with a variety of external systems such as authentication, authorizing, and accounting (AAA) systems, provisioning systems, and other management applications.

PnP Agent Deployment

The following steps indicate the PnP agent deployment procedure on Cisco devices:

1. A Cisco device with a PnP agent contacts the PnP server, requesting for a task, that is, the PnP agent sends UDI along with a request for work.
2. If the PnP server has a task for the device, for example, image installation, configuration, upgrade, and so on, it sends a work request.
3. After the PnP agent receives the work request, it executes the task and sends back a reply to the PnP server about the task status, that is whether it is successful or if an error has occurred, and the corresponding information that is requested.

PnP Agent Network Topology
Figure 9: Network Topology of Cisco PnP Agent Deployment

PnP Agent Initialization

The PnP agent is enabled by default, but can be initiated on a device when the startup configuration is not available.

Absence of Startup Configuration

New Cisco devices are shipped to customers with no startup configuration file in the NVRAM of the devices. When a new device is connected to a network and powered on, the absence of a startup configuration file on the device automatically triggers the PnP agent to discover the PnP server IP address.

CLI Configuration for the PnP Agent

PnP supports devices that are using VLAN 1 by default.

Guidelines and Limitations for Network Plug and Play

Network Plug and Play (PnP) guidelines and limitations are as follows:

- Beginning with NX-OS 9.2(3), PnP is supported on the management port of Cisco Nexus 9500 platform switches.
- PnP runs on both the in-band and the management interfaces. In-band is supported only on FX-series line cards (specifically N9K-X9736C-FX for PnP).
- The PnP deployment method depends on the discovery process that is required for finding the PnP controller or server.
- The discovery mechanism must be deployed, either as a DHCP server discovery process or a Domain Name Server (DNS) discovery process, before launching PnP.
• Configure the DHCP server or the DNS server before deploying PnP.
• The PnP server must communicate with the PnP agent.
• PnP connect does not require a DHCP or DNS configuration.
• IPv6 support for PnP is not available for Cisco Nexus 9500 Series devices.

Cisco DNA Center Support

The following guidelines and limitations are specific for PnP connectivity to the Cisco DNA Center:

• Cisco DNA Center supports the following functionality on the Cisco Nexus 9504, Cisco Nexus 9508, and Cisco Nexus 9516 switches:
  • Discovery
  • Inventory
  • Topology
  • Template Programmer
  • Software Image Management
  • Basic Monitoring
• The following PnP guidelines and limitations are only for the Cisco DNA Center version 1.2.6 and earlier:
  • The startup configuration that is provided during plug and play must ensure that the connectivity for the interface that is connected to the Cisco DNA Center remains intact.
  • The system image .bin and startup configuration must be uploaded to the Cisco DNA Center.
  • The bootflash must have enough space to download the image and configurations from the Cisco DNA Center.

Click here for the user documentation for the Cisco DNA Center.

Troubleshooting Examples for Network Plug and Play

Example: Troubleshooting PnP

The following examples shows the PnP troubleshooting command outputs:

```
Switch# show pnp status
PnP Agent is running
server-connection
  status: Success
time: 08:41:26 Jan 11
interface-info
  status: Success
time: 08:34:00 Jan 11
device-info
  status: Success
time: 08:33:46 Jan 11
config-upgrade
  status: Success
```
time: 08:31:36 Jan 11
capability
  status: Success
time: 08:33:50 Jan 11
backoff
  status: Success
time: 08:41:26 Jan 11
topology
  status: Success
time: 08:33:54 Jan 11

Switch# show pnp version
PnP Agent Version Summary
PnP Agent: 1.6.0
Platform Name: nxos
PnP Platform: 1.5.0.rc2

Switch# show pnp profiles
Created by UDI
DHCP Discovery  PID:N9K-C9504,VID:V01,SN:FOX1813GCZ8
  Primary transport: https
  Address: 10.105.194.248
  Port: 443
  CA file: /etc/pnp/certs/trustpoint/pnplabel
  Work-Request Tracking:
    Pending-WR: Correlator=
    Last-WR: Correlator=
  PnP Response Tracking:
    Last-PR: Correlator=

Switch# show pnp lease
{
  "lease": {
    "uptime": "Fri Jan 11 05:32:17 2019",
    "intf": "Vlan1",
    "ip_addr": "10.77.143.239",
    "mask": "255.255.255.0",
    "gw": "10.77.143.1",
    "domain": "",
    "opt_43": "5A1D;B2;K4;I10.105.194.248;J80",
    "lease": "3600",
    "server": "10.77.143.231",
    "vrf": "1"
  }
}

Switch# show pnp internal trace
1) Event:E_DEBUG, length:49, at 907122 usecs after Fri Jan 11 08:30:44 2019
   [104] pnp_ascii_gen: ascii gen completed rcode[0]
2) Event:E_DEBUG, length:16, at 907094 usecs after Fri Jan 11 08:30:44 2019
   [104] pss type: 5
3) Event:E_DEBUG, length:31, at 907069 usecs after Fri Jan 11 08:30:44 2019
   [104] Entering pnp_ascii_gen_cfg
4) Event:E_DEBUG, length:22, at 907061 usecs after Fri Jan 11 08:30:44 2019
   [104] Calling Ascii gen

5) Event:E_DEBUG, length:16, at 907051 usecs after Fri Jan 11 08:30:44 2019
   [104] pss type: 2

6) Event:E_DEBUG, length:49, at 907018 usecs after Fri Jan 11 08:30:44 2019
   [104] pnp_ascii_gen: fu_num_acfg_pss_entries[0x2]

7) Event:E_DEBUG, length:49, at 973813 usecs after Fri Jan 11 08:29:51 2019
   [104] pnp_ascii_gen: ascii gen completed rcode[0]

8) Event:E_DEBUG, length:16, at 973787 usecs after Fri Jan 11 08:29:51 2019
   [104] pss type: 5

9) Event:E_DEBUG, length:31, at 973760 usecs after Fri Jan 11 08:29:51 2019
   [104] Entering pnp_ascii_gen_cfg

10) Event:E_DEBUG, length:22, at 973751 usecs after Fri Jan 11 08:29:51 2019
    [104] Calling Ascii gen

    [104] pss type: 2

12) Event:E_DEBUG, length:49, at 973707 usecs after Fri Jan 11 08:29:51 2019
    [104] pnp_ascii_gen: fu_num_acfg_pss_entries[0x2]

13) Event:E_DEBUG, length:35, at 535794 usecs after Fri Jan 11 08:04:15 2019
    [104] pnp_pi_spawn_finalize pid 690

14) Event:E_DEBUG, length:41, at 228291 usecs after Fri Jan 11 08:04:13 2019
    [104] + pnp_pi_spawn child_pid: 0xdd526da0

15) Event:E_DEBUG, length:76, at 132853 usecs after Fri Jan 11 08:03:26 2019
    [104] Rx: Direction: PnP PI -> PnP PD, Type: Device Provisioned, Cfg: Present

16) Event:E_DEBUG, length:35, at 440380 usecs after Fri Jan 11 08:03:18 2019
    [104] !!! ACKED Unconfigure Ret:1!!

17) Event:E_DEBUG, length:61, at 440347 usecs after Fri Jan 11 08:03:18 2019
    [104] Tx: Direction: Max, Type: DHCP Unconfigure Done, Len: 16

18) Event:E_DEBUG, length:35, at 440331 usecs after Fri Jan 11 08:03:18 2019
    [102] Unknown timer cancel requested

19) Event:E_DEBUG, length:35, at 440311 usecs after Fri Jan 11 08:03:18 2019
    [104] pnp_pss_runtime_commit success

20) Event:E_DEBUG, length:57, at 440103 usecs after Fri Jan 11 08:03:18 2019
    [104] pnp_pss_runtime_commit: Stored values in runtime PSS

21) Event:E_DEBUG, length:23, at 440051 usecs after Fri Jan 11 08:03:18 2019
    [104] - pnp_vsh_halt:206

22) Event:E_DEBUG, length:17, at 950291 usecs after Fri Jan 11 08:03:15 2019
    [104] Adding "end"

23) Event:E_DEBUG, length:58, at 950269 usecs after Fri Jan 11 08:03:15 2019
    [104] Adding "configure terminal ; no clock protocol none "

24) Event:E_DEBUG, length:33, at 945994 usecs after Fri Jan 11 08:03:15 2019
    [104] - pnp_vsh_config_l3_intf:788
Have startup config, Starting PnP PI....

47) Event:E_DEBUG, length:40, at 831538 usecs after Fri Jan 11 08:01:49 2019
   Posix log directory creation failed

48) Event:E_DEBUG, length:50, at 831479 usecs after Fri Jan 11 08:01:49 2019
   pnp_fire_event: PNP_EVENT_HAVE_STARTUP_CONFIG

49) Event:E_DEBUG, length:35, at 831465 usecs after Fri Jan 11 08:01:49 2019
   Inside : pnp_other_msg_handler

50) Event:E_DEBUG, length:80, at 831368 usecs after Fri Jan 11 08:01:49 2019
   Injecting Event PNP_EVENT_HAVE_STARTUP_CONFIG

51) Event:E_DEBUG, length:50, at 831303 usecs after Fri Jan 11 08:01:49 2019
   Have Startup Config, move the process state to monitor

52) Event:E_DEBUG, length:59, at 831303 usecs after Fri Jan 11 08:01:49 2019
   Have Startup Config, move the process state to monitor

53) Event:E_DEBUG, length:57, at 799379 usecs after Fri Jan 11 08:01:49 2019
   Accelerating PnP, Break Point: Break Point PoAP Init

54) Event:E_DEBUG, length:35, at 799239 usecs after Fri Jan 11 08:01:49 2019
   pnp_pss_runtime_commit success

55) Event:E_DEBUG, length:57, at 799239 usecs after Fri Jan 11 08:01:49 2019
   pnp_pss_runtime_commit: Stored values in runtime PSS

56) Event:E_DEBUG, length:23, at 799226 usecs after Fri Jan 11 08:01:49 2019
   Phase None -> Init

57) Event:E_DEBUG, length:53, at 799200 usecs after Fri Jan 11 08:01:49 2019
   Initializing PnP-agent State machine curr_state 3

58) Event:E_DEBUG, length:35, at 799188 usecs after Fri Jan 11 08:01:49 2019
   pnp_pss_runtime_commit success

59) Event:E_DEBUG, length:57, at 799070 usecs after Fri Jan 11 08:01:49 2019
   pnp_pss_runtime_commit: Stored values in runtime PSS

60) Event:E_DEBUG, length:26, at 798965 usecs after Fri Jan 11 08:01:49 2019
   !!! Box is Online !!!

61) Event:E_DEBUG, length:35, at 798954 usecs after Fri Jan 11 08:01:49 2019
   pnp_pss_runtime_commit success

62) Event:E_DEBUG, length:57, at 798770 usecs after Fri Jan 11 08:01:49 2019
   pnp_pss_runtime_commit: Stored values in runtime PSS

   pnp_demux_mts(463): (Warning) unexpected mts msg (opcode - 7655)

64) Event:E_DEBUG, length:41, at 092701 usecs after Fri Jan 11 07:55:33 2019
   PnP Init Internal subsystem, Done!!!
/isan/etc/pnp/pnp_config.cfg:
/isan/etc/pnp/policy_discovery.conf:
/isan/etc/pnp/certs/platform.json:
/isan/etc/pnp/certs/pnp_status.json:
/isan/etc/pnp/certs/job_status.json:
Understanding the Command-Line Interface

This chapter contains the following sections:

- About the CLI Prompt, on page 59
- Command Modes, on page 60
- Special Characters, on page 64
- Keystroke Shortcuts, on page 64
- Abbreviating Commands, on page 66
- Completing a Partial Command Name, on page 67
- Identifying Your Location in the Command Hierarchy, on page 67
- Using the no Form of a Command, on page 68
- Configuring CLI Variables, on page 69
- Command Aliases, on page 71
- Command Scripts, on page 73
- Context-Sensitive Help, on page 75
- Understanding Regular Expressions, on page 76
- Searching and Filtering show Command Output, on page 78
- Searching and Filtering from the --More-- Prompt, on page 83
- Using the Command History, on page 84
- Enabling or Disabling the CLI Confirmation Prompts, on page 86
- Setting CLI Display Colors, on page 86
- Sending Commands to Modules, on page 86
- Sending Command Output in Email, on page 87
- BIOS Loader Prompt, on page 89
- Examples Using the CLI, on page 89
- Additional References for the CLI, on page 92

About the CLI Prompt

Once you have successfully accessed the device, the CLI prompt displays in the terminal window of your console port or remote workstation as shown in the following example:

User Access Verification
login: admin
Password:<password>
You can change the default device hostname.

From the CLI prompt, you can do the following:

- Use CLI commands for configuring features
- Access the command history
- Use command parsing functions

**Note**

In normal operation, usernames are case sensitive. However, when you are connected to the device through its console port, you can enter a login username in all uppercase letters regardless of how the username was defined. As long as you provide the correct password, the device logs you in.

### Command Modes

This section describes command modes in the Cisco NX-OS CLI.

#### EXEC Command Mode

When you first log in, the Cisco NX-OS software places you in EXEC mode. The commands available in EXEC mode include the show commands that display the device status and configuration information, the clear commands, and other commands that perform actions that you do not save in the device configuration.

#### Global Configuration Command Mode

Global configuration mode provides access to the broadest range of commands. The term indicates characteristics or features that affect the device as a whole. You can enter commands in global configuration mode to configure your device globally or to enter more specific configuration modes to configure specific elements such as interfaces or protocols.

### SUMMARY STEPS

1. configure terminal
DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td>The CLI prompt changes to indicate that you are in global configuration mode.</td>
</tr>
</tbody>
</table>

Interface Configuration Command Mode

One example of a specific configuration mode that you enter from global configuration mode is interface configuration mode. To configure interfaces on your device, you must specify the interface and enter interface configuration mode.

You must enable many features on a per-interface basis. Interface configuration commands modify the operation of the interfaces on the device, such as Ethernet interfaces or management interfaces (mgmt 0).

For more information about configuring interfaces, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

SUMMARY STEPS

1. configure terminal
2. interface type number

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td>The CLI prompt changes to indicate that you are in global configuration mode.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>interface type number</td>
<td>Specifies the interface that you want to configure.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>The CLI places you into interface configuration mode for the specified interface.</td>
</tr>
<tr>
<td>switch(config)# interface ethernet 2/2</td>
<td>The CLI prompt changes to indicate that you are in interface configuration mode.</td>
</tr>
</tbody>
</table>

Subinterface Configuration Command Mode

From global configuration mode, you can access a configuration submode for configuring VLAN interfaces called subinterfaces. In subinterface configuration mode, you can configure multiple virtual interfaces on a single physical interface. Subinterfaces appear to a protocol as distinct physical interfaces.

Subinterfaces also allow multiple encapsulations for a protocol on a single interface. For example, you can configure IEEE 802.1Q encapsulation to associate a subinterface with a VLAN.
For more information about configuring subinterfaces, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

SUMMARY STEPS

1. configure terminal
2. interface type number . subint

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>Purpose</strong></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Specifies the VLAN interface to be configured.</td>
</tr>
<tr>
<td>interface type number . subint</td>
<td>The CLI places you into a subinterface configuration mode for the specified VLAN interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config)# interface ethernet 2/2.1</td>
<td></td>
</tr>
<tr>
<td>switch(config-subif)#</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>The CLI prompt changes to indicate that you are in subinterface configuration mode.</td>
</tr>
</tbody>
</table>

Saving and Restoring a Command Mode

The Cisco NX-OS software allows you to save the current command mode, configure a feature, and then restore the previous command mode. The push command saves the command mode, and the pop command restores the command mode.

