System Setup and Software Installation Guide for Cisco NCS 1014, IOS XR Releases 24.x.x

First Published: 2024-09-30

Last Modified: 2025-02-17

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2025 Cisco Systems, Inc. All rights reserved.

CONTENTS

CHAPTER 1 **Bring-up Cisco NCS 1014** 1

- Boot Using Zero Touch Provisioning 1
  - Fresh Boot Using DHCP 2
  - Build your Configuration File 4
  - Configure ZTP BootScript 8
  - Invoke ZTP Manually through CLI 10
  - Invoke ZTP Through Reload 11
  - ZTP Logging 13
  - Generate Tech Support Information for ZTP 14
- Configure Management Interface 14
- Boot NCS 1014 16
- Boot NCS 1014 Using USB Drive 16
- iPXE 18
  - Setup DHCP Server 18
  - iPXE boot process 20
- Bring-Up Line Card 22
- Configure NTP Server 24
  - Understand NTP 24
  - Synchronize Clock with NTP Server 25
  - Verify the Status of the External Reference Clock 26
  - NTP Troubleshooting Information 27

CHAPTER 2 **Perform Preliminary Checks** 29

- Inventory Support in NCS 1014 29
  - Verify Inventory 30
- Verify Status of Hardware Components 36

Verify Software Version 38
Verify Environmental Parameters 39
Verify Management Interface Status 42
Verify Firmware Version 44
Verify Alarms 46
Verify Context 48
Verify Core Files 53

CHAPTER 3 **Perform System Upgrade and Install Feature Packages** 55
Upgrade Software 55
View Supported Software Upgrade or Downgrade Versions 58
Software Upgrade and Downgrade Matrix 61
Install Packages and RPMs 62
Upgrade FPD 66
Verify if an FPD Upgrade is Required 70
Manual FPD Upgrade 72
Automatic FPD upgrade 73

CHAPTER 4 **Disaster Recovery** 75
Overview 75
CPU Replacement Considerations 75
How the Node Recovers After a Chassis SSD Replacement 76
Health Check of Backup ISO Image 76

CHAPTER 5 **Connection Verification** 77
Power Data Reading 77
Connection Verification 77
CCMD-16 Connection Verification with OLT 78
Verify Connection for CCMD-16 Line Card 78

CHAPTER 6 **System Health Check** 81
System Health Check 81
Enable Health Check 82
Change Health Check Refresh Time 83

View Status of All Metrics 83
Change Threshold Value for a Metric 86
View Health Status of Individual Metric 86
Disable Health Check 88

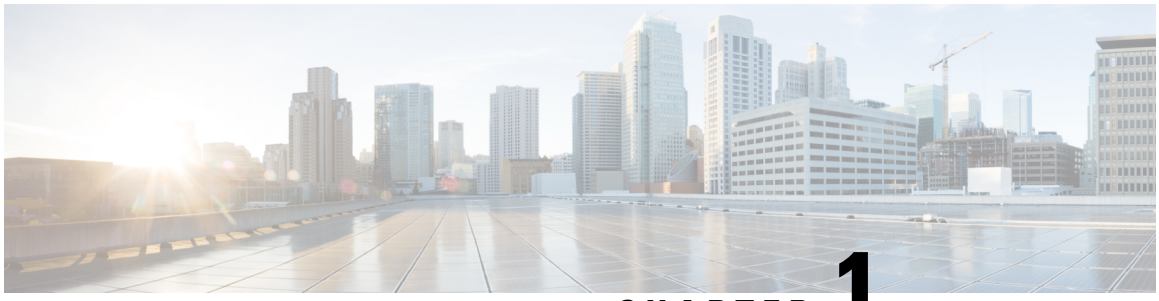
CHAPTER 7 **Configure AAA** 89
Deprecation of Type 7 password and Type 5 secret 89
About TACACS+ 94
Configure TACACS+ Server 95
Configure TACACS+ Server Groups 95
About RADIUS 97
Configure RADIUS Server Groups 97

CHAPTER 8 **Configure Access Control List** 101
Understand Access Control Lists 102
How an ACL Works 103
Apply ACLs 105
Configure an Ingress IPv4 ACL on Management Ethernet Interface 105
Configure an Egress IPv4 ACL on the Management Ethernet Interface 106
Configure an Ingress IPv6 ACL on the Management Ethernet Interface 108
Configure an Egress IPv6 ACL on the Management Ethernet Interface 109
Configure Extended Access Lists 110
Modify ACLs 111

CHAPTER 9 **Smart Licensing** 113
Understanding Smart Licensing 113
Create an ID Token 115
Smart Licensing Transport Modes 116
Configure Callhome 116
Configure Smart 117
Configure CSLU 118
Configure Offline 119
Reserve Specific Licenses for NCS 1014 120
Smart Licensing for QXP Line Card 121

CHAPTER 10 Automated File Management 123

Automated File Management System 123



CHAPTER 1

Bring-up Cisco NCS 1014

After installing the hardware, boot the Cisco NCS 1014 system. You can connect to the XR console port and power on the system. NCS 1014 completes the boot process using the pre-installed operating system (OS) image. If no image is available, NCS 1014 can be booted using the iPXE boot or an external bootable USB drive.

After booting, create the root username and password, and then use it to log on to the XR console.

Note The output of the examples in the procedures is not from the latest software release. The output will change for any explicit references to the current release.

- Boot Using Zero Touch Provisioning, on page 1
- Configure Management Interface, on page 14
- Boot NCS 1014, on page 16
- Boot NCS 1014 Using USB Drive, on page 16
- iPXE, on page 18
- Bring-Up Line Card, on page 22
- Configure NTP Server, on page 24

Boot Using Zero Touch Provisioning

ZTP allows you to provision the network device with day 0 configurations and supports both management ports and data ports.

ZTP provides multiple options such as:

- Automatically apply specific configuration in a large-scale environment.
- Download and install specific IOS XR image.
- Install specific application package or third-party applications automatically.
- Deploy containers without manual intervention.
- Upgrade or downgrade software versions effortlessly on thousands of network devices at a time.

