Programmability for Cisco Catalyst 4500 Series Switches, Cisco IOS Software Configuration Guide

Cisco IOS XE Release 3.9.0E

First Published: August 2016

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Preface

This preface describes who should read this document, how it is organized, and its conventions. The preface also tells you how to obtain Cisco documents, as well as how to obtain technical assistance.

Audience

This guide is for experienced network administrators who are responsible for configuring and maintaining Catalyst 4500 Series Switches.

Organization

This guide is organized into the following chapters:

Chapter	Title	Description
Chapter 1	Configuring Programmability	Presents an overview the feature, the various components involved, and how you can configure it.
Chapter 2	Sample Configuration and Reference Information	Provides sample configuration information for DHCP server configuration
Chapter 3	Using NETCONF and RESTCONF	Provides information about how you can use both interfaces.

Conventions

This document uses the following typographical conventions:

Convention	Description
boldface font	Commands, command options, and keywords are in **boldface**.
italic font	Command arguments for which you supply values are in *italics*.
[ ]	Command elements in square brackets are optional.
{ x | y | z }	Alternative keywords in command lines are grouped in braces and separated by vertical bars.

Convention	Description
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A unquoted set of characters. Do not use quotation marks around the string because the string will include the quotation marks.
`screen` font	System displays are in `screen` font.
`boldface screen` font	Information you must enter verbatim is in **`boldface screen`** font.
`italic screen` font	Arguments for which you supply values are in *`italic screen`* font.
——►	This pointer highlights an important line of text in an example.
^	Represents the key labeled Control—for example, the key combination ^D in a screen display means hold down the Control key while you press the D key.
< >	Nonprinting characters such as passwords are in angle brackets.

Notes use the following conventions:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Cautions use the following conventions:

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Documentation

Refer to the following documents for additional Catalyst 4500 series information:

- Catalyst 4500 Series Switch Documentation Home

  http://www.cisco.com/en/US/products/hw/switches/ps4324/tsd_products_support_series_home.html

Hardware Documents

Installation guides and notes including specifications and relevant safety information are available at the following URLs:

- *Catalyst 4500 E-series Switches Installation Guide*

  http://www.cisco.com/en/US/docs/switches/lan/catalyst4500/hardware/catalyst4500e/installation/guide/Eseries.html

- For information about individual switching modules and supervisors, refer to the *Catalyst 4500 Series Module Installation Guide* at:

http://www.cisco.com/en/US/docs/switches/lan/catalyst4500/hardware/configuration/notes/OL_25315.html

- *Regulatory Compliance and Safety Information for the Catalyst 4500 Series Switches*

  http://www.cisco.com/en/US/docs/switches/lan/catalyst4500/hardware/regulatory/compliance/78_13233.html

- Installation notes for specific supervisor engines or for accessory hardware are available at:

  http://www.cisco.com/en/US/products/hw/switches/ps4324/prod_installation_guides_list.html

- Catalyst 4500-X hardware installation information is available at:

  http://www.cisco.com/en/US/products/ps12332/prod_installation_guides_list.html

Software Documentation

Software release notes, configuration guides, command references, and system message guides are available at the following URLs:

- Cisco 4500-X release notes are available at:

  http://www.cisco.com/en/US/products/ps12332/prod_release_notes_list.html

- Catalyst 4500E release notes are available at:

  http://www.cisco.com/en/US/products/hw/switches/ps4324/prod_release_notes_list.html

Software documents for the Catalyst 4500 E-Series, and Catalyst 4500-X Series switches are available at the following URLs:

- *Catalyst 4500 Series Software Configuration Guide*

  http://www.cisco.com/en/US/products/hw/switches/ps4324/products_installation_and_configuration_guides_list.html

- *Catalyst 4500 Series Software Command Reference*

  http://www.cisco.com/en/US/products/hw/switches/ps4324/prod_command_reference_list.html

Cisco IOS Documentation

Platform-independent Cisco IOS documentation may also apply to the Catalyst 4500 Series Switches. These documents and tools are available at the following URLs:

Cisco IOS XE 15.2E	
Cisco IOS Configuration Guides	http://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-15-2e/products-installation-and-configuration-guides-list.html

Cisco IOS XE 3E	
Cisco IOS XE Configuration Guides	http://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-xe-3e/products-installation-and-configuration-guides-list.html

Cisco IOS 12.4

Cisco IOS Configuration Guides	http://www.cisco.com/en/US/products/ps6350/products_installation_and_configuration_guides_list.html
Cisco IOS Command References	http://www.cisco.com/en/US/products/ps6350/prod_command_reference_list.html
Cisco IOS System Messages	http://www.cisco.com/en/US/products/ps6350/products_system_message_guides_list.html

Tools	
Command Lookup	http://tools.cisco.com/Support/CLILookup/cltSearchAction.do
Error Message Decoder	http://www.cisco.com/pcgi-bin/Support/Errordecoder/index.cgi

Commands in Task Tables

Commands listed in task tables show only the relevant information for completing the task and not all available options for the command. For a complete description of a command, refer to the command reference guide.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation*.

To receive new and revised Cisco technical content directly to your desktop, you can subscribe to the What's New in Cisco Product Documentation RSS feed. The RSS feeds are a free service.

Configuring Programmability

Programmability is supported only on Catalyst 4500-E Series Switches with Supervisor Engine 8-E, 8L-E, and the Catalyst 4500-X Series Switches. The feature is supported on all available license levels for these switches. This chapter describes how to configure the feature and includes the following major sections:

About Programmability

Overview

Programmability is about how you can use data modeling languages and protocols to interact with the operating system (Cisco IOS XE) of a switch.

The traditional way of interacting or communicating with Cisco networking devices, has been manual configuration, through the command line interface (CLI). As deployments become more complex, programmability of devices has enabled a shift from manual network provisioning and configuration to automation.

Managing device configuration programmatically enables you to:

- Configure and control at scale—You can automate network configuration while also overcoming difficulties posed by multiple platforms, multiple operating systems, and multiple vendor devices in your network.
- Check to make sure that dependencies are satisfied before committing a change; and also easily roll-back when changes are not consistently compatible across the network.

To address configuration and monitoring issues, the Internet Engineering Task Force (IETF) has defined new standards in network management:

- Yet Another Next Generation (YANG) data modeling—RFC 6020.
- Network Configuration Protocol (NETCONF)—RFC 6241
- Representational State Transfer Configuration Protocol (RESTCONF)–uses the same data models as defined for NETCONF using YANG (https://tools.ietf.org/html/draft-ietf-netconf-restconf-04).

On Catalyst 4500 Series Switches, the Programmability feature introduces the use of NetCONF and RestCONF interfaces. They reside in a container on the switch and provide interfaces that enable remote management. The YANG data models available with these interfaces determine the scope of functions or actions that can be performed. See Figure 1-1.

Programmability Components

This section describes the network management tools used for programmability, in detail:

- NetCONF—an XML-based protocol that you can use to request information from and make configuration changes to the switch. NetCONF Application Programming Interfaces (APIs) use Secure Shell Version 2 (SSHv2).
- RestCONF—a JSON-based protocol that serves as an additional programming interface to implement the equivalent of NetCONF. RestCONF APIs use HTTP methods.
- YANG models—A data modeling language that defines the payload on NETCONF protocol messages. Data models determine the scope and the kind of functions that can be performed by NetCONF and RestCONF APIs. The following data model is available:

  The Cisco **ned.yang** model—This is a configuration data model; it enables to you perform write (SET) operations. The IETF, or common models are not supported.