The following example shows how to save and restore a command mode:

```
switch# configure terminal
switch(config)# event manager applet test
switch(config-applet)# push
switch(config-applet)# configure terminal
switch(config)# username testuser password newtest
switch(config)# pop
switch(config-applet)#
```

Exiting a Configuration Command Mode

SUMMARY STEPS

1. exit
2. end
3. (Optional) Ctrl-Z
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Exits from the current configuration command mode and returns to the previous configuration command mode.</td>
</tr>
<tr>
<td>exit</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config-if)# exit switch(config)#</td>
</tr>
</tbody>
</table>

| **Step 2** | Exits from the current configuration command mode and returns to EXEC mode. |
| end |  |
| Example: | switch(config-if)# end switch# |

| **Step 3** | Exits the current configuration command mode and returns to EXEC mode. |
| (Optional) Ctrl-Z |  |
| Example: | switch(config-if)# ^Z switch# |

**Caution** If you press Ctrl-Z at the end of a command line in which a valid command has been typed, the CLI adds the command to the running configuration file. In most cases, you should exit a configuration mode using the `exit` or `end` command.

### Command Mode Summary

This table summarizes information about the main command modes.

**Table 4: Command Mode Summary**

<table>
<thead>
<tr>
<th>Mode</th>
<th>Access Method</th>
<th>Prompt</th>
<th>Exit Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXEC</td>
<td>From the login prompt, enter your username and password.</td>
<td>switch#</td>
<td>To exit to the login prompt, use the exit command.</td>
</tr>
<tr>
<td>Global configuration</td>
<td>From EXEC mode, use the configure terminal command.</td>
<td>switch(config)#</td>
<td>To exit to EXEC mode, use the end or exit command or press Ctrl-Z.</td>
</tr>
<tr>
<td>Interface configuration</td>
<td>From global configuration mode, specify an interface with an interface command.</td>
<td>switch(config-if)#</td>
<td>To exit to global configuration mode, use the exit command. To exit to EXEC mode, use the exit command or press Ctrl-Z.</td>
</tr>
<tr>
<td>Subinterface configuration</td>
<td>From global configuration mode, specify a subinterface with an interface command.</td>
<td>switch(config-subif)#</td>
<td>To exit to global configuration mode, use the exit command. To exit to EXEC mode, use the end command or press Ctrl-Z.</td>
</tr>
</tbody>
</table>
### Special Characters

This table lists the characters that have special meaning in Cisco NX-OS text strings and should be used only in regular expressions or other special contexts.

**Table 5: Special Characters**

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%</td>
<td>Percent</td>
</tr>
<tr>
<td>#</td>
<td>Pound, hash, or number</td>
</tr>
<tr>
<td>...</td>
<td>Ellipsis</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Less than or greater than</td>
</tr>
<tr>
<td>[ ]</td>
<td>Brackets</td>
</tr>
<tr>
<td>{ }</td>
<td>Braces</td>
</tr>
</tbody>
</table>

### Keystroke Shortcuts

This table lists command key combinations that can be used in both EXEC and configuration modes.

**Table 6: Keystroke Shortcuts**

<table>
<thead>
<tr>
<th>Keystrokes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl-A</td>
<td>Moves the cursor to the beginning of the line.</td>
</tr>
<tr>
<td>Ctrl-B</td>
<td>Moves the cursor one character to the left. When you enter a command that extends beyond a single line, you can press the Left Arrow or Ctrl-B keys repeatedly to scroll back toward the system prompt and verify the beginning of the command entry, or you can press the Ctrl-A key combination.</td>
</tr>
<tr>
<td>Keystrokes</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Ctrl-C</td>
<td>Cancels the command and returns to the command prompt.</td>
</tr>
<tr>
<td>Ctrl-D</td>
<td>Deletes the character at the cursor.</td>
</tr>
<tr>
<td>Ctrl-E</td>
<td>Moves the cursor to the end of the line.</td>
</tr>
<tr>
<td>Ctrl-F</td>
<td>Moves the cursor one character to the right.</td>
</tr>
<tr>
<td>Ctrl-G</td>
<td>Exits to the previous command mode without removing the command string.</td>
</tr>
<tr>
<td>Ctrl-K</td>
<td>Deletes all characters from the cursor to the end of the command line.</td>
</tr>
<tr>
<td>Ctrl-L</td>
<td>Redisplays the current command line.</td>
</tr>
<tr>
<td>Ctrl-N</td>
<td>Displays the next command in the command history.</td>
</tr>
<tr>
<td>Ctrl-O</td>
<td>Clears the terminal screen.</td>
</tr>
<tr>
<td>Ctrl-P</td>
<td>Displays the previous command in the command history.</td>
</tr>
<tr>
<td>Ctrl-R</td>
<td>Redisplays the current command line.</td>
</tr>
<tr>
<td>Ctrl-T</td>
<td>Transposes the character under the cursor with the character located to the right of the cursor. The cursor is then moved to the right one character.</td>
</tr>
<tr>
<td>Ctrl-U</td>
<td>Deletes all characters from the cursor to the beginning of the command line.</td>
</tr>
<tr>
<td>Ctrl-V</td>
<td>Removes any special meaning for the following keystroke. For example, press Ctrl-V before entering a question mark (?) in a regular expression.</td>
</tr>
<tr>
<td>Ctrl-W</td>
<td>Deletes the word to the left of the cursor.</td>
</tr>
<tr>
<td>Ctrl-X, H</td>
<td>Lists the history of commands you have entered.</td>
</tr>
<tr>
<td></td>
<td>When using this key combination, press and release the Ctrl and X keys together before pressing H.</td>
</tr>
<tr>
<td>Ctrl-Y</td>
<td>Recalls the most recent entry in the buffer (press keys simultaneously).</td>
</tr>
<tr>
<td>Ctrl-Z</td>
<td>Ends a configuration session, and returns you to EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>When used at the end of a command line in which a valid command has been typed, the resulting configuration is first added to the running configuration file.</td>
</tr>
<tr>
<td>Up arrow key</td>
<td>Displays the previous command in the command history.</td>
</tr>
<tr>
<td>Down arrow key</td>
<td>Displays the next command in the command history.</td>
</tr>
<tr>
<td>Right arrow key</td>
<td>Moves your cursor through the command string, either forward or backward, allowing you to edit the current command.</td>
</tr>
<tr>
<td>Left arrow key</td>
<td></td>
</tr>
<tr>
<td>?</td>
<td>Displays a list of available commands.</td>
</tr>
<tr>
<td>Keystrokes</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Tab        | Completes the word for you after you enter the first characters of the word and then press the Tab key. All options that match are presented. Use tabs to complete the following items:  
  - Command names  
  - Scheme names in the file system  
  - Server names in the file system  
  - Filenames in the file system  
Example:  
  switch(config)# xm<Tab>  
  switch(config)# xml<Tab>  
  switch(config)# xml server  
Example:  
  switch(config)# c<Tab>  
  callhome class-map clock  
  cdp cli control-plane  
  switch(config)# cl<Tab>  
  class-map cli clock  
  switch(config)# cl<Tab>  
  class-map  
Example:  
  switch# cd bootflash:<Tab>  
  bootflash://  
  bootflash://sup-1/  
  bootflash://sup-active/  
  bootflash://sup-local/  
  bootflash://module-27/  
  bootflash://module-28/  
Example:  
  switch# cd bootflash://mo<Tab>  
  bootflash://module-27/ bootflash://module-28/  
  switch# cd bootflash://module-28/  
  switch# cd bootflash://module-28/  
  bootflash://module-28/  
**Note**  
You cannot access remote machines using the cd command. If you are on slot 27 and enter the cd bootflash://module-28 command, the following message appears: "Changing directory to a non-local server is not allowed."

## Abbreviating Commands

You can abbreviate commands and keywords by entering the first few characters of a command. The abbreviation must include sufficient characters to make it unique from other commands or keywords. If you
are having trouble entering a command, check the system prompt and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

This table lists examples of command abbreviations.

**Table 7: Examples of Command Abbreviations**

<table>
<thead>
<tr>
<th>Command</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>configure terminal</td>
<td>conf t</td>
</tr>
<tr>
<td>copy running-config startup-config</td>
<td>copy run start</td>
</tr>
<tr>
<td>interface ethernet 1/2</td>
<td>int e 1/2</td>
</tr>
<tr>
<td>show running-config</td>
<td>sh run</td>
</tr>
</tbody>
</table>

**Completing a Partial Command Name**

If you cannot remember a complete command name or if you want to reduce the amount of typing you have to perform, enter the first few letters of the command, and then press the Tab key. The command line parser will complete the command if the string entered is unique to the command mode. If your keyboard does not have a Tab key, press Ctrl-I instead.

The CLI recognizes a command once you have entered enough characters to make the command unique. For example, if you enter `conf` in EXEC mode, the CLI will be able to associate your entry with the `configure` command, because only the `configure` command begins with `conf`.

In the following example, the CLI recognizes the unique string for `conf` in EXEC mode when you press the Tab key:

```
switch# conf<Tab>
switch# configure
```

When you use the command completion feature, the CLI displays the full command name. The CLI does not execute the command until you press the Return or Enter key. This feature allows you to modify the command if the full command was not what you intended by the abbreviation. If you enter a set of characters that could indicate more than one command, a list of matching commands displays.

For example, entering `co<Tab>` lists all commands available in EXEC mode beginning with `co`:

```
switch# co<Tab>
configure  copy
switch# co
```

Note that the characters you entered appear at the prompt again to allow you to complete the command entry.

**Identifying Your Location in the Command Hierarchy**

Some features have a configuration submode hierarchy nested more than one level. In these cases, you can display information about your present working context (PWC).
### SUMMARY STEPS

1. `where detail`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1: <code>where detail</code></td>
<td>Displays the PWC.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>switch# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><code>switch(config)# interface mgmt0</code></td>
<td></td>
</tr>
<tr>
<td><code>switch(config-if)# where detail</code></td>
<td></td>
</tr>
<tr>
<td>mode: conf</td>
<td></td>
</tr>
<tr>
<td>interface mgmt0</td>
<td></td>
</tr>
<tr>
<td>username: admin</td>
<td></td>
</tr>
<tr>
<td>routing-context vrf: default</td>
<td></td>
</tr>
</tbody>
</table>

### Using the no Form of a Command

Almost every configuration command has a no form that can be used to disable a feature, revert to a default value, or remove a configuration.

This example shows how to disable a feature:

```
switch# configure terminal
switch(config)# feature tacacs+
switch(config)# no feature tacacs+
```

This example shows how to revert to the default value for a feature:

```
switch# configure terminal
switch(config)# banner motd #Welcome to the switch#
switch(config)# show banner motd
Welcome to the switch
switch(config)# no banner motd
switch(config)# show banner motd
User Access Verification
```

This example shows how to remove the configuration for a feature:

```
switch# configure terminal
switch(config)# radius-server host 10.10.2.2
switch(config)# show radius-server
retransmission count:0
timeout value:1
deadtime value:1
total number of servers:1

following RADIUS servers are configured:
    10.10.1.1:
        available for authentication on port:1812
        available for accounting on port:1813
    10.10.2.2:
```

---

available for authentication on port:1812
available for accounting on port:1813

switch(config)# no radius-server host 10.10.2.2
switch(config)# show radius-server
retransmission count:0
timeout value:1
deadtime value:1
total number of servers:1

following RADIUS servers are configured:
    10.10.1.1:
        available for authentication on port:1812
        available for accounting on port:1813

This example shows how to use the no form of a command in EXEC mode:

switch# cli var name testinterface ethernet1/2
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2013-05-12-13.43.13"
testinterface="ethernet1/2"

switch# cli no var name testinterface
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2013-05-12-13.43.13"

Configuring CLI Variables

This section describes CLI variables in the Cisco NX-OS CLI.

About CLI Variables

The Cisco NX-OS software supports the definition and use of variables in CLI commands.

You can refer to CLI variables in the following ways:

• Entered directly on the command line.
• Passed to a script initiated using the run-script command. The variables defined in the parent shell are available for use in the child run-script command process.

CLI variables have the following characteristics:

• Cannot have nested references through another variable
• Can persist across switch reloads or exist only for the current session

Cisco NX-OS supports one predefined variable: TIMESTAMP. This variable refers to the current time when the command executes in the format YYYY-MM-DD-HH.MM.SS.

Note

The TIMESTAMP variable name is case sensitive. All letters must be uppercase.
Configuring CLI Session-Only Variables

You can define CLI session variables to persist only for the duration of your CLI session. These variables are useful for scripts that you execute periodically. You can reference the variable by enclosing the name in parentheses and preceding it with a dollar sign ($), for example $(variable-name).

SUMMARY STEPS

1. cli var name variable-name variable-text
2. (Optional) show cli variables

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>cli var name variable-name variable-text</td>
<td>Configures the CLI session variable. The variable-name argument is alphanumeric, case sensitive, and has a maximum length of 31 characters. The variable-text argument is alphanumeric, case sensitive, can contain spaces, and has a maximum length of 200 characters.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# cli var name testinterface ethernet 2/1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td>(Optional) show cli variables</td>
<td>Displays the CLI variable configuration.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# show cli variables</td>
<td></td>
</tr>
</tbody>
</table>

Configuring Persistent CLI Variables

You can configure CLI variables that persist across CLI sessions and device reloads.

SUMMARY STEPS

1. configure terminal
2. cli var name variable-name variable-text
3. exit
4. (Optional) show cli variables
5. (Optional) copy running-config startup-config

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
</tbody>
</table>
| Step 2 | cli var name variable-name variable-text  
Example:  
switch(config)# cli var name testinterface ethernet 2/1 | Configures the CLI persistent variable. The variable name is a case-sensitive, alphanumeric string and must begin with an alphabetic character. The maximum length is 31 characters.  
Note: Beginning with Cisco NX-OS Release 7.0(3)I4(1), variables can include hyphens (-) and underscores (_). |
|---|---|
| Step 3 | exit  
Example:  
switch(config)# exit  
switch# | Exits global configuration mode. |
| Step 4 | (Optional) show cli variables  
Example:  
switch# show cli variables | Displays the CLI variable configuration. |
| Step 5 | (Optional) copy running-config startup-config  
Example:  
switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

**Command Aliases**

This section provides information about command aliases.

**About Command Aliases**

You can define command aliases to replace frequently used commands. The command aliases can represent all or part of the command syntax.

Command alias support has the following characteristics:

- Command aliases are global for all user sessions.
- Command aliases persist across reboots if you save them to the startup configuration.
- Command alias translation always takes precedence over any keyword in any configuration mode or submode.
- Command alias configuration takes effect for other user sessions immediately.
- The Cisco NX-OS software provides one default alias, alias, which is the equivalent to the show cli alias command that displays all user-defined aliases.
- You cannot delete or change the default command alias alias.
- You can nest aliases to a maximum depth of 1. One command alias can refer to another command alias that must refer to a valid command, not to another command alias.
• A command alias always replaces the first command keyword on the command line.
• You can define command aliases for commands in any command mode.
• If you reference a CLI variable in a command alias, the current value of the variable appears in the alias, not the variable reference.
• You can use command aliases for show command searching and filtering.

Defining Command Aliases

You can define command aliases for commonly used commands.