Benefits Using ZTP

ZTP helps you manage large-scale service provider infrastructures effortlessly. Following are the added benefits of using ZTP:

- ZTP helps you to remotely provision a router anywhere in the network. This eliminates the need for an expert to deploy network devices and reduces IT cost.
- Automated provisioning using ZTP removes delay, increases accuracy, provides better customer experience and is cost-effective.

  By automating repeated tasks, ZTP allows network administrators to concentrate on more important stuff.
- ZTP process helps you to quickly restore service. Rather than troubleshooting an issue on hand, you can reset a system to a well-known working status.

Prerequisites:

ZTP does not execute, if a username is already configured in the system.

ZTP is initiated in one of the following ways:

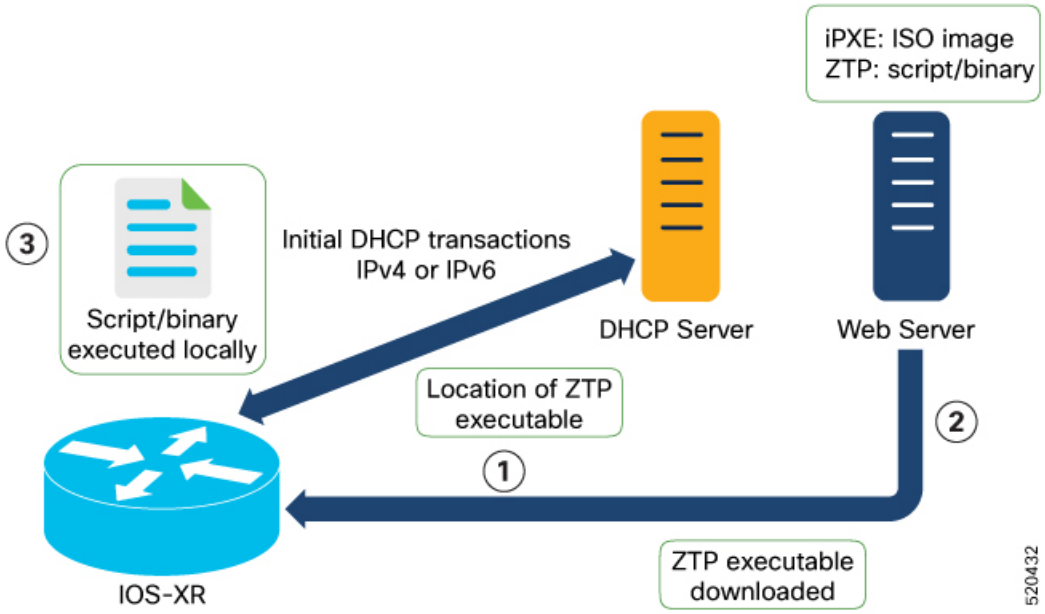
- **Automated Fresh Boot**: When you boot the device, the ZTP process initiates automatically if the device does not have a prior configuration. During the process, the router receives the details of the configuration file from the DHCP server. Use this method for devices that has no pre-loaded configuration. For more information, see Fresh Boot Using DHCP, on page 2.

  You must define the configuration file or the bootscript that is downloaded from the DHCP server.
- **Configuration File**: The first line of the file must contain **!! IOS XR configuration"**, to process the file as a configuration. If you are trying to bring up ten new nodes, you have to define ten configuration files. See Build your Configuration File, on page 4.
- **ZTP Bootscript**: Define the script to be executed on every boot. See Configure ZTP BootScript, on page 8 .
- **Manual Invocation using CLI:** Use this method when you want to forcefully initiate ZTP on a fully configured device using CLI. See Invoke ZTP Manually through CLI, on page 10.
- **Invocation using Reload Command:** Use this method when you want to forcefully initiate ZTP on a fully configured device using the **reload** command. See Invoke ZTP Through Reload, on page 11.

Fresh Boot Using DHCP

The ZTP process initiates when you boot the network device with an IOS XR image. The ZTP process starts only on a device without prior configuration.

This figure depicts the high-level workflow of the ZTP process:



1. ZTP sends DHCP request to fetch the ZTP configuration file or user script. To help the Bootstrap server uniquely identify the device, ZTP sends below DHCP option.
   - DHCP(v4/v6) client-id=Serial Number
   - DHCPv4 option 124: Vendor, Platform, Serial-Number
   - DHCPv6 option 16: Vendor, Platform, Serial-Number

   The following is the default sequential flow of the ZTP process defined in the configuration file. You can modify this sequence in the configuration file, if required.
   - ZTP sends IPv4 DHCP request first on all the management ports. If the request fails, then ZTP sends IPv6 DHCP request on all the management ports.
   - ZTP sends IPv4 DHCP request first on all the data ports. If the request fails, then ZTP sends IPv6 DHCP request on all the data ports.

2. DHCP server identifies the device and responds with DHCP response.

   DHCP server should be configured to respond with DHCP response and supply script/config location with one of the following DHCP options:
   - DHCPv4 using BOOTP filename.
   - DHCPv4 using Option 67 (bootfile-name).
   - DHCPv6 using Option 59 (OPT_BOOTFILE_URL).

3. The network device downloads the file from the web server using the URL location provided in the DHCP response.
4. The device receives a configuration file or script file from the HTTP server.

Note

- If the downloaded file content starts with *!! IOS XR*, it is considered as a configuration file.
- If the downloaded file content starts with *#! /bin/bash*, *#! /bin/sh*, or *#!/usr/bin/python*, it is considered as a script file.

5. The device applies the configuration file or executes the script or binary in the default bash shell.
6. The Network device is now up and running.

Build your Configuration File

Based on the business need, you can use a configuration or script file to initiate the ZTP process.

The configuration file content starts with *!! IOS XR*.