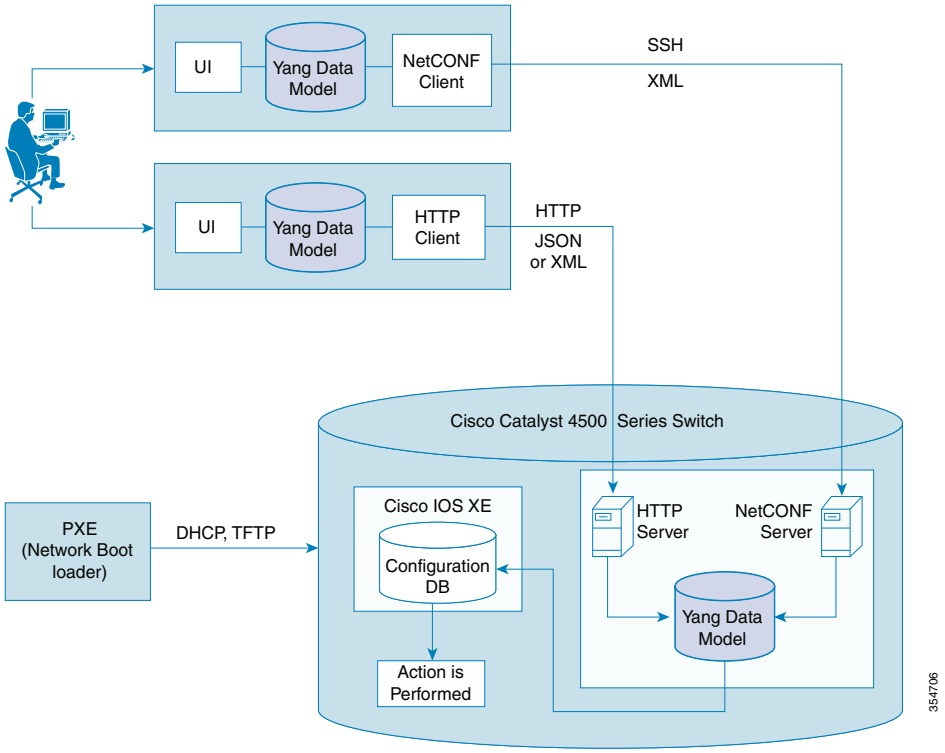
These components, enable you to set up what is required for Programmability:

- Virtual Services Container—Also referred to as a virtual machine (VM), virtual service, or container, is a virtual environment on a device.

  You can install an application within a virtual services container. The application then runs in the virtual services container of the operating system of a device. The application is delivered as an open virtual application (OVA), which is a tar file with a .ova extension. The OVA package is installed and enabled on a device through the device CLI.
- Data Model Interface (DMI)—A container that provides the NetCONF and RestCONF programmable interfaces. You must install and activate this container on the switch. After you activate it, the YANG models and APIs are available for use.
- Pre-Boot Execution Environment (PXE)—A network boot loader that enables a device to retrieve configuration files, scripts and .ova files from the remote DHCP server during initial deployment, without end-user intervention (zero-touch provisioning). You can boot the device and use TFTP to download user configuration files, scripts, and OVA files.

Figure 1-1 shows how the different components of Programmability come together.

Figure 1-1 Programmability Components



Default Configuration

Programmability is not enabled.

Configuring Programmability

You can configure this feature by means of zero touch provisioning (also known as Day 0 configuration) or the standard configuration method (by configuring all required tasks individually).

The following is relevant to both methods of configuration:

- Prerequisites for Configuring Programmability, page 1-4
- Restrictions and Limitations for Configuring Programmability, page 1-5
- PXE Requirements and Process Flow, page 1-6

For zero touch provisioning, you must ensure that you have met:

- Zero-Touch Provisioning Requirements, page 1-5

For the standard configuration method, you must complete the following:

- Installing the DMI Container, page 1-9
- Configuring OneP, page 1-10
- Providing Privilege Access to Use NetCONF and RestCONF, page 1-11
- Enabling Cisco IOS HTTP Services for RestCONF, page 1-11

Prerequisites for Configuring Programmability

- Prerequisites for NetCONF and RestCONF:

  Your access to the switch is configured with privilege level 15. This is required to start working with NetCONF and RestCONF interfaces. See Providing Privilege Access to Use NetCONF and RestCONF, page 1-11.

- To be able to download the device start-up configuration, script, and the ova files to the switch, you must use the Engineering Special image as the boot image:

  With the Catalyst 4500-X Series Switches, use the following boot image and .ova file name:

  - cat4500e-universalk9.SPA.03.09.00.PRT.1.152-5.0.1.PRT.bin
  - prt-1.0.0-r0-cat4500e.ova

  With the Catalyst 4500-E Series Switches, use the following boot image and .ova file name:

  - cat4500es8-universalk9.SPA.03.09.00.PRT.1.152-5.0.1.PRT.bin
  - prt-1.0.0-r0-cat4500es8.ova

- Prerequisites for PXE:

Note If you are not using the PXE to boot, you do not have to upgrade the ROMMON version.

  - The software configuration register is set to autoboot. PXE is supported only if you have enabled autoboot.

    Note For zero touch provisioning, the configuration register is set to autoboot by default.

  - The required ROMMON version is installed:

On Catalyst 4500-X Series Switches, ROMMON version 15.0(1r)SG13 applies.

On Catalyst 4500-E Series Switches, ROMMON version 15.1(1r)SG7 applies.

With the above ROMMON versions, the system prioritizes the PXE boot; if PXE is not available, it follows the usual order.

Restrictions and Limitations for Configuring Programmability

- The IETF, or common data models are not supported. Only the Cisco **ned.yang** model is supported for configuration.
- ISSU is not supported.
- IPv6 addresses are not supported on NETCONF and RESTCONF interfaces.
- The DMI is not supported in the VSS mode.
- Although there is no software restriction, we recommend that you have no more than 4 simultaneous NETCONF sessions.
- Do not use IP address 192.168.x.1 for communication, NETCONF is not supported if you do.
- RESTCONF is not supported with HTTPS.
- Zero touch provisioning (PXE boot) is not supported with Cisco Catalyst 4500E Supervisor Engines 8-E and 8L-E. On these devices you must install and activate the DMI .ova manually.
- NETCONF is not supported on an IP address assigned to a Switched Virtual Interface (SVI) where the port channels are members of that VLAN.

Zero-Touch Provisioning Requirements

For the zero-touch provisioning or Day 0 configuration, ensure that you have completed the following:

- Configured the DHCP server and TFTP server. For more information, see PXE Requirements —Configuring the DHCP Server, page 1-6
- Entered the following global configuration commands in the start-up configuration file. This file is downloaded during the PXE process
  - The **virtual-service *DMI*** command (The virtual service name must be DMI if one opts for day0 configuration).
  - The **activate** command
  - The **ip shared host-interface** *interface-id* command
  - The **onep** command
  - The **service set vty** command
  - The **username** *name* **privilege** level **password** *password* command
  - The **ip http server** command
  - The **ip http authentication local** command

The following is a sample of the device start-up configuration file with the required commands:

```
Switch #show running-config

Building configuration...

<output truncated>
!
username dmi_admin privilege 15 password 0 dmi_admin
<output truncated>
!
interface GigabitEthernet3/47
no switchport
ip address 10.106.18.158 255.255.255.128
!
<output truncated>
ip http server
ip http authentication local
ip route 0.0.0.0 0.0.0.0 10.106.18.129
!
!
!
line con 0
stopbits 1
line vty 0 4
login local
transport input telnet ssh
!
scheduler runtime netinput 100
onep
service set vty
netconf ssh
virtual-service dmi
activate
ip shared host-interface Vlan10
end
```

PXE Requirements and Process Flow

PXE Requirements —Configuring the DHCP Server

To send switch startup configuration files, scripts and .ova files in addition to the bootable image, you must configure the DHCP server.