SUMMARY STEPS

1. configure terminal
2. cli alias name alias-name alias-text
3. exit
4. (Optional) alias
5. (Optional) copy running-config startup-config

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> cli alias name alias-name alias-text</td>
<td>Configures the command alias. The alias name is an alphanumeric string that is not case sensitive and must begin with an alphabetic character. The maximum length is 30 characters.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config)# cli alias name ethint interface ethernet</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> exit</td>
<td>Exits global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config)# exit</td>
<td></td>
</tr>
<tr>
<td>switch#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> (Optional) alias</td>
<td>Displays the command alias configuration.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# alias</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> (Optional) copy running-config startup-config</td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# copy running-config startup-config</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Command Aliases for a User Session

You can create a command alias for the current user session that is not available to any other user on the Cisco NX-OS device. You can also save the command alias for future use by the current user account.

SUMMARY STEPS

1. terminal alias [persist] alias-name command-string

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>terminal alias [persist] alias-name command-string</td>
</tr>
<tr>
<td></td>
<td>Example: switch# terminal alias shintbr show interface brief</td>
</tr>
</tbody>
</table>

Command Scripts

This section describes how you can create scripts of commands to perform multiple tasks.

Running a Command Script

You can create a list of commands in a file and execute them from the CLI. You can use CLI variables in the command script.

Note: You cannot create the script files at the CLI prompt. You can create the script file on a remote device and copy it to the bootflash: or volatile: directory on the Cisco NX-OS device.

SUMMARY STEPS

1. run-script [bootflash: | volatile:] filename

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>run-script [bootflash:</td>
</tr>
<tr>
<td></td>
<td>Example: switch# run-script testfile</td>
</tr>
</tbody>
</table>
Echoing Information to the Terminal

You can echo information to the terminal, which is particularly useful from a command script. You can reference CLI variables and use formatting options in the echoed text.

This table lists the formatting options that you can insert in the text.

<table>
<thead>
<tr>
<th>Formatting Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\b</td>
<td>Inserts back spaces.</td>
</tr>
<tr>
<td>\c</td>
<td>Removes the new line character at the end of the text string.</td>
</tr>
<tr>
<td>\f</td>
<td>Inserts a form feed character.</td>
</tr>
<tr>
<td>\n</td>
<td>Inserts a new line character.</td>
</tr>
<tr>
<td>\r</td>
<td>Returns to the beginning of the text line</td>
</tr>
<tr>
<td>\t</td>
<td>Inserts a horizontal tab character.</td>
</tr>
<tr>
<td>\v</td>
<td>Inserts a vertical tab character.</td>
</tr>
<tr>
<td>\</td>
<td>Displays a backslash character.</td>
</tr>
<tr>
<td>\nnn</td>
<td>Displays the corresponding ASCII octal character.</td>
</tr>
</tbody>
</table>

**SUMMARY STEPS**

1. echo [backslash-interpret] [text]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 echo [backslash-interpret] [text]</td>
<td>The backslash-interpret keyword indicates that the text string contains formatting options. The text argument is alphanumeric, case sensitive, and can contain blanks. The maximum length is 200 characters. The default is a blank line.</td>
</tr>
</tbody>
</table>

**Delaying Command Action**

You can delay a command action for a period of time, which is particularly useful within a command script.

**SUMMARY STEPS**

1. sleep seconds
**Detailed Steps**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>sleep seconds</td>
<td>Causes a delay for a number of seconds. The range is from 0 to 2147483647.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>switch# sleep 30</code></td>
<td></td>
</tr>
</tbody>
</table>

---

### Context-Sensitive Help

The Cisco NX-OS software provides context-sensitive help in the CLI. You can use a question mark (?) at any point in a command to list the valid input options.

CLI uses the caret (^) symbol to isolate input errors. The ^ symbol appears at the point in the command string where you have entered an incorrect command, keyword, or argument.

This table shows example outputs of context sensitive help.

**Table 9: Context-Sensitive Help Example**

<table>
<thead>
<tr>
<th>Example Outputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>switch# clock ?</code></td>
<td>Displays the command syntax for the clock command in EXEC mode.</td>
</tr>
<tr>
<td><code>set  HH:MM:SS Current Time</code></td>
<td>The switch output shows that the <code>set</code> keyword is required for using the clock command.</td>
</tr>
<tr>
<td><code>switch# clock</code></td>
<td></td>
</tr>
<tr>
<td><code>switch# clock set ?</code></td>
<td>Displays the command syntax for setting the time.</td>
</tr>
<tr>
<td><code>WORD  HH:MM:SS Current Time</code></td>
<td>The help output shows that the current time is required for setting the clock and how to format the time.</td>
</tr>
<tr>
<td><code>switch# clock set</code></td>
<td></td>
</tr>
<tr>
<td><code>switch# clock set 13:32:00&lt;CR&gt;</code></td>
<td>Adds the current time.</td>
</tr>
<tr>
<td><code>% Incomplete command</code></td>
<td>The CLI indicates the command is incomplete.</td>
</tr>
<tr>
<td><code>switch#</code></td>
<td></td>
</tr>
<tr>
<td><code>switch# &lt;Ctrl-P&gt;</code></td>
<td>Displays the previous command that you entered.</td>
</tr>
<tr>
<td><code>switch# clock set 13:32:00</code></td>
<td></td>
</tr>
<tr>
<td><code>switch# clock set 13:32:00 ?</code></td>
<td>Displays the additional arguments for the clock set command.</td>
</tr>
<tr>
<td><code>&lt;1-31&gt;</code></td>
<td>Day of the month</td>
</tr>
<tr>
<td><code>switch# clock set 13:32:00</code></td>
<td></td>
</tr>
</tbody>
</table>

---

*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide, Release 9.3(x)*
### Example Outputs

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch# clock set 13:32:00 18 ?</td>
<td>Displays the additional arguments for the clock set command.</td>
</tr>
<tr>
<td>April</td>
<td>Month of the year</td>
</tr>
<tr>
<td>August</td>
<td>Month of the year</td>
</tr>
<tr>
<td>December</td>
<td>Month of the year</td>
</tr>
<tr>
<td>February</td>
<td>Month of the year</td>
</tr>
<tr>
<td>January</td>
<td>Month of the year</td>
</tr>
<tr>
<td>July</td>
<td>Month of the year</td>
</tr>
<tr>
<td>June</td>
<td>Month of the year</td>
</tr>
<tr>
<td>March</td>
<td>Month of the year</td>
</tr>
<tr>
<td>May</td>
<td>Month of the year</td>
</tr>
<tr>
<td>November</td>
<td>Month of the year</td>
</tr>
<tr>
<td>October</td>
<td>Month of the year</td>
</tr>
<tr>
<td>September</td>
<td>Month of the year</td>
</tr>
<tr>
<td>switch# clock set 13:32:00 18 April 13&lt;CR&gt;</td>
<td>Adds the date to the clock setting.</td>
</tr>
<tr>
<td>% Invalid input detected at '^' marker.</td>
<td>The CLI indicates an error with the caret symbol (^) at 13.</td>
</tr>
<tr>
<td>switch# clock set 13:32:00 18 April ?</td>
<td>Displays the correct arguments for the year.</td>
</tr>
<tr>
<td>&lt;2000-2030&gt; Enter the year (no abbreviation)</td>
<td></td>
</tr>
<tr>
<td>switch# clock set 13:32:00 18 April</td>
<td>Enters the correct syntax for the clock set command.</td>
</tr>
<tr>
<td>switch# clock set 13:32:00 18 April 2013&lt;CR&gt;</td>
<td></td>
</tr>
<tr>
<td>switch#</td>
<td></td>
</tr>
</tbody>
</table>

### Understanding Regular Expressions

The Cisco NX-OS software supports regular expressions for searching and filtering in CLI output, such as the show commands. Regular expressions are case sensitive and allow for complex matching requirements.

### Special Characters

You can also use other keyboard characters (such as ! or ~) as single-character patterns, but certain keyboard characters have special meanings when used in regular expressions.

This table lists the keyboard characters that have special meanings.

#### Table 10: Special Characters with Special Meaning

<table>
<thead>
<tr>
<th>Character</th>
<th>Special Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>Matches any single character, including white space.</td>
</tr>
<tr>
<td>*</td>
<td>Matches 0 or more sequences of the pattern.</td>
</tr>
<tr>
<td>+</td>
<td>Matches 1 or more sequences of the pattern.</td>
</tr>
<tr>
<td>?</td>
<td>Matches 0 or 1 occurrences of the pattern.</td>
</tr>
</tbody>
</table>
To use these special characters as single-character patterns, remove the special meaning by preceding each character with a backslash (\). This example contains single-character patterns that match a dollar sign ($), an underscore (_), and a plus sign (+), respectively:

\$ \_ \+

### Multiple-Character Patterns

You can also specify a pattern that contains multiple characters by joining letters, digits, or keyboard characters that do not have special meanings. For example, a4% is a multiple-character regular expression.

With multiple-character patterns, the order is important. The regular expression `a4%` matches the character a followed by a 4 followed by a percent sign (%). If the string does not have a4%, in that order, pattern matching fails. The multiple-character regular expression `a.` (the character a followed by a period) uses the special meaning of the period character to match the letter a followed by any single character. With this example, the strings ab, a!, or a2 are all valid matches for the regular expression.

You can remove the special meaning of a special character by inserting a backslash before it. For example, when the expression `a\.` is used in the command syntax, only the string `a.` will be matched.

### Anchoring

You can match a regular expression pattern against the beginning or the end of the string by anchoring these regular expressions to a portion of the string using the special characters.

This table lists the special characters that you can use for anchoring.

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>^</td>
<td>Matches the beginning of the string.</td>
</tr>
<tr>
<td>$</td>
<td>Matches the end of the string.</td>
</tr>
</tbody>
</table>

For example, the regular expression `^con` matches any string that starts with con, and `sole$` matches any string that ends with sole.
The ^ symbol can also be used to indicate the logical function "not" when used in a bracketed range. For example, the expression [^abcd] indicates a range that matches any single letter, as long as it is not a, b, c, or d.

### Searching and Filtering show Command Output

Often, the output from show commands can be lengthy and cumbersome. The Cisco NX-OS software provides the means to search and filter the output so that you can easily locate information. The searching and filtering options follow a pipe character ( | ) at the end of the show command. You can display the options using the CLI context-sensitive help facility:

```
switch# show running-config | ?

- cut: Print selected parts of lines.
- diff: Show difference between current and previous invocation (creates temp files: remove them with 'diff-clean' command and don't use it on commands with big outputs, like 'show tech'!)
- egrep: Egrep - print lines matching a pattern
- grep: Grep - print lines matching a pattern
- head: Display first lines
- human: Output in human format
- last: Display last lines
- less: Filter for paging
- no-more: Turn-off pagination for command output
- perl: Use perl script to filter output
- section: Show lines that include the pattern as well as the subsequent lines that are more indented than matching line
- sed: Stream Editor
- sort: Stream Sorter
- sscp: Stream SCP (secure copy)
- tr: Translate, squeeze, and/or delete characters
- uniq: Discard all but one of successive identical lines
- vsh: The shell that understands cli command
- wc: Count words, lines, characters
- xml: Output in xml format (according to .xsd definitions)
- begin: Begin with the line that matches
- count: Count number of lines
- end: End with the line that matches
- exclude: Exclude lines that match
- include: Include lines that match
```

### Filtering and Searching Keywords

The Cisco NX-OS CLI provides a set of keywords that you can use with the show commands to search and filter the command output.

---

Note: The ^ symbol can also be used to indicate the logical function "not" when used in a bracketed range. For example, the expression [^abcd] indicates a range that matches any single letter, as long as it is not a, b, c, or d.
This table lists the keywords for filtering and searching the CLI output.

**Table 12: Filtering and Searching Keywords**

<table>
<thead>
<tr>
<th>Keyword Syntax</th>
<th>Description</th>
</tr>
</thead>
</table>
| `begin string` | Starts displaying at the line that contains the text that matches the search string. The search string is case sensitive. Example:  
  `show version | begin Hardware` |
| `count` | Displays the number of lines in the command output. Example:  
  `show running-config | count` |
| `cut [-d character] {-b | -c | -f | -s}` | Displays only part of the output lines. You can display a number of bytes (-b), characters (-c), cut [-d character] {-b | -c | -f | -s}, or fields (-f). You can also use the -d keyword to define a field delimiter other than the tag character default. The -s keyword suppresses the display of the line that does not contain the delimiter. Example:  
  `show file testoutput | cut -b 1-10` |
| `end string` | Displays all lines up to the last occurrence of the search string. Example:  
  `show running-config | end interface` |
| `exclude string` | Displays all lines that do not include the search string. The search string is case sensitive. Example:  
  `show interface brief | exclude down` |
| `head [lines lines]` | Displays the beginning of the output for the number of lines specified. The default number of lines is 10. Example:  
  `show logging logfile | head lines 50` |
| `human` | Displays the output in normal format if you have previously set the output format to XML using the terminal output xml command. Example:  
  `show version | human` |
| `include string` | Displays all lines that include the search string. The search string is case sensitive. Example:  
  `show interface brief | include up` |
### diff Utility

You can compare the output from a `show` command with the output from the previous invocation of that command.

```
diff-clean [all-sessions] [all-users]
```

This table describes the keywords for the `diff` utility.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all-sessions</td>
<td>Removes diff temporary files from all sessions (past and present sessions) of the current user.</td>
</tr>
<tr>
<td>all-users</td>
<td>Removes diff temporary files from all sessions (past and present sessions) of all users.</td>
</tr>
</tbody>
</table>

The Cisco NX-OS software creates temporary files for the most current output for a `show` command for all current and previous users sessions. You can remove these temporary files using the `diff-clean` command.

```
diff-clean [all-sessions | all-users]
```
By default, the `diff-clean` command removes the temporary files for the current user's active session. The `all-sessions` keyword removes temporary files for all past and present sessions for the current user. The `all-users` keyword removes temporary files for all past and present sessions for the all users.

**grep and egrep Utilities**

You can use the Global Regular Expression Print (grep) and Extended grep (egrep) command-line utilities to filter the show command output.

The grep and egrep syntax is as follows:

```
{ grep | egrep } [count] [ignore-case] [invert-match] [line-exp] [line-number] [next lines] [prev lines] [word-exp] [expression ]
```

This table lists the grep and egrep parameters.

**Table 13: grep and egrep Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Displays only the total count of matched lines.</td>
</tr>
<tr>
<td>ignore-case</td>
<td>Specifies to ignore the case difference in matched lines.</td>
</tr>
<tr>
<td>invert-match</td>
<td>Displays lines that do not match the expression.</td>
</tr>
<tr>
<td>line-exp</td>
<td>Displays only lines that match a complete line.</td>
</tr>
<tr>
<td>line-number</td>
<td>Specifies to display the line number before each matched line.</td>
</tr>
<tr>
<td>next lines</td>
<td>Specifies the number of lines to display after a matched line. The default is 0. The range is from 1 to 999.</td>
</tr>
<tr>
<td>prev lines</td>
<td>Specifies the number of lines to display before a matched line. The default is 0. The range is from 1 to 999.</td>
</tr>
<tr>
<td>word-exp</td>
<td>Specifies only lines that match a complete word.</td>
</tr>
<tr>
<td>expression</td>
<td>Specifies a regular expression for searching the output.</td>
</tr>
</tbody>
</table>

**less Utility**

You can use the less utility to display the contents of the show command output one screen at a time. You can enter less commands at the `:` prompt. To display all less commands you can use, enter `h` at the `:` prompt.

**Mini AWK Utility**

AWK is a simple but powerful utility to summarize text output. You can use this utility after a pipe (`|`) to further process the text output of a command. Cisco NX-OS supports a mini AWK, which takes an inline program as an argument.