The following is the sample configuration file. You can automate all the configurations. For more information on creating ZTP configuration file, refer ZTP Configuration Files Creation.

```
!! Building configuration
!! IOS XR Configuration 7.11.1.35I
!! Last configuration change at Fri Sep 15 17:18:53 2023 by cisco
!
hostname IOS_P2B_FLT
logging console debugging
username cisco
 group root-lr
 group cisco-support
 secret 10
$6$4gjnzvvwDCz1z...$bovO.6uRYD9qsujiw6DNjTx6bngeDIVMvXxVbReal6bpd0SRo5qyfHk5S4D23r9hjntYtXnyQWNcrgbK0USB20
!
grpc
 port 57400
!
line template vty
 timestamp disable
 exec-timeout 0 0
!
line template test
 exec-timeout 0 0
!
line console
 timeout login response 30
 timestamp
 exec-timeout 0 0
 width 0
 length 0
!
line default
 timestamp disable
 exec-timeout 0 0
 length 0
 absolute-timeout 0
 session-timeout 0
!
vty-pool default 0 10 line-template default
fpd auto-upgrade enable
ntp
```

```
 max-associations 99
!
call-home
 service active
 contact smart-licensing
 profile CiscoTAC-1
  active
  destination transport-method email disable
  destination transport-method http
 !
!
netconf-yang agent
 ssh
!
hw-module location 0/1/NXR0
 mxponder-slice 0
  trunk-rate 600G
  client-rate 100GE
 !
!
hw-module location 0/2/NXR0
 mxponder-slice 0
  trunk-rate 800G
  client-port-rate 1 client-type 400GE
 !
!
interface MgmtEth0/RP0/CPU0/0
 description mgmt0
 ipv4 address 10.105.57.64 255.255.255.0
!
interface MgmtEth0/RP0/CPU0/1
 ipv4 address 10.127.60.44 255.255.255.0
 ipv6 enable
!
controller Optics0/0/0/0
 description optics0/0/0/0
 pm 30-sec optics threshold opt min 2
 fastpoll enable
 perf-mon enable
!
controller Optics0/0/0/1
 description optics0/0/0/1
 fastpoll enable
!
controller Optics0/0/0/2
 description optics0/0/0/2
 perf-mon enable
!
controller Optics0/0/0/3
 description optics0/0/0/3
!
controller Optics0/0/0/4
 description optics0/0/0/4
!
controller Optics0/0/0/5
 description optics0/0/0/5
!
controller Optics0/0/0/6
 description optics0/0/0/6
!
controller Optics0/0/0/7
 description optics0/0/0/7
!
controller Optics0/0/0/8
```

```
 description optics0/0/0/8
!
controller Optics0/0/0/9
 description optics0/0/0/9
 pm 15-min optics report opt max-tca enable
 pm 15-min optics threshold opt-dbm max -200
 pm 30-sec optics report opr min-tca enable
 pm 30-sec optics report opt max-tca enable
 pm 30-sec optics threshold opr-dbm min 500
 pm 30-sec optics threshold opt-dbm max -210
!
controller Optics0/0/0/10
 description optics0/0/0/10
!
controller Optics0/0/0/11
 description optics0/0/0/11
!
controller Optics0/0/0/12
 description optics0/0/0/12
!
controller Optics0/0/0/13
 description optics0/0/0/13
!
controller Optics0/1/0/0
 description optics0/1/0/0
 pm 15-min optics report opr min-tca enable
 pm 15-min optics threshold opr-dbm min 200
 pm 30-sec optics report opr min-tca enable
 pm 30-sec optics threshold opr-dbm min 200
 fastpoll enable
!
controller Optics0/1/0/1
 description optics0/1/0/1
!
controller Optics0/1/0/2
 description optics0/1/0/2
!
controller Optics0/1/0/3
 description optics0/1/0/3
!
controller Optics0/1/0/4
 description optics0/1/0/4
!
controller Optics0/1/0/5
 description optics0/1/0/5
!
controller Optics0/1/0/6
 description optics0/1/0/6
!
controller Optics0/1/0/7
 description optics0/1/0/7
!
controller Optics0/1/0/8
 description optics0/1/0/8
!
controller Optics0/1/0/9
 description optics0/1/0/9
!
controller Optics0/1/0/10
 description optics0/1/0/10
!
controller Optics0/1/0/11
 description optics0/1/0/11
!
```

```
controller Optics0/1/0/12
 description optics0/1/0/12
!
controller Optics0/1/0/13
 description optics0/1/0/13
!
controller Optics0/2/0/0
 description optics0/2/0/0
 transmit-power -25
 dwdm-carrier 100MHz-grid frequency 1923500
 rx-low-threshold -120
 rx-high-threshold 40
 tx-low-threshold -101
 tx-high-threshold 40
!
controller Optics0/2/0/1
 description optics0/2/0/1
!
controller Optics0/2/0/2
 description optics0/2/0/2
!
controller Optics0/2/0/3
 description optics0/2/0/3
!
controller Optics0/2/0/4
 description optics0/2/0/4
!
controller Optics0/2/0/5
 description optics0/2/0/5
!
controller Optics0/2/0/6
 description optics0/2/0/6
!
controller Optics0/2/0/7
 description optics0/2/0/7
!
controller Optics0/3/0/0
 description optics0/3/0/0
!
controller Optics0/3/0/1
 description optics0/3/0/1
!
controller Optics0/3/0/2
 description optics0/3/0/2
 pm 30-sec optics report opr min-tca enable
 pm 30-sec optics threshold opr-dbm min 200
!
controller Optics0/3/0/3
 description optics0/3/0/3
!
controller Optics0/3/0/4
 description optics0/3/0/4
!
controller Optics0/3/0/5
 description optics0/3/0/5
!
controller Optics0/3/0/6
 description optics0/3/0/6
!
controller Optics0/3/0/7
 description optics0/3/0/7
!
controller Optics0/3/0/8
 description optics0/3/0/8
```

```
!
controller Optics0/3/0/9
 description optics0/3/0/9
!
controller Optics0/3/0/10
 description optics0/3/0/10
!
controller Optics0/3/0/11
 description optics0/3/0/11
!
controller Optics0/3/0/12
 description optics0/3/0/12
!
controller Optics0/3/0/13
 description optics0/3/0/13
!
interface PTP0/RP0/CPU0/0
 shutdown
!
interface PTP0/RP0/CPU0/1
 shutdown
!
router static
 address-family ipv4 unicast
  0.0.0.0/0 10.105.57.1
  0.0.0.0/0 10.127.60.1
 !
!
snmp-server traps sensor
snmp-server traps fru-ctrl
netconf agent tty
!
lldp
!
ains-soak hours 47 minutes 59
ssh timeout 120
ssh server rate-limit 600
ssh server session-limit 100
ssh server v2
ssh server vrf default
ssh server netconf vrf default
end
```

Configure ZTP BootScript

ZTP downloads and executes the script files. These script files include a programmatic approach to complete a task. For example, scripts created using IOS XR commands to perform patch upgrades. The first line of the file must contain **#! /bin/bash** or **#! /bin/sh** for ZTP to process the file as script.