Depending on your existing DHCP server setup (whether on Microsoft Windows or Linux), ensure that you have made the corresponding, requisite settings.

See Sample Configuration and Reference Information, page 2-1.

DHCP Configuration Guidelines:

- In the DHCP configuration file:

  The following information is mandatory: gateway, subnet mask and TFTP server IP address, and the client IP address in the DHCP configuration file. For example:

  ```
  option routers 192.168.20.2;
  ```

```
  option subnet-mask 255.255.255.0;
  next-server 10.106.24.187;

subnet 192.168.20.0 netmask 255.255.255.0 {
  pool {
    allow members of "WS-X45-SUP8L-E";
    range 192.168.20.10 192.168.20.50;
  }
  pool {
    allow members of "WS-4500X-16";
    range 192.168.20.51 192.168.20.100;
  }
}
```

The following information is optional. Depending on your requirement, you can specify one or more options: the boot image name, the start-up configuration file name and path, the script file name and path, and the ova file name and path. For example:

```
filename "iosimage.bin"
#ENTER A FILE NAME. MAKE SURE THAT CONFIG, SCRIPT, AND CONTAINER FILE EXTENTIONS ARE
<config-file>.config,<script-file>.script,<container-file>.ova RESPECTIVELY.
        option EXAMPLE.startup-config "configs/sup8le.config";
        option EXAMPLE.user-script "scripts/hello.script";
        option EXAMPLE.user-ova "container/cat4500e_20160801-172004_47.ova";
        option dhcp-parameter-request-list 43,3;
```

If you are using the above optional parameters, you must use the Engineering Special image as the boot image to be able to download the device start-up configuration, script, and the ova files to the switch.

- When the DHCP server responds successfully, the output displays `Received DHCP_ACK.`
- If you receive a TFTP timeout error, increase the DHCP timeout by using a ROMMON variable *DhcpTimeout*. The default DHCP timeout is 5 seconds. You can increase the DHCP timeout by a maximum of 30 seconds. For example, if **`DhcpTimeout=20`**, the DHCP timeout increases by 20 seconds.
- You can interrupt the autoboot process at any point, by pressing Control +C (switches to the ROMMON mode).
- The device configuration file, scripts and ova files should be saved in the TFTP root folder. This applies to DHCP server configuration using the Microsoft Windows and Linux.
- DHCP information such as IP address, gateway etc., are not permanently stored on switch. They are used only to download files and are deleted when the activity is complete.
- The DHCP boot ignores network information that you configure on the ROMMON, such as IP, gateway, subnet mask etc.

PXE Process Flow

If you have completed the required DHCP server configuration, the PXE follows the sequence of events given below.

1. The switch sends a DHCP discovery packet.
2. The DHCP server responds with an offer containing the TFTP server IP address, the offered IP address for the client, the gateway IP address, the boot file name, and the path and names of the OVA, script, and switch configuration files.
3. The switch sends the DHCP request for the IP address.

4. After the switch receives the DHCP acknowledgment packet from the server, the configuration file and OVA file information is cached in the flash 0 user partition.
5. The switch boots or powers up with the image specified in the *filename* variable in the DHCP configuration file.
6. During bootup, the switch checks for device configuration files, script files, and ova files. If there are such files, the switch sends the file information using DHCP Option 43 and downloads the required files.

The following is sample output of the autoboot process:

```
rommon 2 >
Rommon (G) Signature verification PASSED
Rommon (P) Signature verification PASSED
FPGA   (P) Signature verification PASSED

**********************************************************
*                                                        *
* Welcome to Rom Monitor for    WS-C4500X-16 System.     *
* Copyright (c) 2008-2014 by Cisco Systems, Inc.         *
* All rights reserved.                                   *
*                                                        *
**********************************************************

Rom Monitor (P) Version 15.0(1r)SG13
CPU Rev: 2.2, Board Rev: 9, Board Type: 108
CPLD Mobat Rev: 3.0x74b8.0x01db
Chassis: WS-C4500X-16

MAC Address  : 4c-4e-35-97-10-ff
Ip Address   : Not set.
Netmask      : Not set.
Gateway      : Not set.
TftpServer   : Not set.

Non-Redundant system or peer not running IOS
System Uplinks & Linecards have been reset!!

***** The system will autoboot in 5 seconds *****

Type control-C to prevent autobooting.
. . .
Management Ethernet Link Up: 1Gb Full Duplex
Received DHCP_ACK . . .
DHCP
Bootfile:tftp://10.106.24.187/cat4500e-universalk9.SSA.03.09.00.PR4.46.152-5.0.46.
PR4.bin
```

Note If you are not using PXE to boot, but are still using the new ROMMON versions, the following is displayed at the beginning of the boot process. You can ignore this. The boot process resumes normally.

```
***** The system will autoboot in 5 seconds *****

 Type control-C to prevent autobooting.
 . . .
 Management Ethernet Link Up: 1Gb Full Duplex
 Sending DHCP_DISCOVER    . . . .

 ******** The system will autoboot now ********
```

Installing the DMI Container

This task is mandatory if you have opted for the standard configuration method.

Before you begin, ensure that you have completed the following:

- Downloaded an OVA package that is compatible with the device operating system. The OVA package is available for download in the same location as your system image (.bin) file.
- Ensured that the minimum required disk space - 512 MB, and memory - 256 MB RAM is available on the device for installation and deployment of the DMI container.

To install and activate the DMI by using the virtual services container CLI, perform the following task:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**virtual-service install name** *virtual-services-name* **package** *file* **Example:** `Switch# virtual-service install name dmi package bootflash:/dmi.ova`	Installs an OVA package from the specified location onto a device. Ensure that the ova file is located in the root directory of the storage device.
Step 3	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 4	[**no**] **virtual-service** *virtual-services-name*
	
	Example:
	`Switch (config)# virtual-service dmi`
	`Switch (config-virt-serv)#`
		Configures a virtual services container and enters virtual services configuration mode. Observe these guidelines:
		• Use the virtual-services-name defined during installation of the application.
		• Ensure that installation is complete before proceeding to the next step using the show virtual-service list command.

Step 5	[**no**] **activate** **Example:** `Switch (config-virt-serv)# activate`	Activates the installed virtual services container.

	Command or Action	Purpose
Step 6	**ip shared host-interface** *interface-id* **Example:** `Switch (config-virt-serv)# ip shared host-interface gigabitethernet 3/47`	Maps the virtual service container to the interface that you specify. The IP address of the interface you specify here is used for NETCONF and RESTCONF communication. Observe these guidelines: **Note** You cannot configure a port channel interface as a shared interface. All other interface types are supported. **Note** If you want to change the shared interface that you have configured, enter the same command with the new interface that you want to use. The **no** form of this command is not supported.
Step 7	**end** **Example:** `Switch# end`	Exits virtual services configuration mode and enters privileged EXEC mode.

Configuring OneP

This task is mandatory if you have opted for the standard configuration method.

To enable the requisite, internal OneP infrastructure, perform the following task:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 3	**onep**
	Example:
	`Switch(config)# onep`
	`Switch(config-onep)#`
		Enters the OneP configuration mode.

Step 4	**service set vty** **Example:** `Switch(config-onep)# service set vty`	Enable the VTY service set. The VTY service enables the OneP application to communicate with a network element via a virtual terminal.
Step 5	**end** **Example:** `Switch# end`	Exits onep configuration mode and enters privileged EXEC mode.

Providing Privilege Access to Use NetCONF and RestCONF

This task is mandatory for both zero touch provisioning, and the standard configuration method.