This example shows how the mini AWK utility can be used to summarize the text output of the `show ip route summary vrf all` command:
switch# show ip route summary vrf all | grep "Total number of routes"
Total number of routes: 3
Total number of routes: 10

switch# show ip route summary vrf all | grep "Total number of routes" | awk '{ x = x + $5} END { print x }'
13

**sed Utility**

You can use the Stream Editor (sed) utility to filter and manipulate the show command output as follows:

sed command

The command argument contains sed utility commands.

**sort Utility**

You can use the sort utility to filter show command output.

The sort utility syntax is as follows:


This table describes the sort utility parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-M</td>
<td>Sorts by month.</td>
</tr>
<tr>
<td>-b</td>
<td>Ignores leading blanks (space characters). The default sort includes the leading blanks.</td>
</tr>
<tr>
<td>-d</td>
<td>Sorts by comparing only blanks and alphanumeric characters. The default sort includes all characters.</td>
</tr>
<tr>
<td>-f</td>
<td>Folds lowercase characters into uppercase characters.</td>
</tr>
<tr>
<td>-g</td>
<td>Sorts by comparing a general numeric value.</td>
</tr>
<tr>
<td>-i</td>
<td>Sorts only using printable characters. The default sort includes nonprintable characters.</td>
</tr>
<tr>
<td>-k field-number[, char-position][ordering]</td>
<td>Sorts according to a key value. There is no default key value.</td>
</tr>
<tr>
<td>-n</td>
<td>Sorts according to a numeric string value.</td>
</tr>
<tr>
<td>-r</td>
<td>Reverses order of the sort results. The default sort output is in ascending order.</td>
</tr>
<tr>
<td>-t delimiter</td>
<td>Sorts using a specified delimiter. The default delimiter is the space character.</td>
</tr>
</tbody>
</table>
### Searching and Filtering from the `--More--` Prompt

You can search and filter output from `--More--` prompts in the `show` command output.

This table describes the `--More--` prompt commands.

**Table 15: `--More--` Prompt Commands**

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[lines]&lt;space&gt;</code></td>
<td>Displays output lines for either the specified number of lines or the current screen size.</td>
</tr>
<tr>
<td><code>[lines]z</code></td>
<td>Displays output lines for either the specified number of lines or the current screen size. If you use the <code>lines</code> argument, that value becomes the new default screen size.</td>
</tr>
<tr>
<td><code>[lines]&lt;return&gt;</code></td>
<td>Displays output lines for either the specified number of lines or the current default number of lines. The initial default is 1 line. If you use the optional <code>lines</code> argument, that value becomes the new default number of lines to display for this command.</td>
</tr>
<tr>
<td><code>[lines]d</code> or <code>[lines]Ctrl+shift+D</code></td>
<td>Scrolls through output lines for either the specified number of lines or the current default number of lines. The initial default is 11 lines. If you use the optional <code>lines</code> argument, that value becomes the new default number of lines to display for this command.</td>
</tr>
<tr>
<td><code>q</code> or <code>Q</code> or Ctrl-C</td>
<td>Exits the <code>--More--</code> prompt.</td>
</tr>
<tr>
<td><code>[lines]s</code></td>
<td>Skips forward in the output for either the specified number of lines or the current default number of lines and displays a screen of lines. The default is 1 line.</td>
</tr>
<tr>
<td><code>[lines]f</code></td>
<td>Skips forward in the output for either the specified number of screens or the current default number of screens and displays a screen of lines. The default is 1 screen.</td>
</tr>
<tr>
<td><code>=</code></td>
<td>Displays the current line number.</td>
</tr>
<tr>
<td><code>[count]/expression</code></td>
<td>Skips to the line that matches the regular expression and displays a screen of output lines. Use the optional <code>count</code> argument to search for lines with multiple occurrences of the expression. This command sets the current regular expression that you can use in other commands.</td>
</tr>
<tr>
<td><code>[count]n</code></td>
<td>Skips to the next line that matches the current regular expression and displays a screen of output lines. Use the optional <code>count</code> argument to skip past matches.</td>
</tr>
<tr>
<td>`{!</td>
<td>:[shell-cmd]}`</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-u</code></td>
<td>Removes duplicate lines from the sort results. The sort output displays the duplicate lines.</td>
</tr>
</tbody>
</table>
Using the Command History

The Cisco NX-OS software CLI allows you to access the command history for the current user session. You can recall and reissue commands, with or without modification. You can also clear the command history.

Recalling a Command

You can recall a command in the command history to optionally modify and enter again.

This example shows how to recall a command and reenter it:

```
switch(config)# show cli history
0 11:04:07 configure terminal
1 11:04:28 show interface ethernet 2/24
2 11:04:39 interface ethernet 2/24
3 11:05:13 no shutdown
4 11:05:19 exit
5 11:05:25 show cli history
switch(config)# !1
switch(config)# show interface ethernet 2/24
```

You can also use the Ctrl-P and Ctrl-N keystroke shortcuts to recall commands.

Controlling CLI History Recall

You can control the commands that you recall from the CLI history using the Ctrl-P and Ctrl-N keystroke shortcuts. Cisco NX-OS software recalls all commands from the current command mode and higher command modes. For example, if you are working in global configuration mode, the command recall keystroke shortcuts recall both EXEC mode and global configuration mode commands.

Configuring the CLI Edit Mode

You can recall commands from the CLI history using the Ctrl-P and Ctrl-N keystroke shortcuts and edit them before reissuing them. The default edit mode is emacs. You can change the edit mode to vi.

SUMMARY STEPS

1. [no] terminal edit-mode vi [persist]

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>Repeats the previous command.</td>
</tr>
</tbody>
</table>
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>[no] terminal edit-mode vi [persist]</td>
<td>Changes the CLI edit mode to vi for the user session. The persist keyword makes the setting persistent across sessions for the current username. Use the no to revert to using emacs.</td>
</tr>
</tbody>
</table>

### Displaying the Command History

You can display the command history using the show cli history command.

The show cli history command has the following syntax:

`show cli history [lines] [config-mode | exec-mode | this-mode-only] [unformatted]`

By default, the number of lines displayed is 12 and the output includes the command number and timestamp.

This example shows how to display the default number of lines of the command history:

```
switch# show cli history
```

This example shows how to display 20 lines of the command history:

```
switch# show cli history 20
```

This example shows how to display only the configuration commands in the command history:

```
switch(config)# show cli history config-mode
```

This example shows how to display only the EXEC commands in the command history:

```
switch(config)# show cli history exec-mode
```

This example shows how to display only the commands in the command history for the current command mode:

```
switch(config-if)# show cli history this-mode-only
```

This example shows how to display only the commands in the command history without the command number and timestamp:

```
switch(config)# show cli history unformatted
```
Enabling or Disabling the CLI Confirmation Prompts

For many features, the Cisco NX-OS software displays prompts on the CLI that ask for confirmation before continuing. You can enable or disable these prompts. The default is enabled.

**SUMMARY STEPS**

1. [no] terminal dont-ask [persist]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 [no] terminal dont-ask [persist]</td>
<td>Disables the CLI confirmation prompt. The persist keyword makes the setting persistent across sessions for the current username. The default is enabled.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Use the no form of the command to enable the CLI confirmation prompts.</td>
</tr>
<tr>
<td>switch# terminal dont-ask</td>
<td></td>
</tr>
</tbody>
</table>

Setting CLI Display Colors

You can change the CLI colors to display as follows:

- The prompt displays in green if the previous command succeeded.
- The prompt displays in red if the previous command failed.
- The user input displays in blue.
- The command output displays in the default color.

The default colors are sent by the terminal emulator software.

**SUMMARY STEPS**

1. terminal color [evening] [persist]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 terminal color [evening] [persist]</td>
<td>Sets the CLI display colors for the terminal session. The evening keyword is not supported. The persist keyword makes the setting persistent across sessions for the current username. The default setting is not persistent.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# terminal color</td>
<td></td>
</tr>
</tbody>
</table>

Sending Commands to Modules

You can send commands directly to modules from the supervisor module session using the slot command.

The slot has the following syntax:
slot slot-number [quoted] command-string

By default, the keyword and arguments in the command-string argument are separated by a space. To send more than one command to a module, separate the commands with a space character, a semicolon character (;), and a space character.

The quoted keyword indicates that the command string begins and ends with double quotation marks ("). Use this keyword when you want to redirect the module command output to a filtering utility, such as diff, that is supported only on the supervisor module session.

This example shows how to display and filter module information:

```
switch# slot 27 show version | grep lc
```

This example shows how to filter module information on the supervisor module session:

```
switch# slot 27 quoted "show version" | diff
```

Sending Command Output in Email

You can use the CLI to send the output of a show command to an email address using the pipe operator (|).

```
Sending Command Output in Email

You can use the CLI to send the output of a show command to an email address using the pipe operator (|).

---

The email configuration remains persistent for all show command output until it is reconfigured.

---

Note

The email configuration remains persistent for all show command output until it is reconfigured.
### SUMMARY STEPS

1. configure terminal
2. email
3. smtp-host ip-address smtp-port
4. vrf management
5. from email-address
6. reply-to email-address
7. exit
8. exit
9. show email
10. show-command | email subject subject email-address

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>configure terminal</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch# configure terminal&lt;br&gt;switch(config)#</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>email</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config)# email&lt;br&gt;switch(config-email)#</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>smtp-host ip-address smtp-port</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-email)# smtp-host 198.51.100.1 smtp-port 25</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>vrf management</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-email)# vrf management</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><strong>from email-address</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-email)# from <a href="mailto:admin@Mycompany.com">admin@Mycompany.com</a></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td><strong>reply-to email-address</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-email)# reply-to <a href="mailto:admin@Mycompany.com">admin@Mycompany.com</a></td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td><strong>exit</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-email)# exit&lt;br&gt;switch(config)#</td>
</tr>
</tbody>
</table>
## Purpose

**Command or Action**

**Purpose**

**Step 8**

- `exit`
- **Example:**
  ```
  switch(config)# exit
  switch#
  ```
  Exits global configuration mode.

**Step 9**

- `show email`
- **Example:**
  ```
  switch# show email
  ```
  Displays the email configuration.

**Step 10**

- `show-command | email subject subject email-address`
- **Example:**
  ```
  switch# show interface brief | email subject show-interface admin@Mycompany.com
  Email sent
  ```
  Uses the pipe operator (`|`) to send the output of the specified show command with a subject to an email address.

---

### BIOS Loader Prompt

When the supervisor modules power up, a specialized BIOS image automatically loads and tries to locate a valid nx-os image for booting the system. If a valid nx-os image is not found, the following BIOS loader prompt displays:

```
loader>
```

For information on how to load the Cisco NX-OS software from the `loader>` prompt, see the Cisco Nexus 9000 Series NX-OS Troubleshooting Guide.

### Examples Using the CLI

This section includes examples of using the CLI.

#### Using the System-Defined Timestamp Variable

This example uses `<$(TIMESTAMP)>` when redirecting `show` command output to a file:

```
switch# show running-config > rcfg.$(TIMESTAMP)
Preparing to copy....done
switch# dir
  12667 May 01 12:27:59 2013  rcfg.2013-05-01-12.27.59

Usage for bootflash://sup-local
8192 bytes used
20963328 bytes free
20971520 bytes total
```
Using CLI Session Variables

You can reference a variable using the syntax $(variable-name).

This example shows how to reference a user-defined CLI session variable:

```
switch# show interface $(testinterface)
```

```
Ethernet2/1 is down (Administratively down)
Hardware is 10/100/1000 Ethernet, address is 0000.0000.0000 (bia 0019.076c.4dac)
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
  reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA
auto-duplex, auto-speed
Beacon is turned off
Auto-Negotiation is turned on
Input flow-control is off, output flow-control is off
Auto-mdix is turned on
Switchport monitor is off
Last clearing of "show interface" counters never
5 minute input rate 0 bytes/sec, 0 packets/sec
5 minute output rate 0 bytes/sec, 0 packets/sec
L3 in Switched:
  ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
L3 out Switched:
  ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
Rx
  0 input packets 0 unicast packets 0 multicast packets
  0 broadcast packets 0 jumbo packets 0 storm suppression packets
  0 bytes
Tx
  0 output packets 0 multicast packets
  0 broadcast packets 0 jumbo packets
  0 bytes
  0 input error 0 short frame 0 watchdog
  0 no buffer 0 runt 0 CRC 0 ecc
  0 overrun 0 underrun 0 ignored 0 bad etype drop
  0 bad proto drop 0 if down drop 0 input with dribble
  0 input discard
  0 output error 0 collision 0 deferred
  0 late collision 0 lost carrier 0 no carrier
  0 babble
  0 Rx pause 0 Tx pause 0 reset
```

Defining Command Aliases

This example shows how to define command aliases:

```
cli alias name ethint interface ethernet
cli alias name shintbr show interface brief
cli alias name shintupbr shintbr | include up | include ethernet
```

This example shows how to use a command alias:

```
switch# configure terminal
switch(config)# ethint 2/3
switch(config-if)#
```
Running a Command Script

This example displays the CLI commands specified in the script file:

```
switch# show file testfile
configure terminal
interface ethernet 2/1
no shutdown
end
show interface ethernet 2/1
```

This example displays the run-script command execution output:

```
switch# run-script testfile
'configure terminal'
'interface ethernet 2/1'
'no shutdown'
'end'
'show interface ethernet 2/1'
```

```
Ethernet2/1 is down (Link not connected)
Hardware is 10/100/1000 Ethernet, address is 0019.076c.4dac (bia 0019.076c.4dac)
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
   reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA
Port mode is trunk
auto-duplex, auto-speed
Beacon is turned off
Auto-Negotiation is turned on
Input flow-control is off, output flow-control is off
Auto-mdix is turned on
Switchport monitor is off
Last clearing of "show interface" counters 1d26.2uh
5 minute input rate 0 bytes/sec, 0 packets/sec
5 minute output rate 0 bytes/sec, 0 packets/sec
Rx
0 input packets 0 unicast packets 0 multicast packets
0 broadcast packets 0 jumbo packets 0 storm suppression packets
0 bytes
Tx
0 output packets 0 multicast packets
0 broadcast packets 0 jumbo packets
0 bytes
```

Sending Command Output in Email

This example shows how to send the output of the show interface brief command to an email address using the pipe operator (|):

```
switch(config)# email
switch(config-email)# smt-host 198.51.100.1 smt-port 25
```
switch(config-email)# vrf management
switch(config-email)# from admin@Mycompany.com
switch(config-email)# reply-to admin@Mycompany.com
switch(config-email)# exit
switch(config)# exit
switch# show email
SMTP host: 198.51.100.1
SMTP port: 25
Reply to: admin@Mycompany.com
From: admin@Mycompany.com
VRF: management
switch# show interface brief | email subject show-interface admin@Mycompany.com

Email sent

The email sent to admin@Mycompany.com with the subject "show-interface" shows the output of the command:

```
<table>
<thead>
<tr>
<th>Ethernet</th>
<th>VLAN Type</th>
<th>Mode</th>
<th>Status</th>
<th>Reason</th>
<th>Speed</th>
<th>Port Ch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eth1/1</td>
<td>--</td>
<td>eth</td>
<td>trunk</td>
<td>up</td>
<td>none</td>
<td>10G (D)</td>
</tr>
<tr>
<td>Eth1/2</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/3</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>up</td>
<td>none</td>
<td>10G (D)</td>
</tr>
<tr>
<td>Eth1/4</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/5</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/6</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/7</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/8</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/9</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
<tr>
<td>Eth1/10</td>
<td>--</td>
<td>eth</td>
<td>routed</td>
<td>down</td>
<td>Link not connected</td>
<td>auto(D)</td>
</tr>
</tbody>
</table>
```

Additional References for the CLI

This section includes additional information related to the CLI.

Related Documents for the CLI

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS Licensing</td>
<td>Cisco NX-OS Licensing Guide</td>
</tr>
</tbody>
</table>
CHAPTER 8

Configuring Terminal Settings and Sessions

This chapter contains the following sections:

- About Terminal Settings and Sessions, on page 93
- Licensing Requirements for Terminal Settings and Sessions, on page 94
- Default Settings for File System Parameters, on page 94
- Configuring the Console Port, on page 94
- Configuring Virtual Terminals, on page 96
- Clearing Terminal Sessions, on page 98
- Displaying Terminal and Session Information, on page 99
- Additional References for Terminal Settings and Sessions, on page 99

About Terminal Settings and Sessions

This section includes information about terminal settings and sessions.

Terminal Session Settings

The Cisco NX-OS software features allow you to manage the following characteristics of terminals:

- **Terminal type**
  - Name used by Telnet when communicating with remote hosts
- **Length**
  - Number of lines of command output displayed before pausing
- **Width**
  - Number of characters displayed before wrapping the line
- **Inactive session timeout**
  - Number of minutes that a session remains inactive before the device terminates it

Console Port

The console port is an asynchronous serial port that allows you to connect to the device for initial configuration through a standard RS-232 port with an RJ-45 connector. Any device connected to this port must be capable of asynchronous transmission. You can configure the following parameters for the console port:

- **Data bits**
  - Specifies the number of bits in an 8-bit byte that is used for data.
Inactivesessiontimeout
Specifiesthenumberofminutesasessioncanbeinactivebeforeitisterminated.
Parity
Specifiestheoddorevenparityforerrordetection.
Speed
Specifiesthestopbitsforanasynchronousline.

Configureyourterminalemulatorwith9600baud,8databits,1stopbit,andnoparity.

Virtual Terminals
Youcanusevirtualterminalstoc connecttoyourdevice.SecureShell(SSH)andTelnetcreatevirtual
terminalsessions.Youcanconfigureinanactivesessiontimeoutandamaximumsessionslimitforvirtual
terminals.

Licensing Requirements for Terminal Settings and Sessions
Thefollowingtableshowsthe licensingrequirementsforthisfeature:

<table>
<thead>
<tr>
<th>Product</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS</td>
<td>Terminal setting configuration requires no license. Any feature not included in a license</td>
</tr>
<tr>
<td></td>
<td>package is bundled with the nx-os image and is provided at no extra charge to you. For a</td>
</tr>
<tr>
<td></td>
<td>complete explanation of the Cisco NX-OS licensing scheme, see the Cisco NX-OS Licensing</td>
</tr>
<tr>
<td></td>
<td>Guide.</td>
</tr>
</tbody>
</table>

Default Settings for File System Parameters
This table lists the default settings for the file system parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default filesystem</td>
<td>bootflash:</td>
</tr>
</tbody>
</table>

Configuring the Console Port
You can set the following characteristics for the console port:
• Data bits
• Inactivesessiontimeout
• Parity
• Speed
• Stop bits

**Before you begin**
Log in to the console port.

### SUMMARY STEPS
1. configure terminal
2. line console
3. databits bits
4. exec-timeout minutes
5. parity {even | none | odd}
6. speed {300 | 1200 | 2400 | 4800 | 9600 | 38400 | 57600 | 115200}
7. stopbits {1 | 2}
8. exit
9. (Optional) show line console
10. (Optional) copy running-config startup-config

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> line console</td>
<td>Enters console configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# line console</td>
<td></td>
</tr>
<tr>
<td>switch(config-console)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> databits bits</td>
<td>Configures the number of data bits per byte. The range is from 5 to 8. The default is 8.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-console)# databits 7</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> exec-timeout minutes</td>
<td>Configures the timeout for an inactive session. The range is from 0 to 525600 minutes (8760 hours). A value of 0 minutes disables the session timeout. The default is 30 minutes.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-console)# exec-timeout 30</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> parity {even</td>
<td>none</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-console)# parity even</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring Virtual Terminals

This section describes how to configure virtual terminals on Cisco NX-OS devices.

### Configuring the Inactive Session Timeout

You can configure a timeout for inactive virtual terminal sessions on the device.

#### SUMMARY STEPS

1. configure terminal
2. line vty
3. exec-timeout minutes
4. exit
5. (Optional) show running-config all | begin vty
6. (Optional) copy running-config startup-config

---

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 6</strong> speed {300</td>
<td>1200</td>
</tr>
<tr>
<td>Example: switch(config-console)# speed 115200</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> stopbits {1</td>
<td>2}</td>
</tr>
<tr>
<td>Example: switch(config-console)# stopbits 2</td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> exit</td>
<td>Exits console configuration mode.</td>
</tr>
<tr>
<td>Example: switch(config-console)# exit switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong> (Optional) show line console</td>
<td>Displays the console settings.</td>
</tr>
<tr>
<td>Example: switch(config)# show line console</td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> (Optional) copy running-config startup-config</td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
<tr>
<td>Example: switch(config)# copy running-config startup-config</td>
<td></td>
</tr>
</tbody>
</table>
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2**                      | Enters line configuration mode.                   |
| line vty                        |                                                   |
| **Example:**                    |                                                   |
| switch# line vty                |                                                   |
| switch(config-line)#            |                                                   |

| **Step 3**                      | Configures the inactive session timeout. The range is from 0 to 525600 minutes (8760 hours). A value of 0 minutes disables the timeout. The default value is 30. |
| exec-timeout minutes            |                                                   |
| **Example:**                    |                                                   |
| switch(config-line)# exec-timeout 30 |                                                   |

| **Step 4**                      | Exits line configuration mode.                    |
| exit                            |                                                   |
| **Example:**                    |                                                   |
| switch(config-line)# exit       |                                                   |
| switch(config)#                 |                                                   |

| **Step 5**                      | Displays the virtual terminal configuration.     |
| (Optional) show running-config all | being vty                                           |
| **Example:**                    |                                                   |
| switch(config)# show running-config all | being vty                                           |
| vty                             |                                                   |

| **Step 6**                      | Copies the running configuration to the startup configuration. |
| (Optional) copy running-config startup-config |                                                   |
| **Example:**                    |                                                   |
| switch(config)# copy running-config startup-config |                                                   |

### Configuring the Session Limit

You can limit the number of virtual terminal sessions on your device.

### SUMMARY STEPS

1. configure terminal
2. line vty
3. session-limit sessions
4. exit
5. (Optional) show running-config all | being vty
6. (Optional) copy running-config startup-config
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure terminal&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch# configure terminal&lt;br&gt;switch(config)#</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>line vty&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch# line vty&lt;br&gt;switch(config-line)#</td>
<td>Enters line configuration mode.</td>
</tr>
<tr>
<td>Step 3</td>
<td>session-limit sessions&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-line)# session-limit 10</td>
<td>Configures the maximum number of virtual sessions for your device. The range is from 1 to 64. The default is 32.</td>
</tr>
<tr>
<td>Step 4</td>
<td>exit&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config-line)# exit&lt;br&gt;switch(config)#</td>
<td>Exits line configuration mode.</td>
</tr>
<tr>
<td>Step 5</td>
<td>(Optional) show running-config all</td>
<td>Displays the virtual terminal configuration.</td>
</tr>
<tr>
<td>Step 6</td>
<td>(Optional) copy running-config startup-config&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch(config)# copy running-config startup-config</td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
</tbody>
</table>

## Clearing Terminal Sessions

You can clear terminal sessions on your device.

### SUMMARY STEPS

1. (Optional) show users
2. clear line name

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>(Optional) show users&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;switch# show users</td>
<td>Displays the user sessions on the device.</td>
</tr>
</tbody>
</table>
**Clears a terminal session on a specific line. The line name is case sensitive.**

**Example:**

```
switch# clear line pts/0
```

### Displaying Terminal and Session Information

To display terminal and session information, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show terminal</code></td>
<td>Displays terminal settings.</td>
</tr>
<tr>
<td><code>show line</code></td>
<td>Displays the COM1 and console ports settings.</td>
</tr>
<tr>
<td><code>show users</code></td>
<td>Displays virtual terminal sessions.</td>
</tr>
<tr>
<td><code>show running-config [all]</code></td>
<td>Displays the user account configuration in the running configuration. The <code>all</code> keyword displays the default values for the user accounts.</td>
</tr>
</tbody>
</table>

### Additional References for Terminal Settings and Sessions

This section includes additional references for terminal settings and sessions on Cisco NX-OS devices.

### Related Documents for Terminal Settings and Sessions

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensing</td>
<td>Cisco NX-OS Licensing Guide</td>
</tr>
</tbody>
</table>
Basic Device Management

This chapter contains the following sections:

- About Basic Device Management, on page 101
- Licensing Requirements for Basic Device Management, on page 102
- Default Settings for Basic Device Parameters, on page 102
- Changing the Device Hostname, on page 103
- Configuring the MOTD Banner, on page 103
- Configuring the Time Zone, on page 104
- Configuring Summer Time (Daylight Saving Time), on page 105
- Manually Setting the Device Clock, on page 106
- Setting the Clock Manager, on page 107
- Managing Users, on page 108
- Verifying the Device Configuration, on page 109
- Additional References for Basic Device Management, on page 109

About Basic Device Management

This section provides information about basic device management.

Device Hostname

You can change the device hostname displayed in the command prompt from the default (switch) to another character string. When you give the device a unique hostname, you can easily identify the device from the command-line interface (CLI) prompt.

Message-of-the-Day Banner

The message-of-the-day (MOTD) banner displays before the user login prompt on the device. This message can contain any information that you want to display for users of the device.

Device Clock

If you do not synchronize your device with a valid outside timing mechanism, such as an NTP clock source, you can manually set the clock time when your device boots.
Clock Manager

The Cisco NX-OS device might contain clocks of different types that might need to be synchronized. These clocks are a part of various components (such as the supervisor, line card processors, or line cards), and each might be using a different protocol.

The clock manager provides a way to synchronize these different clocks.

Time Zone and Summer Time (Daylight Saving Time)

You can configure the time zone and summer time (daylight saving time) setting for your device. These values offset the clock time from Coordinated Universal Time (UTC). UTC is International Atomic Time (TAI) with leap seconds added periodically to compensate for the Earth's slowing rotation. UTC was formerly called Greenwich Mean Time (GMT).

User Sessions

You can display the active user session on your device. You can also send messages to the user sessions. For more information about managing user sessions and accounts, see the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Licensing Requirements for Basic Device Management

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Product</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS</td>
<td>Basic device management requires no license. Any feature not included in a license package is bundled with the nx-os image and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the Cisco NX-OS Licensing Guide.</td>
</tr>
</tbody>
</table>

Default Settings for Basic Device Parameters

This table lists the default settings for basic device parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOTD banner text</td>
<td>User Access Verification</td>
</tr>
<tr>
<td>Clock time zone</td>
<td>UTC</td>
</tr>
</tbody>
</table>
Changing the Device Hostname

You can change the device hostname displayed in the command prompt from the default (switch) to another character string.

SUMMARY STEPS

1. configure terminal
2. { hostname | switchname } name
3. exit
4. (Optional) copy running-config startup-config

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td>Step 2 { hostname</td>
<td>switchname } name</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Using the hostname command:</td>
<td></td>
</tr>
<tr>
<td>switch(config)# hostname Engineering1 Engineering1(config)#</td>
<td></td>
</tr>
<tr>
<td>Using the switchname command:</td>
<td></td>
</tr>
<tr>
<td>Engineering1(config)# switchname Engineering2 Engineering2(config)#</td>
<td></td>
</tr>
<tr>
<td>Step 3 exit</td>
<td>Exits global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Engineering2(config)# exit</td>
<td></td>
</tr>
<tr>
<td>Engineering2#</td>
<td></td>
</tr>
<tr>
<td>Step 4 (Optional) copy running-config startup-config</td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Engineering2# copy running-config startup-config</td>
<td></td>
</tr>
</tbody>
</table>

Configuring the MOTD Banner

You can configure the MOTD to display before the login prompt on the terminal when a user logs in. The MOTD banner has the following characteristics:

• Maximum of 80 characters per line
- Maximum of 40 lines

**SUMMARY STEPS**

1. configure terminal
2. banner motd delimiting-character message delimiting-character
3. exit
4. (Optional) show banner motd
5. (Optional) copy running-config startup-config

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch# configure terminal switch(config)#</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>banner motd delimiting-character message delimiting-character</th>
<th>Configures the MOTD banner. Do not use the delimiting-character in the message text.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong> switch(config)# banner motd #$Welcome to the Switch# switch(config)#</td>
<td><strong>Note</strong> Do not use &quot;$&quot; or &quot;%&quot; as a delimiting character.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
<th>exit</th>
<th>Exits global configuration mode.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong> switch(config)# exit switch#</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 4</th>
<th>(Optional) show banner motd</th>
<th>Displays the configured MOTD banner.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong> switch# show banner motd</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 5</th>
<th>(Optional) copy running-config startup-config</th>
<th>Copies the running configuration to the startup configuration.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong> switch# copy running-config startup-config</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Configuring the Time Zone**

You can configure the time zone to offset the device clock time from UTC.

**SUMMARY STEPS**

1. configure terminal
2. clock timezone zone-name offset-hours offset-minutes
3. exit
4. (Optional) show clock
5. (Optional) copy running-config startup-config

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>
| **Example:**
switch# configure terminal
switch(config)# | |
| **Step 2** clock timezone zone-name offset-hours offset-minutes | Configures the time zone. The zone-name argument is a 3-character string for the time zone acronym (for example, PST or EST). The offset-hours argument is the offset from the UTC and the range is from –23 to 23 hours. The range for the offset-minutes argument is from 0 to 59 minutes. |
| **Example:**
switch(config)# clock timezone EST -5 0 | |
| **Step 3** exit | Exits global configuration mode. |
| **Example:**
switch(config)# exit
switch# | |
| **Step 4** (Optional) show clock | Displays the time and time zone. |
| **Example:**
switch# show clock | |
| **Step 5** (Optional) copy running-config startup-config | Copies the running configuration to the startup configuration. |
| **Example:**
switch# copy running-config startup-config | |

### Configuring Summer Time (Daylight Saving Time)

You can configure when summer time, or daylight saving time, is in effect for the device and the offset in minutes.

### SUMMARY STEPS

1. configure terminal
2. clock summer-time zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes
3. exit
4. (Optional) show clock detail
5. (Optional) copy running-config startup-config
**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch# configure terminal switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> clock summer-time zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes</td>
<td>Configures summer time or daylight saving time.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch(config)# clock summer-time PDT 1 Sunday March 02:00 1 Sunday November 02:00 60</td>
<td>The zone-name argument is a three character string for the time zone acronym (for example, PST and EST). The values for the start-day and end-day arguments are Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. The values for the start-month and end-month arguments are January, February, March, April, May, June, July, August, September, October, November, and December. The value for the start-time and end-time arguments are in the format hh:mm. The range for the offset-minutes argument is from 0 to 1440 minutes.</td>
</tr>
<tr>
<td><strong>Step 3</strong> exit</td>
<td>Exits global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch(config)# exit switch#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> (Optional) show clock detail</td>
<td>Displays the configured MOTD banner.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch(config)# show clock detail</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> (Optional) copy running-config startup-config</td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch# copy running-config startup-config</td>
<td></td>
</tr>
</tbody>
</table>

**Manually Setting the Device Clock**

You can set the clock manually if your device cannot access a remote time source.

**Before you begin**

Configure the time zone.

**SUMMARY STEPS**

1. clock set time day month year
2. (Optional) show clock

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> clock set time day month year</td>
<td>Configures the device clock.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch# clock set 15:00:00 30 May 2013 Fri May 30 15:14:00 PDT 2013</td>
<td>The format for the time argument is ( hh : mm : ss ). The range for the day argument is from 1 to 31. The values for the month argument are January, February, March, April, May, June, July, August, September, October, November, and December. The range for the year argument is from 2000 to 2030.</td>
</tr>
</tbody>
</table>

| **Step 2** (Optional) show clock | Displays the current clock value. |
| **Example:** switch(config)# show clock |

**Setting the Clock Manager**

You can configure the clock manager to synchronize all the clocks of the components in the Cisco Nexus device.

**SUMMARY STEPS**

1. clock protocol protocol
2. (Optional) show run clock_manager

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> clock protocol protocol</td>
<td>Configures the clock manager.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch# clock protocol ntp</td>
<td>The values for the protocol argument are ntp, ptp, and none. The following describes the values:</td>
</tr>
<tr>
<td></td>
<td>• ntp — Synchronizes clocks with Network Time Protocol (NTP).</td>
</tr>
<tr>
<td></td>
<td>• ptp — Synchronizes clocks with Precision Time Protocol (PTP) as described by IEEE 1588.</td>
</tr>
<tr>
<td></td>
<td>• none — Uses clock set HH:MM:SS to set the supervisor clock.</td>
</tr>
<tr>
<td><strong>Note</strong> When none is used, the clock must be configured.</td>
<td></td>
</tr>
</tbody>
</table>
### Managing Users

You can display information about users logged into the device and send messages to those users.

#### Displaying Information about the User Sessions

You can display information about the user session on the device.

**SUMMARY STEPS**

1. show users

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show users</td>
<td>Displays the user sessions.</td>
</tr>
</tbody>
</table>

**Example:**