You can either use the ZTP bash script or the ZTP configuration file.

To manually execute a script during every boot, use the following configuration:

```
Router#configure
Router(config)#ztp bootscript /disk0:/myscript
Router(config)#commit
```

To ensure that we have connectivity in the third-party namespace for applications to use, the above configuration waits for the first data plane interface to be configured and wait an extra minute for the management interface to be configured with an IP address. If the delay is not desired, use:

```
Router#configure
Router(config)#ztp bootscript preip /disk0:/myscript
Router(config)#commit
```

Note When the above command is first configured, you will be prompted if you wish to invoke it now. The prompt helps with testing.

This is the example content of **/disk0:/myscript**:

```
host ncs1010_P1B_DT_08_ETH0 {
#hardware ethernet 68:9e:0b:b8:6f:5c ;
option dhcp-client-identifier "FCB2437B05N" ;
if exists user-class and option user-class = "iPXE" {
filename "http://10.33.0.51/P1B_DT_08/ncs1010-x64.iso";
} else {
filename "http://10.33.0.51/P1B_DT_08/startup.cfg";
}
fixed-address 10.33.0.19;
}
```

The following is the sample content of the ZTP bash script.

```
#! /bin/bash
#
# NCS1014 Demo Sample
# ZTP installation of config and day-0 SMU's
#
source ztp_helper

wget http://downloads.sourceforge.net/project/yourcode/application.tgz
#install the downloaded application.tgz

#Run XR CLI's from the script
`xrcmd "show version"`
```

The following is the sample content of the ZTP configuration file.

```
Tue May 4 18:08:59.544 UTC
Building configuration...
IOS XR Configuration 7.11.1.35I
!! Last configuration change at Fri Sep 15 17:18:53 2023 by cisco
!
line console
exec-timeout 0 0
!
line default
exec-timeout 0 0
session-timeout 0
!
vty-pool default 0 20
alias alarms show alarms brief system active
interface MgmtEth0/RP0/CPU0/0
ipv4 address dhcp
no shut
!
interface MgmtEth0/RP0/CPU0/1
description noshut-interface-ztp
ipv4 address 10.127.60.160 255.255.255.0
no shut
!
```

```
interface PTP0/RP0/CPU0/0
description noshut-interface-ztp
no shut
!
interface PTP0/RP0/CPU0/1
description noshut-interface-ztp
no shut
end
```

Invoke ZTP Manually through CLI

Manual ZTP can be invoked through CLI commands. This manual way helps you to provision the router in stages. Ideal for testing out ZTP configuration without a reboot. If you want to invoke a ZTP on an interface (data ports or management ports), you need not bring up and configure the interface first. You can execute the **ztp initiate** command, even if the interface is down, so that the ZTP script brings it up and invoke *dhclient*. ZTP can run on all interfaces irrespective of whether the interfaces are up or not.

Use the **ztp initiate**, **ztp terminate**, and **ztp clean** commands to force ZTP to run on more interfaces.

- **ztp initiate**—Invokes a new ZTP DHCP session. Logs can be found in the */disk0:/ztp/ztp.log* location.
- **ztp terminate**—Terminates current ZTP sessions.
- **ztp clean**—Removes only the ZTP state files.

The log file *ztp.log* is saved in the */var/log/ztp.log* folder, and a copy of log file is available in the */disk0:/ztp/ztp.log* location using a soft link. However, executing **ztp clean** clears files saved on disk and not from the **/var/log/ztp.log** folder where current ZTP logs are saved. To get a log from current ZTP run, you must manually remove the ZTP log file from **/var/log/ztp.log**.

SUMMARY STEPS

1. (optional) **ztp clean**
2. **ztp initiate**
3. (Optional) **ztp terminate**

DETAILED STEPS

Procedure

Step 1 (optional) **ztp clean**

Example:

```
RP/0/RP0/CPU0:ios#ztp clean
Fri Sep 15 17:12:33.477 IST
This would remove all ZTP temporary files.
Would you like to proceed? [no]: yes
All ZTP operation files have been removed.
ZTP logs are present in /var/log/ztp*.log for logrotate.
Please remove manually if needed.
If you now wish ZTP to run again from boot, do 'conf t/commit replace' followed by reload.
RP/0/RP0/CPU0:ios#
```

Removes all the ZTP logs and saved settings.

Step 2 **ztp initiate**

Example:

```
RP/0/RP0/CPU0:ios#ztp initiate
Fri Sep 15 17:13:28.580 IST
Inititaing ZTP may change your configuration.
Interfaces might be brought up if they are in shutdown state
Would you like to proceed? [no]: yes
ZTP will now run in the background.
Please use "show logging" or look at /var/log/ztp.log to check progress.
RP/0/RP0/CPU0:ios#
```

Reboots the Cisco NCS 1014 system.

Use the **show logging** command or see the */var/log/ztp.log* to check progress.

Step 3 (Optional) **ztp terminate**

Example:

```
RP/0/RP0/CPU0:ios#ztp terminate
Fri Sep 15 17:15:04.592 IST
This would terminate active ZTP session if any (this may leave your system in a partially configured
 state)
Would you like to proceed? [no]: yes
Terminating ZTP
RP/0/RP0/CPU0:ios#
```

Terminates the ZTP process.

Invoke ZTP Through Reload

The ZTP process can be automatically invoked using the **reload** command.