To start working with NetCONF and RestCONF APIs you must be a user with privilege level 15. To provide this, perform the following task:

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**username** *name* **privilege** *level* **password** *password* **Example:** `Switch (config)# username example-name privilege 15 password example_password`	Establishes a username-based authentication system. Configure the following keywords: • **privilege** *level*—Sets the privilege level for the user. For the programmability feature, it must be 15. • **password** *password*—Sets a password to access the CLI view.
Step 4	**end** **Example:** `Switch# end`	Exits global configuration mode and enters privileged EXEC mode.

With the above task completed, the NetCONF interface is available. See Examples for NETCONF RPCs, page 3-1

To use the RestCONF interface, you must perform one more task. See Enabling Cisco IOS HTTP Services for RestCONF, page 1-11.

Enabling Cisco IOS HTTP Services for RestCONF

This task is mandatory if you want to use the RestCONF interface and have opted for the standard configuration method.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**ip http server** **Example:** `Switch (config)# ip http server`	Enables the HTTP server on your system.
Step 4	**ip http authentication local** **Example:** `Switch(config-onep)# ip http authentication local`	Indicates that the login user name, password and privilege level access combination specified in the local system configuration (with the **username** global configuration command) should be used for authentication and authorization.
Step 5	**end** **Example:** `Switch# end`	Exits global configuration mode and enters privileged EXEC mode.

With the above task completed, the RESTCONF interface is available. See Examples for RESTCONF RPCs:, page 3-2

Monitoring Programmability

Use these commands in the privileged EXEC mode, to display the Programmability settings you have configured:

Table 1-1 Monitoring Programmability

Show Command	Purpose

show onep session all	Displays OneP session information. To verify if NetCONF and RestCONF interfaces are configured correctly, ensure that these three sessions are listed: NetworkElementSynchronizer, SyncFromDaemon and CiaAuthDaemon. The following is sample output for this command:
	`Switch # show onep session all`
	`ID Username State ReconnectTimer ConnectTime ApplicationName`
	`8145 Connected 0 Thu Jul 28 06:07:05.304`
	`com.cisco.NetworkElementSynchronizer`
	`3234 Connected 0 Thu Jul 28 06:07:06.504 com.cisco.SyncFromDaemon`
	`7249 Connected 0 Thu Jul 28 06:07:07.343 com.cisco.CiaAuthDaemon`

show virtual-service [global]	Displays available memory, disk space, and CPU allocated for applications.
show virtual-service detail [**name** *virtual-services-name*]	Displays a list of resources committed to a specified application, including attached devices.

show virtual-service list	Displays the list of applications installed in the virtual services container. The following is sample output for this command:
	`Switch# show virtual-service list`
	`Virtual Service List:`
	`Name Status Package Name`
	`------------------------------------------------------------------------`
	`dmi Activated cat4500e_20160725-212823.ova`

show virtual-service storage pool list	Displays an overview of storage locations (pools) used for virtual service containers.
show virtual-service storage volume list	Displays an overview of storage volume information for virtual service containers.
show virtual-service version name *virtual-services-name* **installed**	Displays the version of an installed application.
show virtual-service tech-support	Displays container-based information.
show virtual-service redundancy state	Displays synchronization status
show virtual-service utilization statistics CPU	Displays virtual service CPU utilization statistics.

Troubleshooting Programmability

This section shows sample output for the some of the errors you may encounter while configuring the feature. In some cases a solution is described, and in others, sample configuration output serves as a guideline for correct configuration.

TFTP Timeout Error

If you receive a TFTP timeout error, increase the DHCP timeout by using a ROMMON variable *DhcpTimeout*. The default DHCP timeout is 5 seconds. You can increase the DHCP timeout by a maximum of 30 seconds. For example, if **`DhcpTimeout=20`**, the DHCP timeout increases by 20 seconds

File Not Found Errors

If you receive such an error, check the path you have entered for the `filename` field in the DHCP configuration file and make sure that the file exists in your TFTP server. See sample output below, it shows a successful TFTP session:

```
Filename     : /cat4500e-universalk9.SSA.03.09.00.PR4.46.152-5.0.46.PR4.bin
 IP Address   : 192.168.20.16
 Loading from TftpServer: 10.106.24.187
  TftpBlkSize  : 1468
 RxDataPacket : 130207

Loaded 191143008 bytes successfully.

Checking digital signature....
[/cat4500e-universalk9.SSA.03.09.00.PR4.46.152-5.0.46.PR4.bin]
Digitally Signed Development Software with key version A

Rommon reg: 0x00084F80
Reset2Reg: 0x00004F00

Image load status: 0x00000000
###
Winter 110 controller 0x0468AFAC..0x047F4313 Size:0x002FDB9D
 Program Done!
######################
[    0.058359] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[    0.148582] pci 0001:04:00.0: ignoring class b20 (doesn't match header type 01)
[    0.241172] pci 0002:0c:00.0: ignoring class b20 (doesn't match header type 01)
Starting System Services
devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=4,mode=600,ptmxmode=000 0 0

diagsk10-post version 5.1.4.1

prod: WS-C4500X-16 part: 73-13860-03 serial: JAE155209ZG


Power-on-self-test for Module 1: WS-C4500X-16

CPU Subsystem Tests ...
 seeprom: Pass
```

```
Traffic: L3 Loopback ...
 Test Results: Pass

Traffic: L2 Loopback ...
 Test Results: Pass
post done(56 secs)
Exiting to ios...
Downloading config files from 10.106.24.187 to /bootflash/pxe/user-startup-config
configs/4500x_start.config
.Received 2201 bytes in 0.0 seconds
Downloading script files from 10.106.24.187 to /bootflash/pxe/scripts
scripts/hello.script
.Received 90 bytes in 0.0 seconds
Downloading ova files from 10.106.24.187 to /bootflash/pxe/ova
container/cat4500e_20160717-183651_33.ova
.................Received 164270080 bytes in 32.0 seconds
Continuing with IOS boot..
Aug 1 06:23:42 %IOSXE-3-PLATFORM: process kernel: [ 124.746012]
mpc85xx_pci_err_probe: Unable to requiest irq 0 for MPC85xx PCI err
Aug 1 06:23:42 %IOSXE-3-PLATFORM: process kernel: [ 124.756621]
mpc85xx_pcie_err_probe: Unable to requiest irq 0 for MPC85xx PCIe err
Loading gsbu64atomic as gdb64atomic
Loading pds_helper module
Loading container module
Failed to bring interface "eth1" up
Using 1 for MTS slot
Platform Manager: starting in standalone mode (active)

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706

Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3 Switch  Software
(cat4500e-UNIVERSALK9-M), Version 03.09.00.PR4.46 EARLY DEPLOYMENT [PROD IMAGE]
ENGINEERING NOVA_WEEKLY BUILD, synced to  V152_5_1_E
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2016 by Cisco Systems, Inc.
Compiled Sun 31-Jul-16 16:31 by sabind

Cisco IOS-XE software, Copyright (c) 2005-2015 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.
(http://www.gnu.org/licenses/gpl-2.0.html) For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
```

```
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco WS-C4500X-16 (MPC8572) processor (revision 3) with 4194304K bytes of
physical memory.
Processor board ID JAE155209ZG
MPC8572 CPU at 1.5GHz, Cisco Catalyst 4500X
Last reset from Reload
1 Virtual Ethernet interface
16 Ten Gigabit Ethernet interfaces
511K bytes of non-volatile configuration memory.

Press RETURN to get started!
Switch>
```

Startup Configuration Errors

If you encounter errors when you replace existing startup configuration with new configuration, the system does not replace existing startup configuration. You must to resolve the errors in the device (switch) configuration file before resuming.