```
switch# show users
```

#### Sending a Message to Users

You can send a message to active users currently using the device CLI.

**SUMMARY STEPS**

1. (Optional) show users
2. send [session line] message-text

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Optional) show users</td>
<td>Displays the active user sessions.</td>
</tr>
</tbody>
</table>

**Example:**

```
switch# show users
```
### Verifying the Device Configuration

To verify the configuration, use one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config</td>
<td>Displays the running configuration.</td>
</tr>
<tr>
<td>show startup-config</td>
<td>Displays the startup configuration.</td>
</tr>
<tr>
<td>show time-stamp running-config last-changed</td>
<td>Displays the timestamp when the running configuration was last changed.</td>
</tr>
</tbody>
</table>

### Additional References for Basic Device Management

You can find additional information related to basic device management.

### Related Documents for Basic Device Management

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensing</td>
<td>Cisco NX-OS Licensing Guide</td>
</tr>
</tbody>
</table>
Using the Device File Systems, Directories, and Files

This chapter contains the following sections:

- About the Device File Systems, Directories, and Files, on page 111
- Licensing Requirements for File Systems, Directories, and Files, on page 113
- Guidelines and Limitations, on page 113
- Default Settings for File System Parameters, on page 113
- Configuring the FTP, HTTP, or TFTP Source Interface, on page 113
- Working with Directories, on page 114
- Working with Files, on page 117
- Working with Archive Files, on page 122
- Examples of Using the File System, on page 125
- Additional References for File Systems, on page 128

About the Device File Systems, Directories, and Files

This section describes file systems, directories, and files on the Cisco NX-OS device.

File Systems

The syntax for specifying a local file system is `filesystem :[/ / modules /]`. 
This table describes file systems that you can reference on your device.

**Table 18: File System Syntax Components**

<table>
<thead>
<tr>
<th>File System Name</th>
<th>Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bootflash</td>
<td>sup-active</td>
<td>Internal CompactFlash memory located on the active supervisor module used for storing image files, configuration files, and other miscellaneous files. The initial default directory is bootflash.</td>
</tr>
<tr>
<td></td>
<td>sup-local</td>
<td></td>
</tr>
<tr>
<td></td>
<td>sup-standby</td>
<td>Internal CompactFlash memory located on the standby supervisor module used for storing image files, configuration files, and other miscellaneous files.</td>
</tr>
<tr>
<td></td>
<td>sup-remote</td>
<td></td>
</tr>
<tr>
<td>volatile</td>
<td>—</td>
<td>Volatile random-access memory (VRAM) located on a supervisor module used for temporary or pending changes.</td>
</tr>
<tr>
<td>log</td>
<td>—</td>
<td>Memory on the active supervisor that stores logging file statistics.</td>
</tr>
<tr>
<td>system</td>
<td>—</td>
<td>Memory on a supervisor module used for storing the running-configuration file.</td>
</tr>
<tr>
<td>debug</td>
<td>—</td>
<td>Memory on a supervisor module used for debug logs.</td>
</tr>
</tbody>
</table>

**Directories**

You can create directories on bootflash; and external flash memory (usb1; and usb2;). You can navigate through these directories and use them for files.

**Files**

You create and access files on bootflash;: volatile:; usb1;: and usb2: filesystems. You can only access files on the system: filesystem. You can use the log: filesystem for debug log files.

You can download files, such as the nx-os image file, from remote servers using FTP, Secure Copy (SCP), Secure Shell FTP (SFTP), and TFTP. You can also copy files from an external server to the device, because the device can act as an SCP server.
Licensing Requirements for File Systems, Directories, and Files

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Product</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS</td>
<td>Using the file systems, directories, and files requires no license. Any feature not included in a license package is bundled with the nx-os image and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the Cisco NX-OS Licensing Guide.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

Guidelines and limitations for device file systems, directories, and files are as follows:

- The `show tech-support details` command cannot be terminated using Ctrl+Z. Instead, use Ctrl+C to terminate the command.

Default Settings for File System Parameters

This table lists the default settings for the file system parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default filesystem</td>
<td>bootflash:</td>
</tr>
</tbody>
</table>

Configuring the FTP, HTTP, or TFTP Source Interface

You can configure the source interface for the File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), or Trivial File Transfer Protocol (TFTP). This configuration allows you to use the IP address associated with the configured source interface when copy packets are transferred.

SUMMARY STEPS

1. configure terminal
2. [no] ip {ftp | http | tftp} source-interface {ethernet slot/port | loopback number}
3. (Optional) copy running-config startup-config
### Working with Directories

This section describes how to work with directories on the Cisco NX-OS device.

#### Identifying the Current Directory

You can display the directory name of your current directory.

**SUMMARY STEPS**

1. `pwd`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>pwd</code></td>
<td>Displays the name of your current directory.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>switch# pwd</code></td>
<td></td>
</tr>
</tbody>
</table>

#### Changing the Current Directory

You can change the current directory for file system operations. The initial default directory is bootflash:

**SUMMARY STEPS**

1. (Optional) `pwd`
2. `cd` `{directory | filesystem :[/]module /}[directory]`
Creating a Directory

You can create directories in the bootflash: and flash device file systems.

SUMMARY STEPS

1. (Optional) pwd
2. (Optional) cd {directory | filesystem :[/ module /][directory]}
3. mkdir [filesystem :[/ module /]]directory

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>(Optional) pwd</td>
<td>Displays the name of your current default directory.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# pwd</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>cd {directory</td>
<td>filesystem :[/ module /][directory]}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# cd usb1:</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>mkdir [filesystem :[/ module /]]directory</td>
<td>Creates a new directory. The filesystem argument is case sensitive. The directory argument is alphanumeric, case sensitive, and has a maximum of 64 characters.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# mkdir test</td>
<td></td>
</tr>
</tbody>
</table>

Displaying Directory Contents

You can display the contents of a directory.

SUMMARY STEPS

1. dir [directory | filesystem :[/ module /][directory]]
Deleting a Directory

You can remove directories from the file systems on your device.

**Before you begin**

Ensure that the directory is empty before you try to delete it.

**SUMMARY STEPS**

1. (Optional) `pwd`
2. (Optional) `dir filesystem://module/][directory]`
3. `rmdir filesystem://module/][directory]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> dir [directory</td>
<td>filesystem://module/][directory]]</td>
</tr>
<tr>
<td>Example: switch# dir bootflash:test</td>
<td></td>
</tr>
</tbody>
</table>

Deleting a Directory

You can remove directories from the file systems on your device.

**Before you begin**

Ensure that the directory is empty before you try to delete it.

**SUMMARY STEPS**

1. (Optional) `pwd`
2. (Optional) `dir filesystem://module/][directory]`
3. `rmdir filesystem://module/][directory]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> (Optional) <code>pwd</code></td>
<td>Displays the name of your current default directory.</td>
</tr>
<tr>
<td>Example: switch# pwd</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> (Optional) <code>dir filesystem://module/][directory]</code></td>
<td>Displays the contents of the current directory. The file system, module, and directory names are case sensitive. If the directory is not empty, you must delete all the files before you can delete the directory.</td>
</tr>
<tr>
<td>Example: switch# dir bootflash:test</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>rmdir filesystem://module/][directory]</code></td>
<td>Deletes a directory. The file system and directory names are case sensitive.</td>
</tr>
<tr>
<td>Example: switch# rmdir test</td>
<td></td>
</tr>
</tbody>
</table>

Accessing Directories on the Standby Supervisor Module

You can access all file systems on the standby supervisor module (remote) from a session on the active supervisor module. This feature is useful when copying files to the active supervisor modules requires similar files to exist on the standby supervisor module. To access the file systems on the standby supervisor module from a session on the active supervisor module, you specify the standby supervisor module in the path to the file using either `filesystem://sup-remote/` or `filesystem://sup-standby/`. 
Working with Files

This section describes how to work with files on the Cisco NX-OS device.

Moving Files

You can move a file from one directory to another directory.

⚠️ Caution

If a file with the same name already exists in the destination directory, that file is overwritten by the moved file.

You can use the move command to rename a file by moving the file within the same directory.

SUMMARY STEPS

1. (Optional) pwd
2. (Optional) dir [filesystem :// module /][directory]]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>(Optional) pwd</td>
</tr>
<tr>
<td>Example: switch# pwd</td>
<td>Displays the name of your current default directory.</td>
</tr>
<tr>
<td>Step 2</td>
<td>(Optional) dir [filesystem :// module /][directory]]</td>
</tr>
<tr>
<td>Example: switch# dir bootflash</td>
<td>Displays the contents of the current directory. The file system and directory name are case sensitive.</td>
</tr>
<tr>
<td>Step 3</td>
<td>move [filesystem :// module /][directory /]</td>
</tr>
<tr>
<td>Example: switch# move test old_tests/test1</td>
<td>Moves a file. The file system, module, and directory names are case sensitive. The target-filename argument is alphanumeric, case sensitive, and has a maximum of 64 characters. If the target-filename argument is not specified, the filename defaults to the source-filename argument value.</td>
</tr>
</tbody>
</table>

Copying Files

You can make copies of files, either within the same directory or on another directory. For more information, see the Cisco Nexus 9000 Series NX-OS Troubleshooting Guide.
Using the dir command to ensure that enough space is available in the target file system. If enough space is not available, use the delete command to remove unneeded files.

**SUMMARY STEPS**

1. (Optional) `pwd`
2. (Optional) `dir filesystem:// module /][directory]]`
3. `copy filesystem:// module /][directory /] source-filename | {filesystem :// module /][directory /] | directory /][target-filename]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** (Optional) `pwd`
**Example:**
`switch# pwd`

Displays the name of your current default directory.

| **Step 2** (Optional) `dir filesystem:// module /][directory]]`
**Example:**
`switch# dir bootflash`

Displays the contents of the current directory. The file system and directory name are case sensitive.

| **Step 3** `copy filesystem:// module /][directory /] source-filename | {filesystem :// module /][directory /] | directory /][source-filename] | {filesystem :// module /][directory /] | directory /][target-filename]`
**Example:**
`switch# copy test old_tests/test1`

Copies a file. The file system, module, and directory names are case sensitive. The source-filename argument is alphanumeric, case sensitive, and has a maximum of 64 characters. If the target-filename argument is not specified, the filename defaults to the source-filename argument value.

**Deleting Files**

You can delete a file from a directory.

**SUMMARY STEPS**

1. (Optional) `dir filesystem:// module /][directory]]`
2. `delete {filesystem :// module /][directory /] | directory /][filename]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** (Optional) `dir filesystem:// module /][directory]]`
**Example:**
`switch# dir bootflash:`

Displays the contents of the current directory. The file system and directory name are case sensitive.
Purpose Command or Action | Purpose
--- | ---
**Step 2** | **delete** {filesystem :// module /][directory /] | directory /} filename
**Example:** | switch# delete bootflash:old_config.cfg
| Deletes a file. The file system, module, and directory names are case sensitive. The source-filename argument is case sensitive. | **Caution** If you specify a directory, the delete command deletes the entire directory and all its contents.

---

**Displaying File Contents**

You can display the contents of a file.

**SUMMARY STEPS**

1. `show file [filesystem :// module /][directory /]filename`

**DETAILLED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | `show file [filesystem :// module /][directory /]filename`
**Example:** | switch# show file bootflash:test-results
| Displays the file contents. |

---

**Displaying File Checksums**

You can display checksums to check the file integrity.

**SUMMARY STEPS**

1. `show file [filesystem :// module /][directory /]filename {cksum | md5sum}`

**DETAILLED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | `show file [filesystem :// module /][directory /]filename {cksum | md5sum}`
**Example:** | switch# show file bootflash:trunks2.cfg cksum
| Displays the checksum or MD5 checksum of the file. |

---

**Compressing and Uncompressing Files**

You can compress and uncompress files on your device using Lempel-Ziv 1977 (LZ77) coding.
Displaying the Last Lines in a File

You can display the last lines of a file.

SUMMARY STEPS

1. tail [filesystem :[/ module /]] [directory /] filename [lines]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Displays the last lines of a file. The default number of lines is 10. The range is from 0 to 80 lines.</td>
</tr>
<tr>
<td>(Optional) tail [filesystem :[/ module /]] [directory /] filename [lines]</td>
<td>Display the last lines of a file. The default number of lines is 10. The range is from 0 to 80 lines.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# tail ospf-gr.conf</td>
<td></td>
</tr>
</tbody>
</table>

Redirecting show Command Output to a File

You can redirect show command output to a file on bootflash:, volatile:, or a remote server. You can also specify the format for the command output.

SUMMARY STEPS

1. (Optional) terminal redirection-mode {ascii | zipped}
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> (Optional) terminal redirection-mode {ascii</td>
<td>Sets the redirection mode for the show command output for the user session. The default mode is ascii.</td>
</tr>
<tr>
<td></td>
<td>zipped}</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td>switch# terminal redirection-mode zipped</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Step 2</strong> show-command &gt; [filesystem:// module /][directory]</th>
<th>Redirects the output from a show command to a file.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td>switch# show tech-support &gt; bootflash:techinfo</td>
</tr>
</tbody>
</table>

### Finding Files

You can find the files in the current working directory and its subdirectories that have names that begin with a specific character string.

### SUMMARY STEPS

1. (Optional) `pwd`
2. (Optional) `cd {filesystem:// module /}[directory] | directory`
3. `find filename-prefix`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> (Optional) <code>pwd</code></td>
<td>Displays the name of your current default directory.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# pwd</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** (Optional) `cd {filesystem:// module /}[directory] | directory} | Changes the default directory. |
|--------------------------------------------------------------|-------------------------------------------------------------------------|
| **Example:**                                               |
| switch# cd bootflash:test_scripts                           |

<table>
<thead>
<tr>
<th><strong>Step 3</strong> <code>find filename-prefix</code></th>
<th>Finds all filenames in the default directory and in its subdirectories beginning with the filename prefix. The filename prefix is case sensitive.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# find bgp_script</td>
<td></td>
</tr>
</tbody>
</table>

### Formatting the Bootflash

Use the `format bootflash:` CLI command to format the onboard flash memory (bootflash:). If the command errors out due to the **Deactivate all virtual-services and try again** error message, destroy
the Guest Shell using the guestshell destroy CLI command and rerun the format bootflash: command, for example,