SUMMARY STEPS

1. **configure**
2. **commit replace**
3. **ztp clean**
4. **reload**

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:P2B_DT_02#configure
```

Enters the configuration mode.

Step 2 **commit replace**

Example:

```
RP/0/RP0/CPU0:ios(config)#commit replace
Fri Sep 15 11:47:31.746 UTC

This commit will replace or remove the entire running configuration. This
operation can be service affecting.
Do you wish to proceed? [no]: yes
RP/0/RP0/CPU0:ios(config)#
```

Removes the entire running configuration.

Step 3 **ztp clean**

Example:

```
RP/0/RP0/CPU0:ios#ztp clean
Fri Sep 15 11:48:13.669 UTC
This would remove all ZTP temporary files.
Would you like to proceed? [no]: yes
All ZTP operation files have been removed.
ZTP logs are present in /var/log/ztp*.log for logrotate.
Please remove manually if needed.
If you now wish ZTP to run again from boot, do 'conf t/commit replace' followed by reload.
RP/0/RP0/CPU0:ios#
```

Removes all the ZTP logs and saved settings.

Step 4 **reload**

Example:

```
RP/0/RP0/CPU0:ios#reload
Fri Apr 29 06:50:12.312 UTC
Proceed with reload? [confirm]

RP/0/RP0/CPU0:ios#
Preparing system for backup. This may take a few minutes especially for large configurations.
 Status report: node0_RP0_CPU0: BACKUP INPROGRESS
 Status report: node0_RP0_CPU0: BACKUP HAS COMPLETED SUCCESSFULLY
[Done]
```

After the node comes up, you can see that the ZTP is initiated and the configuration has been restored successfully.

```
RP/0/RP0/CPU0:Sep 25 17:29:19.875 UTC: pyztp2[325]: %INFRA-ZTP-4-START : ZTP has started. Interfaces
 might be brought up if they are shutdown
RP/0/RP0/CPU0:Sep 25 17:30:33.286 UTC: pyztp2[325]: %INFRA-ZTP-6-DISCOVERY_COMPLETED : Discovery
successful on MgmtDhcp4Fetcher. Will proceed with fetching.
RP/0/RP0/CPU0:Sep 25 17:30:47.362 UTC: pyztp2[325]: %INFRA-ZTP-6-FETCHING_COMPLETED : Provisioning
file fetched successfully
RP/0/RP0/CPU0:Sep 25 17:31:30.889 UTC: pyztp2[325]: %INFRA-ZTP-4-CONFIG_INITIATED : ZTP has initiated
 config load and commit operations
RP/0/RP0/CPU0:Sep 25 17:32:36.100 UTC: pyztp2[325]: %INFRA-ZTP-4-CONFIG_FINISHED : ZTP has finished
 config load and commit operations
RP/0/RP0/CPU0:Sep 25 17:32:41.059 UTC: pyztp2[325]: %INFRA-ZTP-6-CFG_TAMP_SAVE_HASH : Config hash
saved after ztp Config is: (643013d9a43a3d2576012a24eb9745a8f960480d0053d06ed81146cb3c3d54c5).
RP/0/RP0/CPU0:Sep 25 17:32:44.089 UTC: pyztp2[325]: %INFRA-ZTP-4-PROVISIONING_COMPLETED : ZTP has
successfully completed the provisioning
RP/0/RP0/CPU0:Sep 25 17:32:52.909 UTC: pyztp2[325]: %INFRA-ZTP-4-EXITED : ZTP exited
User Access Verification

Username: cisco
Password:
ios con0/RP0/CPU0 is now available
```

Reboots the Cisco NCS 1014 system.

ZTP Logging

ZTP logs its operation on the flash file system in the */disk0:/ztp/* directory. ZTP logs all the transactions with the DHCP server and all the state transitions.

The following example displays the execution of a simple configuration script downloaded from a data interface using the command **ztp initiate interface MgmtEth 0/RP0/CPU0/0 verbose**. This script unshuts all the interfaces of the system and configures a load interval of 30 seconds on all of them.

```
2023-09-25 17:37:31,693  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: Sending standby sync message. done = False
2023-09-25 17:37:31,716  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: [privileged] getting engine status. done = False
2023-09-25 17:37:31,717  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: Fetching provisioning data. done = False
2023-09-25 17:37:31,718  28136 [Engine       ] INF: ZAdmin, current state:active: state tag
 changed to fetch
2023-09-25 17:37:31,721  28136 [Xr           ] INF: Downloading the file to /tmp/ztp.script
2023-09-25 17:37:31,948  28136 [ReportBootz  ] INF: User script downloaded successfully.
Provisioning in progress.
2023-09-25 17:37:31,950  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: Config device work for ZAdmin. done = False
2023-09-25 17:37:31,951  28136 [ZtpHelpers   ] DEB: Executing: source /pkg/bin/ztp_helper.sh
 && echo -ne  | xrcmd "show version"
2023-09-25 17:37:32,956  28136 [ZAdmin       ] DEB: Proceeding to provision the router
2023-09-25 17:37:32,958  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: ZAdmin: Apply configuration. done = False
2023-09-25 17:37:32,959  28136 [Engine       ] INF: ZAdmin, current state:active: state tag
 changed to provision
2023-09-25 17:37:32,975  28136 [Env          ] DEB: No MTU configs detected
2023-09-25 17:37:32,977  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: ZAdmin: Apply configuration. done = False
2023-09-25 17:37:33,021  28136 [Xr           ] DEB: Will apply the following config:
/disk0:/ztp/customer/config.candidate
2023-09-25 17:37:33,022  28136 [Xr           ] INF: Applying user configurations
2023-09-25 17:37:33,023  28136 [Configuration] INF: Provisioning via config replace
2023-09-25 17:38:14,445  28136 [Configuration] INF: Configuration has been applied
2023-09-25 17:38:14,447  28136 [Env          ] DEB: cfg::createRefOnConfigCommit: called
2023-09-25 17:38:15,778  28136 [Env          ] DEB: cfg:: Generating hash for File name:
/disk0:/ztp/customer/config.inithash_tmp
2023-09-25 17:38:15,780  28136 [Env          ] DEB: cfg::_generateCfgAndSaveHash:: HASH :
643013d9a43a3d2576012a24eb9745a8f960480d0053d06ed81146cb3c3d54c5, type : 1
2023-09-25 17:38:17,743  28136 [Env          ] DEB: cfg::getRefOnConfigCommit: called
2023-09-25 17:38:17,818  28136 [Env         ] DEB: cfg::getRefOnConfigCommit :: ret : data
 : 643013d9a43a3d2576012a24eb9745a8f960480d0053d06ed81146cb3c3d54c5, len: 64
2023-09-25 17:38:17,819  28136 [Env         ] INF: Env::getConfigRefHashOnCommit: get data
 from tam : success:b'643013d9a43a3d2576012a24eb9745a8f960480d0053d06ed81146cb3c3d54c5'
2023-09-25 17:38:17,821  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: Sending standby sync message. done = False
2023-09-25 17:38:17,836  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: [privileged] getting engine status. done = False
2023-09-25 17:38:17,837  28136 [Engine      ] DEB: ZAdmin, current state:active. Processing
 work: ZAdmin: Execute post-configuration script. done = False
2023-09-25 17:38:17,873  28136 [Env         ] INF: Env::cleanup, success:True, exiting:False
2023-09-25 17:38:17,876  28136 [ZtpHelpers   ] DEB: Executing: source /pkg/bin/ztp_helper.sh
 && echo -ne  | xrcmd "show running-config"
2023-09-25 17:38:19,582  28136 [Env          ] INF: Executing command ip netns exec
vrf-default /sbin/dhclient -4 -cf /etc/dhcp/dhclient.conf.ztp -lf
```