Debugging the DMI

To start debugging the DMI container:

Step 1 Set the logging level to “debug” in cisco-ia.yang model.

Step 2 Enter the following commands in the privilege EXEC Mode:

Note These are hidden commands and do not support tab or word help (the question mark (?) at the system prompt).

- show_ciam_log
- show_confd_log
- show_genet_log
- show_monit_log
- show_nes_log
- show_odm_log
- show_snmp_log
- show_sync_log
- show_wd_log
- show_all_logs

Step 3 To display NETCONF statistical information, such as, the number of sessions, netconf RPCs, packets and so on, use the ietf-netconf-monitoring.yang model.

Sample Configuration and Reference Information

This chapter provides sample DHCP server configurations. It includes the following sections:

DHCP Server Settings on Linux

The following is sample configuration that is saved in *dhcpd.conf* file. Use this as reference when you configure DHCP server settings on Linux.

This sample output covers a scenario where different files are sent to multiple devices of the same vendor specific class, but each one of the devices has a different MAC address.

Comments throughout the sample configuration provide guidelines for important steps.

Note You must restart the DHCP service every time you make a change in the *dhcpd.conf* file.

```
allow booting;
allow bootp;
ddns-update-style none;

#DEFINE AN OPTION SPACE. "EXAMPLE" IS USED HERE. IT IS A VARIABLE YOU CAN SET.
#MAINTAIN code 1,2 AND 3 CONSISTENTLY SINCE THE VALUES CORRESPOND TO CONFIG,SCRIPT AND
OVA FILES RESEPCTIVELY.
option space EXAMPLE;
option EXAMPLE.startup-config code 1=text;
option EXAMPLE.user-script code 2=text;
option EXAMPLE.user-ova code 3=text;

#ENTER THESE DETAILS AS APPLICABLE TO YOUR NETWORK
option domain-name "example.com";
option domain-name-servers 192.168.20.10, 192.168.10.10, 72.163.128.140;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.20.255;

#DEFINE A CLASS FOR THE VENDOR-SPECIFIC IDENTIFIER NAME THAT THE DEVICE HAS.
#EXAMPLE:FOR SUP8E/8LE IT IS "WS-X45-SUP8L-E"
#FOR CATALYST 4500-X IT IS "WS-4500X-16"
#ALSO DEFINE THE ROUTER,TFTP SERVER IDENTIFIER,NEXT SERVER IP DETAILS - AS APPLICABLE
TO YOUR NETWORK
```

```
class "WS-X45-SUP8L-E" {
  match pick-first-value (option dhcp-client-identifier, hardware);
  option routers 192.168.20.2;
  option subnet-mask 255.255.255.0;
  server-identifier 192.168.10.10;
  next-server 10.106.24.187;
}

class "WS-4500X-16" {
  match pick-first-value (option dhcp-client-identifier, hardware);
  option routers 192.168.20.2;
  option subnet-mask 255.255.255.0;
  server-identifier 192.168.10.10;
  next-server 10.106.24.187;
}

#DEFINE A SUBCLASS TO ADD THE DEVICE BASED ON IT'S MAC ADDRESS TO RECEIVE
CONFIGURATION FILES.
#THIS APPLIES WHEN YOU HAVE MULTIPLE DEVICES WITH SAME VENDOR-SPECIFIC IDENTIFIER AND
YOU WANT TO PUSH DIFFERENT CONFIGURATIONS BASED ON THE MAC ADDRESS

subclass "WS-X45-SUP8L-E" 1:e4:aa:5d:c4:a5:a6 {
# MENTION THE BOOTFILENAME.THIS .BIN IMAHE FILE SHOULD RESIDE IN THE TFTPBOOT FOLDER.
        filename "cat4500es8-universalk9.SSA.03.09.00.PR4.47.152-5.0.47.PR4.bin";
        option routers 192.168.20.2;

#SPECIFY THAT THE OPTION 43 AND ROUTER(3) DETAILS HAVE TO BE SENT TO THE CLIENT SWITCH
        option dhcp-parameter-request-list 43,3;
        option vendor-class-identifier "WS-X45-SUP8L-E";
        vendor-option-space EXAMPLE;

#SPECIFY THE PATH OF THE FILES YOU WANT TO SEND.
#MAKE SURE THESE FILES RESIDE IN IDENTICAL FOLDERS (configs/,scripts/,container/) IN
the TFTPBOOT FOLDER. YOU MUST CREATE THE IDENTICAL FOLDERS WITH THE SAME NAME AND
CASE.
#ENTER A FILE NAME. MAKE SURE THAT CONFIG, SCRIPT, AND CONTAINER FILE EXTENTIONS ARE
<config-file>.config,<script-file>.script,<container-file>.ova RESPECTIVELY.

        option EXAMPLE.startup-config "configs/sup8le.config";
        option EXAMPLE.user-script "scripts/hello.script";
        option EXAMPLE.user-ova "container/cat4500e_20160801-172004_47.ova";
        option dhcp-parameter-request-list 43,3;
}

subclass "WS-X45-SUP8L-E" 1:e4:aa:5d:c4:a5:a1 {
        filename "cat4500es8-universalk9.SSA.03.09.00.PR4.47.152-5.0.47.PR4.bin";
        option routers 192.168.20.2;
        option dhcp-parameter-request-list 43,3;
        option vendor-class-identifier "WS-X45-SUP8L-E";
        vendor-option-space EXAMPLE;
        option EXAMPLE1.startup-config "configs/sup8le-config.config";
        option EXAMPLE1.user-script "scripts/hello12.script";
        option EXAMPLE1.user-ova "container/cat4500es8_20160801-172004_47.ova";
        option dhcp-parameter-request-list 43,3;
}

subclass "WS-4500X-16" 1:30:e4:db:f8:a4:9f {
        filename "cat4500e-universalk9.SSA.03.09.00.PR4.47.152-5.0.47.PR4.bin";
        option routers 192.168.20.2;
        option dhcp-parameter-request-list 43,3;
        option vendor-class-identifier "WS-4500X-16";
        vendor-option-space EXAMPLE;
        option EXAMPLE1.startup-config "configs/4500X_start.config";
```

```
        option EXAMPLE1.user-script "scripts/hello12.script";
        option EXAMPLE1.user-ova "container/cat4500e_20160801-170415_47.ova";
        option dhcp-parameter-request-list 43,3;
}

#ASSIGN A POOL TO GIVE IP ADDRESSES TO THE MEMBERS OF THE VENDOR-SPECIFIC CLASS
subnet 192.168.20.0 netmask 255.255.255.0 {
  pool {
    allow members of "WS-X45-SUP8L-E";
    range 192.168.20.10 192.168.20.50;
  }
  pool {
    allow members of "WS-4500X-16";
    range 192.168.20.51 192.168.20.100;
  }
}

subnet 192.168.10.0 netmask 255.255.255.0 {
  range 192.168.10.12 92.168.10.100;
  option routers 192.168.10.10;
  option subnet-mask 255.255.255.0;
  server-identifier 192.168.10.10;
  next-server 10.106.24.187;
}
```

Configuring DHCP Option 43 (for Microsoft Windows)

DHCP Option 43 is used by clients and servers to exchange vendor-specific information. (RFC 2132).

This section describes the DHCP Option 43 configuration information that pertains to sending device configuration files, script files, and .ova files to the switch. It is applicable only if you use OpenDhcpServer as the DHCP server, with Microsoft Windows. Other DHCP servers have their own methods to configure this option and the information is available on the Internet.

To send any file, you must convert the file name along with the extension, to a hexadecimal format.