```
switch# sh virtual-service list
Virtual Service List:
Name Status Package Name
--------------------------------------------------------
guestshell+ Activated guestshell.ova

switch#
```

```
switch# guestshell destroy
You are about to destroy the guest shell and all of its contents. Be sure to save your work. Are you sure you want to continue? (y/n) [n] y
```

```
switch# 2018 Jan 17 18:42:24 switch %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Deactivating virtual service 'guestshell+'
```

```
switch# format bootflash:
```

## Working with Archive Files

The Cisco NX-OS software supports archive files. You can create an archive file, append files to an existing archive file, extract files from an archive file, and list the files in an archive file.

### Creating an Archive File

You can create an archive file and add files to it. You can specify the following compression types:

- bzip2
- gzip
- Uncompressed

The default is gzip.

### SUMMARY STEPS

1. tar create {bootflash: | volatile:}archive-filename [absolute] [bz2-compress] [gz-compress] [remove] [uncompressed] [verbose] filename-list

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Creates an archive file and adds files to it. The filename is alphanumeric, not case sensitive, and has a maximum length of 240 characters. The absolute keyword specifies that the leading backslash characters (_) should not be removed from the names of the files.</td>
</tr>
</tbody>
</table>

Example:
Appending Files to an Archive File

You can append files to an existing archive file on your device.

**Before you begin**

You have created an archive file on your device.

**SUMMARY STEPS**

1. `tar append {bootflash: | volatile:} archive-filename [absolute] [remove] [verbose] filename-list`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`tar append {bootflash:</td>
<td>volatile:} archive-filename [absolute] [remove] [verbose] filename-list`</td>
</tr>
</tbody>
</table>
Extracting Files from an Archive File

You can extract files to an existing archive file on your device.

Before you begin
You have created an archive file on your device.

**SUMMARY STEPS**

1. `tar extract {bootflash: | volatile:}archive-filename [keep-old] [screen] [to {bootflash: | volatile:}[/ directory-name]] [verbose]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>`tar extract {bootflash:</td>
<td>volatile:}archive-filename [keep-old] [screen] [to {bootflash:</td>
</tr>
</tbody>
</table>

**Displaying the Filenames in an Archive File**

You can display the names of the files in an archive files using the `tar list` command.
tar list {bootflash: | volatile:}archive-filename

The archive filename is not case sensitive.

switch# tar list bootflash:config-archive.tar.gz
config-file
new-config

Examples of Using the File System

This section includes examples of how to use the file system on the Cisco NX-OS device.

Accessing Directories on Standby Supervisor Modules

This example shows how to list the files on the standby supervisor module:

switch# dir bootflash://sup-remote
...
16384 Jan 01 13:23:30 2011 lost+found/
297054208 Oct 21 18:55:36 2013 n9000-dk9.6.1.2.11.1.bin
...

Usage for bootflash://sup-remote
190361600 bytes used
19234234368 bytes free
21137850368 bytes total

This example shows how to delete a file on the standby supervisor module:

switch# delete bootflash://sup-remote/aOldConfig.txt

Moving Files

This example shows how to move a file on an external flash device:

switch# move usb1:samplefile usb1:mystorage/samplefile

This example shows how to move a file in the default file system:

switch# move samplefile mystorage/samplefile

Copying Files

This example shows how to copy the file called samplefile from the root directory of the usb1: file system to the mystorage directory:
Deleting a Directory

You can remove directories from the file systems on your device.

**Before you begin**

Ensure that the directory is empty before you try to delete it.

### SUMMARY STEPS

1. (Optional) `pwd`
2. (Optional) `dir [filesystem:// module/][directory]`
3. `rmdir [filesystem:// module/][directory]`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | (Optional) `pwd` | Displays the name of your current default directory.  
**Example:**  
```
switch# pwd
```
| **Step 2** | (Optional) `dir [filesystem:// module/][directory]` | Displays the contents of the current directory. The file system, module, and directory names are case sensitive. If the directory is not empty, you must delete all the files before you can delete the directory.  
**Example:**  
```
switch# dir bootflash:test
```
| **Step 3** | `rmdir [filesystem:// module/][directory]` | Deletes a directory. The file system and directory name are case sensitive.  
**Example:**  
```
switch# rmdir test
```
Displaying File Contents

This example shows how to display the contents of a file on an external flash device:

```
switch# show file usb1:test
configure terminal
interface ethernet 1/1
no shutdown
end
show interface ethernet 1/1
```

This example shows how to display the contents of a file that resides in the current directory:

```
switch# show file myfile
```

Displaying File Checksums

This example shows how to display the checksum of a file:

```
switch# show file bootflash:trunks2.cfg cksum
583547619
```

This example shows how to display the MD5 checksum of a file:

```
switch# show file bootflash:trunks2.cfg md5sum
3b94707198aabefcf46459de10c9281c
```

Compressing and Uncompressing Files

This example shows how to compress a file:

```
switch# dir
  1525859  Jul 04 00:51:03 2013 Samplefile
  ...
switch# gzip volatile:Samplefile
switch# dir
  266069  Jul 04 00:51:03 2013 Samplefile.gz
  ...
```

This example shows how to uncompress a compressed file:

```
switch# dir
  266069  Jul 04 00:51:03 2013 Samplefile.gz
  ...
switch# gunzip samplefile
switch# dir
  1525859  Jul 04 00:51:03 2013 Samplefile
  ...
```
Redirecting show Command Output

This example shows how to direct the output to a file on the bootflash: file system:

```
switch# show interface > bootflash:switch1-intf.cfg
```

This example shows how to direct the output to a file on external flash memory:

```
switch# show interface > usb1:switch-intf.cfg
```

This example shows how to direct the output to a file on a TFTP server:

```
switch# show interface > tftp://10.10.1.1/home/configs/switch-intf.cfg
Preparing to copy...done
```

This example shows how to direct the output of the show tech-support command to a file:

```
switch# show tech-support > Samplefile
Building Configuration ...
switch# dir
  1525859  Jul 04 00:51:03 2013 Samplefile
Usage for volatile://
  1527808 bytes used
  19443712 bytes free
  20971520 bytes total
```

Finding Files

This example shows how to find a file in the current default directory:

```
switch# find smm_shm.cfg
/usr/bin/find: ./lost+found: Permission denied
./smm_shm.cfg
./newer-fs/isan/etc/routing-sw/smm_shm.cfg
./newer-fs/isan/etc/smm_shm.cfg
```

Additional References for File Systems

This section includes additional information related to the file systems.

Related Documents for File Systems

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensing</td>
<td>Cisco NX-OS Licensing Guide</td>
</tr>
</tbody>
</table>
CHAPTER 11

Working with Configuration Files

This chapter contains the following sections:

• About Configuration Files, on page 129
• Licensing Requirements for Configuration Files, on page 130
• Guidelines and Limitations for Configuration Files, on page 130
• Managing Configuration Files, on page 130
• Configuration Archive and Configuration Log, on page 141
• Verifying the Device Configuration, on page 145
• Examples of Working with Configuration Files, on page 145
• Additional References for Configuration Files, on page 146

About Configuration Files

Configuration files contain the Cisco NX-OS software commands used to configure the features on a Cisco NX-OS device. Commands are parsed (translated and executed) by the Cisco NX-OS software when the system is booted (from the startup-config file) or when you enter commands at the CLI in a configuration mode.

To change the startup configuration file, you can either save the running-configuration file to the startup configuration using the `copy running-config startup-config` command or copy a configuration file from a file server to the startup configuration.

Types of Configuration Files

The Cisco NX-OS software has two types of configuration files, running configuration and startup configuration. The device uses the startup configuration (startup-config) during device startup to configure the software features. The running configuration (running-config) contains the current changes that you make to the startup-configuration file. The two configuration files can be different. You might want to change the device configuration for a short time period rather than permanently. In this case, you would change the running configuration by using commands in global configuration mode but not save the changes to the startup configuration.

To change the running configuration, use the `configure terminal` command to enter global configuration mode. As you use the Cisco NX-OS configuration modes, commands generally are executed immediately and are saved to the running configuration file either immediately after you enter them or when you exit a configuration mode.
To change the startup-configuration file, you can either save the running configuration file to the startup configuration or download a configuration file from a file server to the startup configuration.

**Related Topics**
- Saving the Running Configuration to the Startup Configuration, on page 130
- Downloading the Startup Configuration From a Remote Server, on page 133

## Licensing Requirements for Configuration Files

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Product</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS</td>
<td>Configuration files require no license. Any feature not included in a license package is bundled with the nx-os image and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the Cisco NX-OS Licensing Guide.</td>
</tr>
</tbody>
</table>

## Guidelines and Limitations for Configuration Files

Configuration file guidelines and limitations are as follows:

- Beginning with NX-OS 7.0(3)I7(4), the `reload timer` command is supported to enable a reboot after a delay of 5-60 seconds.

## Managing Configuration Files

This section describes how to manage configuration files.

### Saving the Running Configuration to the Startup Configuration

You can save the running configuration to the startup configuration to save your changes for the next time you that reload the device.

**SUMMARY STEPS**

1. (Optional) show running-config
2. copy running-config startup-config

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Optional) show running-config</td>
<td>Displays the running configuration.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>switch# show running-config</td>
<td></td>
</tr>
</tbody>
</table>
Copying a Configuration File to a Remote Server

You can copy a configuration file stored in the internal memory to a remote server as a backup or to use for configuring other Cisco NX-OS devices.

SUMMARY STEPS

1. copy running-config scheme:// server[/url/][filename]
2. copy startup-config scheme:// server[/url/][filename]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>copy running-config scheme:// server[/url/][filename]</td>
<td>Copies the running-configuration file to a remote server. For the scheme argument, you can enter tftp:, ftp:, scp:, or sftp:. The server argument is the address or name of the remote server, and the url argument is the path to the source file on the remote server. The server, url, and filename arguments are case sensitive.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch# copy running-config</td>
<td></td>
</tr>
<tr>
<td></td>
<td>tftp://10.10.1.1/sw1-run-config.bak</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>copy startup-config scheme:// server[/url/][filename]</td>
<td>Copies the startup-configuration file to a remote server. For the scheme argument, you can enter tftp:, ftp:, scp:, or sftp:. The server argument is the address or name of the remote server, and the url argument is the path to the source file on the remote server. The server, url, and filename arguments are case sensitive.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch# copy startup-config</td>
<td></td>
</tr>
<tr>
<td></td>
<td>tftp://10.10.1.1/sw1-start-config.bak</td>
<td></td>
</tr>
</tbody>
</table>

Example

This example shows how to copy the configuration file to a remote server:

```
switch# copy running-config
tftp://10.10.1.1/sw1-run-config.bak
switch# copy startup-config
tftp://10.10.1.1/sw1-start-config.bak
```
Downloading the Running Configuration From a Remote Server

You can configure your Cisco NX-OS device by using configuration files that you created on another Cisco NX-OS device and uploaded to a remote server. You then download the file from the remote server to your device using TFTP, FTP, Secure Copy (SCP), or Secure Shell FTP (SFTP) to the running configuration.

Before you begin

Ensure that the configuration file that you want to download is in the correct directory on the remote server.
Ensure that the permissions on the file are set correctly. Permissions on the file should be set to world-read.
Ensure that your device has a route to the remote server. Your device and the remote server must be in the same subnetwork if you do not have a router or a default gateway to route traffic between subnets.
Check connectivity to the remote server using the `ping` or `ping6` command.

**SUMMARY STEPS**

1. copy scheme:// server /[url] /[filename] running-config
2. (Optional) show running-config
3. (Optional) copy running-config startup-config
4. (Optional) show startup-config

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Downloads the running-configuration file from a remote server.</td>
</tr>
<tr>
<td>copy scheme:// server /[url] /[filename] running-config</td>
<td>For the scheme argument, you can enter <code>tftp</code>, <code>ftp</code>, <code>scp</code>, or <code>sftp</code>. The server argument is the address or name of the remote server, and the <code>url</code> argument is the path to the source file on the remote server. The server, <code>url</code>, and <code>filename</code> arguments are case sensitive.</td>
</tr>
<tr>
<td><strong>Example:</strong> switch# copy tftp://10.10.1.1/my-config running-config</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Displays the running configuration.</td>
</tr>
<tr>
<td>(Optional) show running-config</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> switch# show running-config</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
<tr>
<td>(Optional) copy running-config startup-config</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> switch# copy running-config startup-config</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Displays the startup configuration.</td>
</tr>
<tr>
<td>(Optional) show startup-config</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> switch# show startup-config</td>
<td></td>
</tr>
</tbody>
</table>

**Related Topics**

Copying Files, on page 125
Downloading the Startup Configuration From a Remote Server

You can configure your Cisco NX-OS device by using configuration files that you created on another Cisco NX-OS device and uploaded to a remote server. You then download the file from the remote server to your device using TFTP, FTP, Secure Copy (SCP), or Secure Shell FTP (SFTP) to the startup configuration.

This procedure disrupts all traffic on the Cisco NX-OS device.

Before you begin

Log in to a session on the console port.

Ensure that the configuration file that you want to download is in the correct directory on the remote server.

Ensure that the permissions on the file are set correctly. Permissions on the file should be set to world-read.

Ensure that your device has a route to the remote server. Your device and the remote server must be in the same subnetwork if you do not have a router or a default gateway to route traffic between subnets.

Check connectivity to the remote server using the `ping` or `ping6` command.

**SUMMARY STEPS**

1. write erase
2. reload
3. copy scheme :// server [/url /]filename running-config
4. copy running-config startup-config
5. (Optional) show startup-config

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>write erase</td>
<td>Erases the startup configuration file.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# write erase</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2**        |         |
| reload             | Reloads the Cisco NX-OS device. |
| **Example:**       |         |
| switch# reload     |         |
| This command will reboot the system. (y/n)? [n] y |
| ...                |         |
| Enter the password for "admin": <password> |
| ...                |         |
| Would you like to enter the basic configuration dialog (yes/no): n |
| switch#             |         |

| **Step 3**        |         |
| copy scheme :// server [/url /]filename running-config | Downloads the running configuration file from a remote server. |
| **Example:**      |         |
|
For the `scheme` argument, you can enter `tftp:`, `ftp:`, `scp:`, or `sftp:`. The `server` argument is the address or name of the remote server, and the `url` argument is the path to the source file on the remote server.

The server, `url`, and `filename` arguments are case sensitive.

### Step 4
```
copy running-config startup-config
```
**Example:**
```
switch# copy running-config
startup-config
```
Saves the running configuration file to the startup configuration file.

### Step 5
(Optional) `show startup-config`
```
Example:
switch# show startup-config
```
Displays the running configuration.

### Related Topics
- Copying Files, on page 125

## Copying Configuration Files to an External Flash Memory Device

You can copy configuration files to an external flash memory device as a backup for later use.

### Before you begin
- Insert the external Flash memory device into the active supervisor module.

### SUMMARY STEPS

1. (Optional) `dir {usb1: | usb2:}[directory/]`
2. `copy running-config {usb1: | usb2:}[directory/]filename`
3. `copy startup-config {usb1: | usb2:}[directory/]filename`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1
```
(Optional) dir {usb1: | usb2:}[directory/]
```
**Example:**
```
switch# dir usb1:
```
Displays the files on the external flash memory device.

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 2
```
copy running-config {usb1: | usb2:}[directory/]filename
```
**Example:**
```
switch# copy running-config
usb1:dsn-running-config.cfg
```
Copies the running configuration to an external flash memory device. The `filename` argument is case sensitive.

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 3
```
copy startup-config {usb1: | usb2:}[directory/]filename
```
**Example:**
```
```
Copies the startup configuration to an external flash memory device. The `filename` argument is case sensitive.
Copying the Running Configuration from an External Flash Memory Device

You can configure your device by copying configuration files created on another Cisco NX-OS device and saved to an external flash memory device.

Before you begin
Insert the external flash memory device into the active supervisor module.

SUMMARY STEPS

1. (Optional) `dir {usb1: | usb2:}[directory /]`
2. `copy {usb1: | usb2:}[directory /]filename running-config`
3. (Optional) `show running-config`
4. (Optional) `copy running-config startup-config`
5. (Optional) `show startup-config`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>switch# copy startup-config</code> <code>usb1:dsn-startup-config.cfg</code></td>
<td>Displays the files on the external flash memory device.</td>
</tr>
<tr>
<td><code>switch# dir usb1:</code></td>
<td></td>
</tr>
<tr>
<td><code>switch# copy usb1:dsn-config.cfg running-config</code></td>
<td>Copies the running configuration from an external flash memory device. The filename argument is case sensitive.</td>
</tr>
<tr>
<td><code>switch# show running-config</code></td>
<td>Displays the running configuration.</td>
</tr>
<tr>
<td><code>switch# copy running-config startup-config</code></td>
<td>Copies the running configuration to the startup configuration.</td>
</tr>
<tr>
<td><code>switch# show startup-config</code></td>
<td>Displays the startup configuration.</td>
</tr>
</tbody>
</table>
Copying the Startup Configuration From an External Flash Memory Device

You can recover the startup configuration on your device by downloading a new startup configuration file saved on an external flash memory device.

Before you begin

Insert the external flash memory device into the active supervisor module.

**SUMMARY STEPS**

1. (Optional) `dir {usb1: | usb2:}[directory /]`
2. `copy {usb1: | usb2:}[directory /]filename startup-config`
3. (Optional) `show startup-config`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Displays the files on the external flash memory device.</td>
</tr>
<tr>
<td>(Optional) `dir {usb1:</td>
<td>usb2:}[directory /]`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>switch# dir usb1:</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Copies the startup configuration from an external flash memory device. The filename argument is case sensitive.</td>
</tr>
<tr>
<td>copy {usb1:</td>
<td>usb2:}[directory /]filename startup-config</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>switch# copy usb1:dsn-config.cfg startup-config</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Displays the startup configuration.</td>
</tr>
<tr>
<td>(Optional) <code>show startup-config</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>switch# show startup-config</td>
</tr>
</tbody>
</table>

Related Topics

Copying Files, on page 125

Copying Configuration Files to an Internal File System

You can copy configuration files to the internal memory as a backup for later use.

**SUMMARY STEPS**

1. `copy running-config [filesystem :][directory /] [directory /]filename`
2. `copy startup-config [filesystem :][directory /] [directory /]filename`
**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Copies the running-configuration file to internal memory. The filesystem, directory, and filename arguments are case sensitive.</td>
</tr>
<tr>
<td>copy running-config [filesystem :][directory /]</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>switch# copy running-config</td>
</tr>
<tr>
<td></td>
<td>bootflash:sw1-run-config.bak</td>
</tr>
</tbody>
</table>

**Step 2** | Copies the startup-configuration file to internal memory. The filesystem, directory, and filename arguments are case sensitive. |
| copy startup-config [filesystem :][directory /] | | |
| Example: | switch# copy startup-config |
| | bootflash:sw1-start-config.bak |

**Related Topics**

* Copying Files, on page 117

---

**Rolling Back to a Previous Configuration**

Problems, such as memory corruption, can occur that make it necessary for you to recover your configuration from a backed up version.

---

**Note**

Each time that you enter a `copy running-config startup-config` command, a binary file is created and the ASCII file is updated. A valid binary configuration file reduces the overall boot time significantly. A binary file cannot be uploaded, but its contents can be used to overwrite the existing startup configuration. The `write erase` command clears the binary file.

---

**SUMMARY STEPS**

1. write erase
2. reload
3. copy configuration-file running-configuration
4. copy running-config startup-config

---

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Clears the current configuration of the switch.</td>
</tr>
<tr>
<td>write erase</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>switch# write erase</td>
</tr>
</tbody>
</table>

**Step 2** | Restarts the device. You will be prompted to provide an nx-os image file for the device to boot and run. |
| reload | | |
| Example: | switch# reload |
Removing the Configuration for a Missing Module

When you remove an I/O module from the chassis, you can also remove the configuration for that module from the running configuration.

![Note](image)

You can only remove the configuration for an empty slot in the chassis.

**Before you begin**

Remove the I/O module from the chassis.

**SUMMARY STEPS**

1. (Optional) show hardware
2. purge module slot running-config
3. (Optional) copy running-config startup-config

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**
((Optional) show hardware
Example:
switch# show hardware

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 2**
purge module slot running-config
Example:
switch# purge module 3 running-config

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3**
( Offensive) copy running-config startup-config
Example:
switch# copy running-config startup-config
Erasing a Configuration

You can erase the configuration on your device to return to the configuration defaults. "Configuration" refers to the startup configuration as seen in 'show startup'. No other internal application or process states are cleared.

You can erase the following configuration files saved in the persistent memory on the device:

- Startup
- Boot
- Debug

The write erase command erases the entire startup configuration, except for the following:

- Boot variable definitions
- The IPv4 and IPv6 configuration on the mgmt0 interface, including the following:
  - Address
  - Subnet mask
  - Default Gateway/Route in the management VRF

To remove the boot variable definitions and the IPv4/IPv6 configuration on the mgmt0 interface, use the write erase boot command. To remove all application persistency files such as patch rpms, third party rpms, application configuration in /etc directory other than configuration, use 'install reset'. This command was added as of the 7.0(3)I6(1) release.

Note
When there are multiple IPv6 default routes present in the management VRF, the default route that is displayed first in the show ipv6 static-route command for the management VRF just before using 'copy r s' gets restored after the write erase and reload.

Note
After you enter the write erase command, you must reload the ASCII configuration twice to apply the breakout configuration.

SUMMARY STEPS

1. write erase [boot | debug]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**
write erase [boot | debug] | Erases configurations in persistent memory. The default action erases the startup configuration.
The boot option erases the boot variable definitions and the IPv4 configuration on the mgmt0 interface. |
| **Example:**
switch# write erase
Warning: This command will erase the | |
Clearing Inactive Configurations

You can clear inactive QoS and/or ACL configurations.

**SUMMARY STEPS**

1. (Optional) `show running-config type inactive-if-config`
2. `clear inactive-config policy`
3. (Optional) `show inactive-if-config log`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**

(Optional) `show running-config type inactive-if-config`

*Example:*

```
# show running-config ipqos inactive-if-config
```

 Displays any inactive access control list (ACL) or quality of service (QoS) configurations.

The values for the type argument are `aclmgr` and `ipqos`.

- `aclmgr` — Displays any inactive configurations for aclmgr.
- `ipqos` — Displays any inactive configurations for qosmgr.

**Step 2**

`clear inactive-config policy`

*Example:*

```
# clear inactive-config qos
clear qos inactive config
```

Inactive if config for QoS manager is saved at `/bootflash/qos_inactive_if_config.cfg` for vdc default.

You can see the log file `@ show inactive-if-config log`.

Clears inactive configurations.

The values for the policy argument are `qos` and `acl`.

The following describes the values:

- `qos` — Clears inactive QoS configurations.
- `acl` — Clears inactive ACL configurations.
- `acl qos` — Clears inactive ACL configurations and inactive QoS configurations.

**Step 3**

(Optional) `show inactive-if-config log`

*Example:*

Displays the commands that were used to clear the inactive configurations.
**Configuration Archive and Configuration Log**

This section contains information on configuration archive and configuration log.

**Information About Configuration Archive**

The configuration archive is intended to provide a mechanism to store, organize, and manage an archive of the configuration files to enhance the configuration rollback capability provided by the `configure replace` command. Before configuration archiving was introduced, you could save copies of the running configuration using the `copy running-config destination-url` command, storing the replacement file either locally or remotely. However, this method lacked any automated file management. The configuration replace and configuration rollback provides the capability to automatically save copies of the running configuration to the configuration archive. These archived files serve as checkpoint configuration references and can be used by the `configure replace` command to revert to the previous configuration states.

The `archive config` command allows you to save configurations in the configuration archive using a standard location and filename prefix that is automatically appended with an incremental version number (and optional timestamp) as each consecutive file is saved. This functionality provides a means for consistent identification of saved configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The `show archive` command displays information for all configuration files saved in the configuration archive.

The configuration archive, wherein the configuration files are stored and are available for use with the `configure replace` command, can be located on the following file systems: bootflash, FTP, and TFTP.

**Note**
The TFTP and FTP for this feature use VRF management.

**Configuring the Characteristics of the Configuration Archive**

Before using the `archive config` command, the configuration archive must be configured. Complete the following steps to configure the characteristics of the configuration archive:

**SUMMARY STEPS**

1. `switch# configure terminal`
2. `switch(config)# archive`
3. `switch(config-archive)# path url`
4. `switch(config-archive)# maximum number`
5. `switch(config-archive)# time-period minutes`
6. `switch# write-memory`
7. `switch# archive config`
8. `switch# show archive log config all`
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> switch# configure terminal</td>
<td>Enters the global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> switch(config)# archive</td>
<td>Enters the archive configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config)# archive</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong> This command does not apply to Cisco Nexus 9300-EX, -FX, and -R Series switches.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> switch(config-archive)# path url</td>
<td>Specifies the location and the filename prefix for the files in the configuration archive.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-archive)# path bootflash:myconfig</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong> If a directory is specified in the path instead of the file, the directory name must be followed by a forward slash as follows: path flash:/directory / . The forward slash is not necessary after a filename; it is necessary only when specifying a directory.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> switch(config-archive)# maximum number</td>
<td>(Optional) Sets the maximum number of archive files of the running configuration to be saved in the configuration archive.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-archive)# maximum 14</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong> Before using this command, you must configure the path to specify the location and filename prefix for the files in the configuration archive.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> switch(config-archive)# time-period minutes</td>
<td>(Optional) Sets the time increment for automatically saving an archive file of the current running configuration in the configuration archive.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-archive)# time-period 10</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong> The minutes argument specifies how often, in minutes, to automatically save an archive file of the current running configuration in the configuration archive.</td>
<td></td>
</tr>
</tbody>
</table>
Before using this command, you must configure the path command to specify the location and filename prefix for the files in the configuration archive.

Enables the command. It is disabled by default. Entering this command causes an archive to occur when the command copy r s is performed.

Saves the current running configuration file to the configuration archive.

You must configure the path before using the archive config command.

Displays the configuration log entries for all the users.

Information About Configuration Log

The configuration change logging tracks the changes that are made to the running configuration by using the data in the accounting log. This configuration log tracks the changes that are initiated only through the CLI. Only complete commands that result in the invocation of action routines are logged. The following types of entries are not logged:

- Commands that result in a syntax error message
- Partial commands that invoke the device help system

The configuration log tracks the changes that are initiated only through the CLI. For each configuration command that is executed, the following information is logged:

- A configuration change sequence number
- The line from which the command was executed
- The name of the user that executed the command
Displaying Configuration Log Entries

To display the configuration log entries, the configuration change logging provides the `show archive log config all` command.

**SUMMARY STEPS**

1. `switch# show archive log config all`
2. `switch# show archive log config user username`
3. `switch# show archive log config user username first-index start-number [ last-index end-number ]`

**DETAILED STEPS**

**Step 1**

`switch# show archive log config all`

Displays the configuration log entries for all users

**Example:**

```
switch# show archive log config all

INDEX  LINE  USER          LOGGED COMMAND
1      console0 user01 | logging console 1
2      console0 user01 | logging monitor 2
3      console0 user02 | system default switchport shutdown
4      console0 user02 | interface mgmt0
5      console0 user02 | no shutdown
```

**Step 2**

`switch# show archive log config user username`

Displays the configuration log entries for the specified username.

**Example:**

```
The following example displays the configuration log entries for a specified username.

switch# show archive log config user user02

INDEX  LINE  USER          LOGGED COMMAND
3      console0 user02 | system default switchport shutdown
4      console0 user02 | interface mgmt0
5      console0 user02 | no shutdown
```
Step 3

switch# show archive log config user username first-index start-number [ last-index end-number ]

Displays the configuration log entries by the index numbers. If you specify a number for the optional last-index, all the log entries with the index numbers in the range from the value entered for the start-number through the end-number for the specified user are displayed.

Example:
The following example displays the configuration log entry numbers 4 and 5 for a user with the username, user02. The range for the first-index and last-index is 1 to 200000000.

switch# show archive log config user user02 first-index 4 last-index 5
Last Log cleared/wrapped time is : Wed Oct 19 00:53:08 2016

<table>
<thead>
<tr>
<th>INDEX</th>
<th>LINE</th>
<th>USER</th>
<th>LOGGED COMMAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>console0</td>
<td>user02</td>
<td>interface mgmt0</td>
</tr>
<tr>
<td>5</td>
<td>console0</td>
<td>user02</td>
<td>no shutdown</td>
</tr>
</tbody>
</table>

Verifying the Device Configuration

To verify the configuration, use one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config</td>
<td>Displays the running configuration.</td>
</tr>
<tr>
<td>show startup-config</td>
<td>Displays the startup configuration.</td>
</tr>
<tr>
<td>show time-stamp running-config last-changed</td>
<td>Displays the timestamp when the running configuration was last changed.</td>
</tr>
</tbody>
</table>

Examples of Working with Configuration Files

This section includes examples of working with configuration files.

Copying Configuration Files

This example shows how to overwrite the contents of an existing configuration in NVRAM:

switch# copy nvram:snapshot-config nvram:startup-config

Warning: this command is going to overwrite your current startup-config.
Do you wish to continue? [y/n] [y] y

Note

This command does not apply to Cisco Nexus 9300-EX Series switches.

This example shows how to copy a running configuration to the bootflash: file system:
Backing Up Configuration Files

This example shows how to back up the startup configuration to the bootflash: file system (ASCII file):

```
switch# copy startup-config bootflash:my-config
```

This example shows how to back up the startup configuration to the TFTP server (ASCII file):

```
switch# copy startup-config tftp://172.16.10.100/my-config
```

This example shows how to back up the running configuration to the bootflash: file system (ASCII file):

```
switch# copy running-config bootflash:my-config
```

Rolling Back to a Previous Configuration

To roll back your configuration to a snapshot copy of a previously saved configuration, you need to perform the following steps:

1. Clear the current running image with the `write erase` command.
2. Restart the device with the `reload` command.
3. Copy the previously saved configuration file to the running configuration with the `copy configuration-file running-configuration` command.
4. Copy the running configuration to the start-up configuration with the `copy running-config startup-config` command.

Additional References for Configuration Files

This section includes additional information related to managing configuration files.

Related Documents for Configuration Files

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensing</td>
<td>Cisco NX-OS Licensing Guide</td>
</tr>
</tbody>
</table>