```
/var/lib/dhcp/dhclient.leases.ztp -sf /etc/dhcp/dhclient-script.ztp2 -r Mg0_RP0_CPU0_0 to
release IP
2023-09-25 17:38:20,695  28136 [Xr          ] INF: Removing linux route with ip 10.105.57.107
2023-09-25 17:38:20,731  28136 [Xr          ] INF: Failed to remove default route to to_xr
 via 10.105.57.107 with error: Error: RTNETLINK answers: No such process encountered while
 executing command: ip netns exec vrf-default ip route del default dev to_xr src 10.105.57.107
 metric 512
2023-09-25 17:38:20,736  28136 [Engine       ] INF: ZAdmin, current state:active, exit
code:success
2023-09-25 17:38:20,737  28136 [Engine       ] INF: ZAdmin, current state:final, exit
code:success: state changed to final
2023-09-25 17:38:22,846  28136 [Engine       ] DEB: ZAdmin, current state:final, exit
code:success. Processing work: Sending standby sync message. done = False
2023-09-25 17:38:22,847  28136 [Engine       ] WAR: ZAdmin, current state:final, exit
code:success: work is ignored: work=<desc='Sending standby sync message' done=False
priv=False>
2023-09-25 17:38:22,848  28136 [Engine       ] DEB: ZAdmin, current state:final, exit
code:success. Processing work: [privileged] getting engine status. done = False
2023-09-25 17:38:27,853  28136 [__main__     ] DEB: Moved to final state
2023-09-25 17:38:27,854  28136 [__main__     ] DEB: ZTP completed successfully
2023-09-25 17:38:27,855  28136 [__main__     ] INF: Exiting SUCCESSFULLY
```

Generate Tech Support Information for ZTP

When you have a problem that you cannot resolve in the ztp process, contact the Cisco Systems technical support representative. To analyze a problem, your technical support representative needs certain information about the situation and the symptoms that you are experiencing. To speed up the resolution, collect the necessary data before you contact your representative.

Use the **show tech-support ztp** command to collect all debugging information of ztp process.

Example:

```
RP/0/RP0/CPU0:ios#show tech-support ztp
Thu Jul 28 08:33:27.531 UTC
++ Show tech start time: 2022-Jul-28.083327.UTC ++
Thu Jul 28 08:33:28 UTC 2022 Waiting for gathering to complete
..
Thu Jul 28 08:33:34 UTC 2022 Compressing show tech output
Show tech output available at 0/RP0/CPU0 :
/harddisk:/showtech/showtech-R1-ZTP-2022-Jul-28.083327.UTC.tgz
++ Show tech end time: 2022-Jul-28.083334.UTC ++
RP/0/RP0/CPU0:ios#
```

In the above example, the tech support information is saved as .tgz file in the specified location. This information can be shared with the Cisco Technical Support representatives for troubleshooting the ztp process.

Configure Management Interface

The management interface can be used for system management and remote communication. To use the management interface for system management, you must configure an IP address and subnet mask. To use the management interface for remote communication, you must configure a static route.

Before you begin

- Consult your network administrator to procure IP addresses and a subnet mask for the management interface.

- Ensure that the management interface is connected to the management network.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:ios#configure
```

Enters IOS XR configuration mode.

Step 2 **interface mgmtEth** *rack/slot/instance/port*

Example:

```
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/0
```

Enters interface configuration mode for the management interface.

Step 3 **ipv4 address** *ipv4-address subnet-mask*

Example:

```
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 10.1.1.1 255.0.0.0
```

Assigns an IP address and a subnet mask to the management interface.

Step 4 **no shutdown**

Example:

```
RP/0/RP0/CPU0:ios(config-if)#no shutdown
```

Places the management interface in an "up" state.

Step 5 **exit**

Example:

```
RP/0/RP0/CPU0:ios(config-if)#exit
```

Exits the management interface configuration mode.

Step 6 **router static address-family ipv4 unicast** *0.0.0.0/0 default-gateway*

Example:

```
RP/0/RP0/CPU0:ios(config)#router static address-family ipv4 unicast 0.0.0.0/0 10.25.0.1
```

Specifies the IP address of the default gateway to configure a static route. This IP address must be used for communication with devices on other networks.

Step 7 Use the **commit** or **end** command.

commit-Saves the configuration changes and remains within the configuration session.

end-Prompts user to take one of these actions:

- **Yes**-Saves configuration changes and exits the configuration session.
- **No**-Exits the configuration session without committing the configuration changes.

- **Cancel**-Remains in the configuration session without committing the configuration changes.

Boot NCS 1014

Use the console port to connect to NCS 1014. By default, the console port connects to the XR mode. If necessary, you can establish subsequent connections through the management port, after it is configured.

