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.

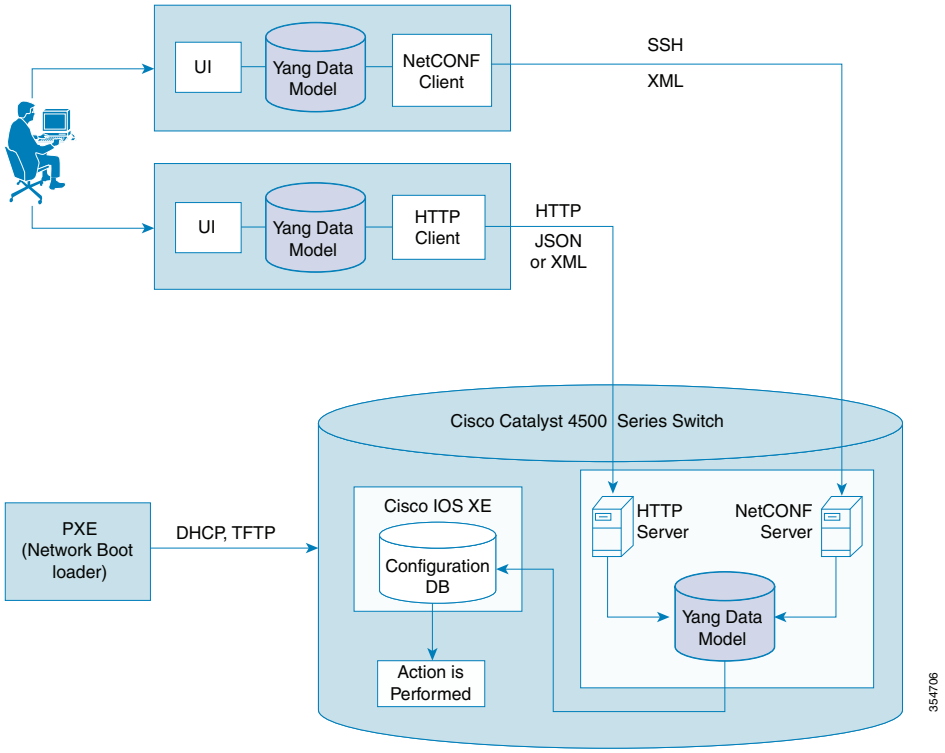
These components, enable you to set up what is required for Programmability:

- 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.