<File code><length of filename.ext in hexadecimal value><hex value of the filename.ext>

Use the relevant codes to specify the type of file you want to send

- code 01—A configuration file. For example, to send a `text.config` file, the format is:

  `43=<01>:<0B>:<74:65:78:74:2E:63:6F:6E:66:69:67>`

- code 02—A script file. For example to send a `t1.script` file, the format is:

  `43=<02>:<09?:74:31:2E:73:63:72:69:70:74,`

- code 03—A .ova file. For example, to send a `Sup8E.ova` file, the format is:

  `43=<03>:<09>:53:75:70:38:45:2E:6F:76:61`

This example concatenates the configuration, script, and .ova files:

```
43=01:0B:74:65:78:74:2e:63:6f:6e:66:69:67:02:09:74:31:2e:73:63:72:69:70:74:03:09:53:75:70:
38:45:2e:6f:76:61:ff
```

Microsoft Windows DHCP Server Configuration

The following example shows how to configure the DHCP Server on Microsoft Windows.

Note The example uses OpenDhcpServer and Solarwinds TFTP server. Information about configuring both is available on the Internet. The use of both applications here is only meant to serve as an example for configuration, and are not product recommendations.

Figure 2-1 Solarwinds TFTP Server

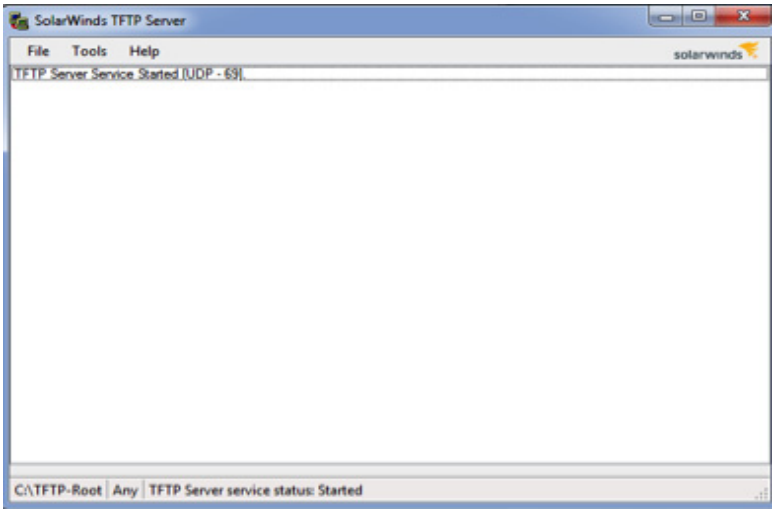
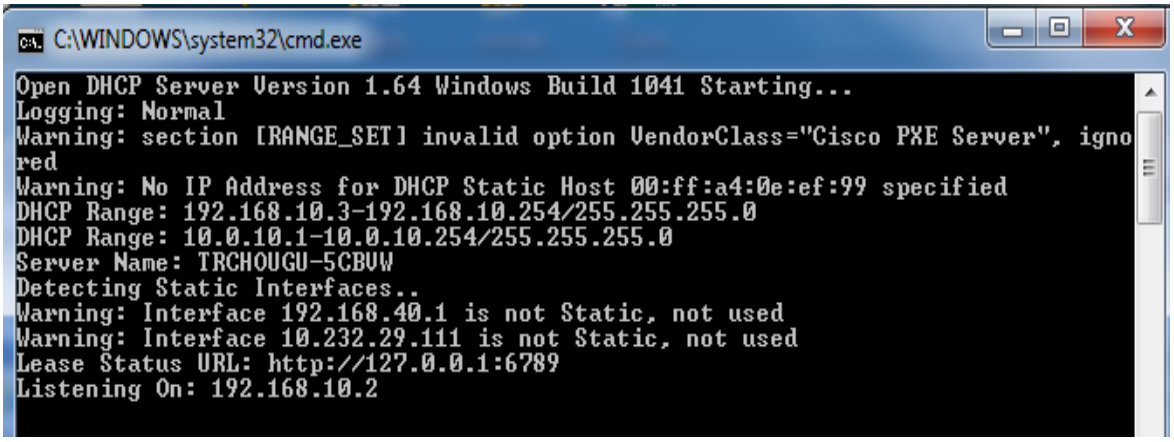