Procedure

Step 1 Connect a terminal to the console port of the RP.

Step 2 Start the terminal emulation program on your workstation.

The console settings are 115200 bps for NCS1K14-CNTLR-K9, 9600 bps for NCS1K14-CTLR-B-K9, 8 data bits, 1 stop bit and no parity.

Step 3 Power on the NCS 1014.

To power on the shelves, install the AC or DC power supplies and cables. As NCS 1014 boots up, you can view the boot process details at the console of the terminal emulation program.

Step 4 Press **Enter**.

The boot process is complete when the system prompts you to enter the root-system username. If the prompt does not appear, wait for a while to give NCS 1014 more time to complete the initial boot procedure; then press **Enter**.

If the boot process fails, it may be because the preinstalled image on the NCS 1014 is corrupt. In this case, you can boot NCS 1014 using an external bootable USB drive.

Boot NCS 1014 Using USB Drive

The bootable USB drive is used to reimage NCS 1014 for system upgrade or to boot the NCS 1014 in case of boot failure. A bootable USB drive is created by copying a compressed boot file into a USB drive. The USB drive becomes bootable after the contents of the compressed file are extracted.

You can complete this task using the Windows, Linux, or MAC operating systems available on your local machine. The exact operation to be performed for each generic step that is outlined here depends on the operating system in use.

Use this task to boot the NCS 1014 using the USB drive.

Before you begin

- You need a USB drive with a storage capacity of at least 4 GB.
- The USB drive should have a single partition.

- Copy the compressed boot file from the software download page at Cisco.com to your local machine. The filename for the compressed boot file is in the format *ncs1010-x64-usb-(release_number).zip*.

Procedure

Step 1 Connect the USB drive to your local machine and format it with the FAT32 file system.

Step 2 Copy the compressed boot file to the USB drive.

Step 3 Verify that the copy operation is successful. To verify, compare the file size at source and destination. Also, verify the MD5 checksum value.

Step 4 Extract the content of the compressed boot file by unzipping it in the USB drive. This makes the USB drive a bootable drive.

Note
You must extract the contents of the zipped file ("EFI" and "boot" directories) directly in the root folder of the USB drive. If the unzipping application places the extracted files in a new folder, move the "EFI" and "boot" directories to the root folder of the USB drive.

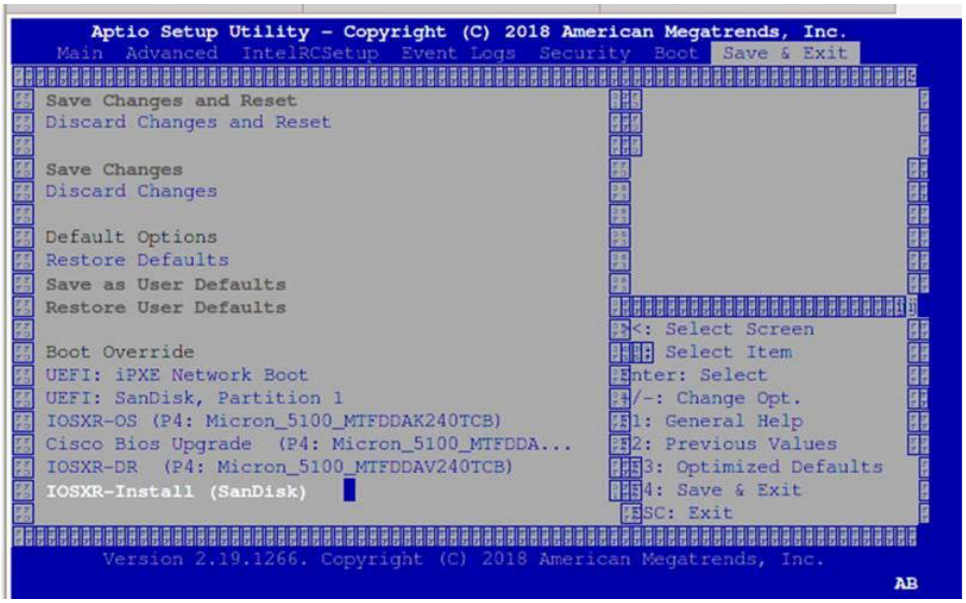
Step 5 Insert the USB drive in one of the USB ports of NCS 1014.

Step 6 Reboot NCS 1014 using power cycle or console.

Note
Use the **reload bootmedia usb noprompt** command to boot the NCS1010 from the USB. If you are using the **reload bootmedia usb noprompt** command, then you can skip the remaining steps.

Step 7 Press **Esc** to enter BIOS.

Step 8 Select the **Save & Exit** tab of BIOS.



Step 9 Select **IOS -XR Install**.

The BIOS UI displays the USB drive vendor in the brackets, in this case, SMART USB 1084.

The system detects USB and boots the image from USB.

```
Booting from USB..
Loading Kernel..
Verifying (loop)/boot/bzImage...
(loop)/boot/bzImage verified using attached signature.
Loading initrd..
Verifying (loop)/boot/initrd.img...
```

Step 10 Remove the USB drive after the `Rebooting the system after installation` message is displayed. The NCS 1014 reboots automatically.

Note
The USB must be removed only after the image is loaded successfully.

iPXE

iPXE is a Preboot Execution Environment that is included in the network card of the management interfaces and works at the system firmware (UEFI) level of the chassis. iPXE is used to reimage the system, and boot the chassis in case of boot failure or in the absence of a valid bootable partition. iPXE downloads the ISO image, proceeds with the installation of the image, and finally bootstraps inside the new installation.

Note The time taken for iPXE to download the ISO image depends on the network speed. Ensure that the network speed is sufficient to complete the image download in less than 10 minutes. The chassis reloads if the image is not downloaded by 10 minutes.

iPXE acts as a bootloader and provides the flexibility to choose the image that the system will boot based on the Platform Identifier (PID), the Serial Number, or the management mac-address. You must define iPXE in the DHCP server configuration file.

iPXE boot methods

To start the iPXE boot process, use one of these methods.

- **Preferred method:** run either the **reload bootmedia network location all** command or **reload bootmedia network location 0/RP0** command.
- **Alternative method:** power cycle the NCS 1014 chassis and start the iPXE boot process in the BIOS interface.