Figure 2-2 *OpenDHCPServer*



The important sections of this sample configuration are **bold**:

```
#This is configuration file.
#Any entry starting with a punctuation character will be a comment
#This file should be in the same folder where exe file is.
#You need to remove ; from begining of sample values and replace with
#your own values below if you need change settings
```

```
[LISTEN_ON]
#Specify the Interfaces you would like Server to listen
#if you have more than one NIC card on your server
#always specify which cards will listen DHCP/DNS requests
#Requests from diffent Interfaces look for matching DHCP ranges.
#Requests from relay agents look for matching range to relay agent IP.
#upto 125 interfaces can be specified
#Default is All static Interfaces
;192.168.0.1


[LOGGING]
#LogLevel can be set as None, Errors or All
#It is advisable to keep logging to Normal, Normal include errors
#and DHCP renewal messages. Normal is defaul logging also.
;LogLevel=None
;LogLevel=Normal
;LogLevel=All
;LogLevel=Debug


[REPLICATION_SERVERS]
#You can have 2 instances of Open DHCP Servers in a network. Open DHCP Server
#will send replication inform messages to other instance of Open DHCP
#server and leases will be replicated. The IP address alloted by one server
#will not be realotted by other server to another host. Also when one server
#goes down, other can will renew the leases, without NAK and DISCOVER. You need
#to specify Primary and secondary servers for replication to work.
#Make sure that Primary & Secondary Server entries are identical on both
#servers. You may copy the entire ini file on both servers and change the
#LISTEN_ON on individual servers, if needed.
;Primary=192.168.0.253
;Secondary=192.168.0.254


[HTTP_INTERFACE]
#This is http inerface for viewing lease status,
#Default is first interface, port 6789
#You can change it here to any network interface.
;HTTPServer=192.168.55.1:6789
#Also to limit the clients access, you can specify upto 8
#HTTP client IPs Here. If no Client IP is specified then All
#Clients can access the HTTP Interface
;HTTPClient=192.168.0.11
;HTTPClient=192.168.23.123
#You can also change the title of html page
;HTTPTitle=This is Custom Title


#Sections below are other DHCP Sections. Clients can be alotted addresses in
#two ways, dynamically from DHCP Range or statically. For static addresses,
#client section needs to be created for each static client
#against its MAC Address. BOOTP clients are allways static.
#The DHCP Ranges are grouped into [RANGE_SET]s, so that range specific options
#can be specified for a group of ranges at one place. The total ranges together
#in all [RANGE_SET]s is also 125 and there can also be 125 [RANGE_SET]s max.
#You can specify one or more ranges in each [RANGE_SET] section, in format
#specified. Open DHCP Server will alot addresses from these ranges. Static Hosts
#and BootP clients do not need ranges. No need to specify any [RANGE_SET]
#or DHCP_Range if all clients are Static.

#The Policy for alloting dynamic address is:-
#1)First Look if MacAddress is specified as Static DHCP Client and use that IP
#2)If not found look for old expired/active address of same host
```

```
#3)If not,look at requested IP Address and it is free
#4)If not, allot virgin IP Adress, if any available
#5)If no virgin IP address exists, allot expired IP address of other host.
#From 2) to 6), requests from diffent Interfaces look for matching DHCP ranges
#of Interface IP and requests from relay agents look for matching range to
#relay agent IP.

#All the ranges in a [RANGE_SET] section can be further restricted
#by Filter_Mac_Range, Filter_Vender_Class and Filter_User_Class
#If for example Mac Range is specified, then this section's ranges
#will only be available to hosts, whoes Mac Address
#Falls in this range. Also if any host has matching Filter_Mac_Range in
#any DHCP_RANGE section then other DHCP Range sections
#without Filter_Mac_Range or not having matching Mac Range will
#not be available to it. Each Manufacturer has a fixed Mac Range.
#Same Mac ranges can repeat in many DHCP_RANGE sections.
#For Filter_Vendor_Class (option 60) and Filter_User_Class filter (option 77),
#the range would only be available to matching value of Filter_Vender_Class
#and Filter_User_Class sent in client request. If Filter_Vender_Class and
#Filter_User_Class do match in one or more ranges, other ranges with missing
#or not matching values would not be available to such clients.
#You can specify upto 32 Filter_Mac_Range, Filter_Vender_Class and
#Filter_User_Class in each [RANGE_SET].

#Generally you dont have to specify any filters for relay agent. The range is
#automatically selected based on relay agent IP and range's subnetmask. Relay agent
#always sends it's subnet side IP. This server would only use the DHCP Range, which
#matches this IP. This would ensure that correct range is used. This feature
#eliminate the need of additional configuration. For matching purpose, range is
#recalculated using Subnet Mask of range and Relay Agent IP. However if you want
#to manually configure the subnet selection, you can use FilterSubnetSelection in
#a RANGE_SET. If this fitler is specified it will be first matched with SubnetSelection
#Option 118 sent by client. If client sends no such option, it will be matched
#with relay Agent IP. If not relay agent IP is sent, Listening Interface's IP
#will be matched. You can also override the Target Relay Agent using TargetRelayAgent
option.

[RANGE_SET]
#This is first and simple DHCP range section example,
#This example may be good enough for simple/home use.
#If you need range filters, look at example below
DHCPRange=192.168.10.3-192.168.10.254
VendorClass="Cisco PXE Server"
43=01:0B:74:65:78:74:2e:63:6f:6e:66:69:67:02:09:74:31:2e:73:63:72:69:70:74:03:09:53:75:70:
38:45:2e:6f:76:61
;43="text.config"01:0B:74:65:78:74:2E:63:6F:6E:66:69:67,"t1.script"02:09:74:31:2E:73:63:72
:69:70:74,";;Sup8E.ova"03:09:53:75:70:38:45:2E:6F:76:61
#Following are range specific DHCP options.
#You can copy more options names from [GLOBAL_OPTIONS]
SubnetMask=255.255.255.0
;DomainServer=192.168.10.2
Router=192.168.10.2
#Lease Time can be different for this Range
;AddressTime=360

[RANGE_SET]
#This section is also simple [RANGE_SET] section
#Here the options are specified as flat options.
;DHCPRange=192.168.0.1-192.168.0.254
;DHCPRange=192.168.4.1-192.168.4.254
;DHCPRange=192.168.5.1-192.168.5.254
#Following are flat range specific DHCP options.
#SubnetMask below
;1=255.255.255.0
```

```
#DomainServers below
;6=192.168.0.1, 192.168.0.2
#Router
;3=192.168.0.1
#AddressTime
;51=11000


[RANGE_SET]
#This is filtered [RANGE_SET] section.
#First eight entries in this example are filters.
#Currently only following types of filters are supported
#However 32 filters of each type can be specified
;FilterMacRange=00:0d:60:c5:4e:00-00:0d:60:c5:4e:ff
;FilterMacRange=00:0e:12:c5:4e:00-00:0e:12:c5:4e:ff
;FilterMacRange=00:0f:60:c5:4e:a1-00:0f:60:c5:4e:a1
;FilterVendorClass="MSFT 5.0"
;FilterVendorClass="MSFT 5.1"
;FilterVendorClass="MSFT 5.2"
;FilterUserClass="My User Class 4.0"
;FilterUserClass=123,56,87,123,109,0,23,56,156,209,234,56
;FilterUserClass=00:0d:60:c5:4e:0d:60:c5:4e
#You can select RANGE_SET based on FilterSubnetSelection
;FilterSubnetSelection=192.168.55.1
;FilterSubnetSelection=192.168.33.1
;TargetRelayAgent=192.168.44.11
#Next few are actual ranges of this section.
;DHCPRange=10.0.0.5-10.0.0.10
DHCPRange=10.0.10.1-10.0.10.254
;DHCPRange=10.0.1.1-10.0.1.254
;DHCPRange=10.0.2.1-10.0.2.254
#Following are range specific DHCP options.
#You can copy more option names from [GLOBAL_OPTIONS]
#or add flat options like 240="this is the string value"
#or as IP like 6=192.168.5.1
#or byte array like 6=123,45,1,0,3,67,4,3,22,4,3,5
#or hex array like 6=23:89:a5:ba:a9:e4
;SubnetMask=255.255.255.0
;DomainServer=10.5.6.90, 11.4.5.6
;Router=11.5.6.7, 10.0.99.1
#AddressTime can be different for this range
#specify 0 for infinity.AddressTime
;AddressTime=360
;Ethernet=no
;NETBIOSNameSrv=192.168.0.201
#You can also use hex array or byte array with named options
#If you want to send option 43 back to client for
#ranges in this section, specify it as flat option like:-
;43="this is return string"
#or use the byte array in value
;43=123,56,87,123,109,0,23,56,156,209,234,56
#or use the hex array in value
;43=a6:87:b6:c9:ae:eb:89:09:a4:67:d5

[GLOBAL_OPTIONS]
#These are global DHCP Options and would supplement
#client specific options and [RANGE_SET] options.
#Options tags start with 1 and goes up to 254, you can
#always specify option like 1=255.255.255.0, but it may
#be difficult to remember option tags. Try using Option Names
#If no matching name found, you can use tag=value (flat options)
#You can also specify the value as byte array or even hex array.
#Some options having sub-options can only be specified as hex/byte
#array If options have client specific values, move/copy them
```

```
#to specific Static Client's section. If any option has DHCP range
#specific value, move/copy them to [RANGE_SET] sections.
#You may quote stings values (must quote if sting contain chars
#like comma, dot or colon) for example NDS_Tree_Name="my.NDS.Tree"
#or 43="this is return string" or use the byte array in value
#like 43=123,56,87,123,109,0,23,56,156,209,234,56 or use the hex
#array in value 43=a6:87:b6:c9:ae:eb:89:09:a4:67:d5

;DomainName="workgroup.com"
;SubNetMask=255.255.255.0
;DomainServer=192.168.1.1, 192.168.1.2
;Router=192.168.1.1
#AddressTime is default lease time for server
#specify 0 for infinity lease time
;AddressTime=36000
;RenewalTime=0
;RebindingTime=0
#NextServer is PXEBoot TFTP Server
NextServer=192.168.10.2
;TimeOffset=3000
;TimeServer=192.168.0.1
;NameServer=192.168.0.1
;LogServer=192.168.0.1
;QuotesServer=192.168.0.1
;LPRServer=192.168.0.1
;ImpressServer=192.168.0.1
;RLPServer=192.168.0.1
;BootFileSize=2345
;SwapServer=192.168.0.1
;RootPath=/opt/boot/
;ExtensionFile=bootdir/files
;ForwardOn/Off=yes
;SrcRteOn/Off=yes
;PolicyFilter=192.168.34.1/255.255.255.240
;DefaultIPTTL=234
;MTUTimeout=3453
;MTUPlateau=ac:c0:12:09:02:24:0a:4D:61:63:20:48:44:5F:4E:42:53
;MTUInterface=23553
;MTUSubnet=yes
;BroadcastAddress=192.168.0.255
;MaskDiscovery=yes
;MaskSupplier=yes
;RouterDiscovery=yes
;RouterRequest=192.168.67.1
;StaticRoute=192.168.11.1/255.255.255.0, 192.168.12.1/255.255.255.0
;Trailers=yes
;ARPTimeout=3453
;Ethernet=yes
;DefaultTCPTTL=21
;KeepaliveTime=120
;KeepaliveData=yes
;NISDomain=my.nis.domain
;NISServers=192.168.110.1, 192.168.120.1, 192.168.130.1
;NTPServers=192.168.116.1, 192.168.126.1, 192.168.136.1
;NETBIOSNameSrv=192.168.5.1
;NETBIOSDistSrv=192.168.5.1
;NETBIOSNodeType=8
;NETBIOSScope=NETBIOS.COM
;XWindowFont=192.168.0.1
;XWindowManager=192.168.0.1
;NetwareIPDomain=NETWAREDOMAIN.COM
;NetWareIPOption=123,7,0,45,234,20,27,167,198,34,112,45
;NISDomainName=NISDOMAINNAME.COM
;NISServerAddr=192.168.0.1
```

```
;TFTPServerName=MyTFTPServer
BootFileName=cat4500es8-universalk9.SSA.03.09.00.PR4.9.152-5.0.9.PR4.bin
;BootFileOption=BootFileOption.ini
;HomeAgentAddrs=192.168.0.1
;SMTPServer=192.168.0.1
;POP3Server=192.168.0.1
;NNTPServer=192.168.0.1
;WWWServer=192.168.0.1
;FingerServer=192.168.0.1
;IRCServer=192.168.0.1
;StreetTalkServer=192.168.0.1
;STDAServer=192.168.0.1
;NDSServers=192.168.0.1
;NDSTreeName="myNDSTree"
;NDSContext=NewContext
;LDAP="ldap://192.168.1.1"
;AutoConfig=yes
;NameServiceSearch=23,0,235,4,2,0,236,7,94,34,87,4,127,254,23
;SubnetSelectionOption=255.255.255.240
#Option TFTPServerIPaddress is for phone use only, for PXEBoot use NextServer option
;TFTPServerIPaddress=192.168.4.1
;CallServerIPaddress=192.168.0.1
;DiscriminationString=""
;RemoteStatisticsServerIPAddress=192.168.50.1
;HTTPProxyPhone=192.168.51.1
;IPTelephone="MCIPADD=10.10.0.1,MCPORT=1719,TFTPSRVR=10.10.0.254,TFTPDIR=,VLANTEST=0"
#next few are sample flat option, (global mac boot options)
#option mac-version
;230=00:00:00:00
#option mac-nb-img
;234=ac:11:00:09:02:24:0a:4D:61:63:20:48:44:5F:4E:42:53:00:00:00:0:02:1b:53:68:61:72:65:64
:49:6D:61;:67:65:73:00:4E:65:74:42:6F:6F:74:20:48:44:2E:69:6D:67
#option mac-apps-img
;235="\opt\isv\boot\bootimage.bin"


#Following sections are Static Client DHCP entries/options
#If no IP is given, then that host will never be allooted any IP
#More option Names can be copied from DHCP-OPTIONS to clients.
#For BOOTP requests, only these options would be sent.
#For DHCP requests. Missing Options will be supplimented from
#first [DHCP-RANGE] options (if IP falls in any range), other
#options will be supplemented from [DHCP-OPTIONS].


[00:41:42:41:42:00]
#This is a client with MAC addr 00:41:42:41:42:00
IP=192.168.0.200
#No other options specified for this client
#For non BOOPT requests, Missing Options will be supplemented from first [RANGE_SET]
#options, if IP falls in any range. and other missing would be added from
[GLOBAL_OPTIONS].

[00:41:42:41:42:05]
#This is a client with MAC addr 00:41:42:41:42:05
IP=192.168.0.211
#DHCP will offer following hostname to this client
;HostName=TestHost
#For example, you can specify DNS Servers, Routers separately for this client
;DomainServer=10.5.6.90, 11.4.5.6
;Router=11.5.6.7, 4.6.7.34
;NETBIOSNodeType=8
#AddressTime can be different for this client
#specify 0 for infinity.AddressTime
```

```
;AddressTime=36000

[00:ff:a4:0e:ef:d5]
#this is an example for MacOsX network boot, client specific options
#for client having MAC addr 00:ff:a4:0e:ef:d5
IP=10.10.0.12
#you can omit the comments, these are for guidance only
#Next Server (TFTP Boot Server) and Boot File can be different for this client
;BootFileName=pxelinux.0
;BootFileSize=255
;RootPath="/"
;ExtensionFile="/linux/"
;NextServer=192.168.0.1
#option mac-nc-client-unknown
;220=00:00:00:00
#option mac-nc-client-id
;221=4D:61:63:20:4E:43:20:23:38
#option mac-username
;232="bootuser"
#option mac-password
;233="bootpassword"
#option mac-machine-name
;237=myComputer
#option mac-client-nb-img
;238="\opt\isv\boot\image.bin"

[00:ff:a4:0e:ef:99]
#This host has no IP
#This host will never get an
#IP, even from Dynamic Ranges
#You can disable a host from
#Getting an IP from this Server.
#using this kind of entries
```

Using NETCONF and RESTCONF

NETCONF uses a simple RPC-based (Remote Procedure Call) mechanism to facilitate communication between a client and a server. The client can be a script or an application running as part of a network manager. The server is typically a network device (switch or router).

NETCONF uses Secure Shell Version 2(SSHv2) as the transport layer across network devices and RESTCONF uses HTTP.

NETCONF and RESTCONF also support capability discovery and model downloads. Supported models are discovered using the ietf-netconf-monitoring model. Revision dates for each model are shown in the capabilities response. Data models are available for optional download from a device using the get-schema rpc. You can use these YANG models to understand or export the data model.

To use NETCONF and RESTCONF you must complete all the required tasks as per the Configuring Programmability, page 1-4 section. The following shows examples of the RPCs you can send and the kind of action that is performed.

Examples for NETCONF RPCs

Get the running-configuration of the switch by sending the following RPC:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter>
      <native xmlns="http://cisco.com/ns/yang/ned/ios"/>
    </filter>
  </get-config>
</rpc>
```

Change the description of an interface by sending the following RPC

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <native xmlns="http://cisco.com/ns/yang/ned/ios">
        <interface>
```

```
          <TenGigabitEthernet>
            <name>4/1</name>
            <description>to_distribution</description>
           </TenGigabitEthernet>
        </interface>
      </native>
    </config>
  </edit-config>
</rpc>
```

Remove the description from an interface by sending the following RPC

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <native xmlns="http://cisco.com/ns/yang/ned/ios">
        <interface>
          <TenGigabitEthernet>
            <name>4/1</name>
            <description xc:operation="delete"/>
           </TenGigabitEthernet>
        </interface>
      </native>
    </config>
  </edit-config>
</rpc>
```

Examples for RESTCONF RPCs:

Get the TFTP source interface by sending the following RPC:

```
GET http://10.106.30.33:80/restconf/api/running/native/ip/tftp/source-interface
```

Configure the TFTP source interface by sending the following RPC:

```
PATCH
http://10.106.30.33:80/restconf/api/running/native/ip/tftp/source-interface/GigabitEth
ernet
payload = "{\n    \"GigabitEthernet\": \"2/2\"\n}"
```

Enter a HTTP delete request by sending the following RPC:

```
DELETE http://10.106.30.33:55080/api/running/native/ip/tftp/source-interface/
```

Note For the HTTP delete request do not use:
`http://10.106.30.33:80/restconf/api/running/native/ip/tftp/source-interface/`