Note Software installation using iPXE boot with IPv6 is not supported.

Setup DHCP Server

A DHCP server must be configured for IPv4, IPv6, or both communication protocols.

Note For DHCPv6, a routing advertisement (RA) message must be sent to all nodes in the network that indicates which method is to be used to obtain the IPv6 address. Configure Router-advertise-daemon (radvd, install using yum install radvd) to allow the client to send the DHCP request. For example:

```
interface eth3
{
        AdvSendAdvert on;
        MinRtrAdvInterval 60;
        MaxRtrAdvInterval 180;
        AdvManagedFlag on;
        AdvOtherConfigFlag on;
        prefix 2001:1851:c622:1::/64
        {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr off;
        };
};
```

To setup a DHCP server:

1. Create the `dhcpd.conf` file (for IPv4, IPv6 or both communication protocols), `dhcpv6.conf` file (for IPv6) or both in the `/etc/` directory. This configuration file stores the network information such as the path to the script, location of the ISO install file, location of the provisioning configuration file, serial number, MAC address of the chassis.
2. Test the server when the DHCP server is running:

   For example, for ipv4:

   a. Use MAC address of the chassis:

      Ensure that the above configuration is successful.

   b. Use serial number of the chassis:

      ```
      host demo {
      option dhcp-client-identifier "<chassis-serial-number>";
        filename "http://<IP-address>/<hardware-platform>-mini-x.iso";
        fixed-address <IP-address>;
      }
      ```

      The serial number of the chassis is derived from the BIOS and is used as an identifier.

      Example

      ```
      host 10.89.205.202 {
             hardware ethernet 40:55:39:xx:xx:xx;
              option dhcp-client-identifier "<FCB2437B066>";
                if exists user-class and option user-class = "iPXE" {
                filename "http://10.89.205.127/box1/ncs1010-x64.iso";
             } else {
                 filename "http://10.89.205.127/box1/StartupConfig.cfg";
             }
             fixed-address 10.89.205.202;
          }
      ```

iPXE boot process

Follow these steps to run the iPXE boot process using the CLI and BIOS interface.

Before you begin

Before you perform the iPXE boot, ensure that:

- DHCP server is set and running.
- Management port of the NCS 1014 chassis is in UP state.

Procedure

Step 1 (Preferred method) Run the **reload bootmedia network location all** command or **reload bootmedia network location 0/RP0** command to initiate the iPXE boot process and reimage the chassis.

Example:

```
RP/0/RP0/CPU0:ios#reload bootmedia network location all
Mon Dec  4 09:49:14.220 UTC
Proceed with reload? [confirm]
Preparing system for backup. This may take a few minutes especially for large configurations.
 Status report: node0_RP0_CPU0: BACKUP INPROGRESS
 Status report: node0_RP0_CPU0: BACKUP HAS COMPLETED SUCCESSFULLY
[Done]

[ 3490.493853] systemd-shutdown[1]: Could not detach DM /dev/dm-9: Device or resource busy
[ 3490.601094] systemd-shutdown[1]: Could not detach DM /dev/dm-8: Device or resource busy
[ 3490.710401] systemd-shutdown[1]: Could not detach DM /dev/dm-7: Device or resource busy
[ 3490.849417] systemd-shutdown[1]: Failed to finalize  DM devices, ignoring
[ 3492.144874] Unsupported TPM Send Cmd! tpm_tag=8001,tpm_ordinal=0145
[ 3492.229149] tpm tpm0: tpm_try_transmit: send(): error -11
[ 3492.307885] reboot: Restarting system
Shelf Assembly Reset
NCS1014, Initializing Devices
Booting from Primary Flash
Aldrin: Programmed MI 4
Continue boot...
Version 2.19.1266. Copyright (C) 2023 American Megatrends, Inc.
BIOS Date: 10/06/2023 16:47:27 Ver: 0ACHI0480
Press <DEL> or <ESC> to enter setup.
TAM: Chip DB Verified

Software Boot OK, Validated

iPXE initialising devices...ok

iPXE 1.0.0+ (8b3e3) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP HTTPS TFTP VLAN EFI ISO9660 ISO9660_grub Menu
Trying net0-2051 and net0-2052...
net0-2051: 40:14:82:ba:d1:42 using NII on NII-0000:06:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
  [Link status: Unknown (http://ipxe.org/1a086194)]
```

Example:

```
RP/0/RP0/CPU0:sanity_176#reload bootmedia network location 0/RP0
Mon Feb 10 16:34:42.282 IST
Proceed with reload? [confirm]
```

```
RP/0/RP0/CPU0:Feb 10 16:34:44.277 IST: shelfmgr_exec_cli[65652]: %PLATFORM-SHELFMGR-6-USER_OP : User
 mols requested 'PXE reimage' of 0/RP0
RP/0/RP0/CPU0:Feb 10 16:34:44.284 IST: processmgr[51]: %OS-SYSMGR-6-INFO : Received request for
graceful go-down from shelfmgr. Reload Reason:User initiated card reimage-network. Timeout 55

Preparing system for backup. This may take a few minutes especially for large configurations.
RP/0/RP0/CPU0:Feb 10 16:34:44.284 IST: processmgr[51]: %OS-SYSMGR-6-INFO : Prepared RMF to reboot
 Status report: node0_RP0_CPU0: BACKUP INPROGRESS
RP/0/RP0/CPU0:sanity_176# Status report: node0_RP0_CPU0: BACKUP HAS COMPLETED SUCCESSFULLY
[Done]
RP/0/RP0/CPU0:Feb 10 16:34:45.974 IST: obflmgr[317]: %PLATFORM-OBFL-6-INFO : Unmounted OBFL directory
 for 0/Rack
RP/0/RP0/CPU0:Feb 10 16:34:46.225 IST: obflmgr[317]: %PLATFORM-OBFL-6-INFO : Unmounted OBFL directory
 for 0/RP0/CPU0
RP/0/RP0/CPU0:Feb 10 16:34:46.555 IST: processmgr[51]: %MGBL-SCONBKUP-6-INTERNAL_INFO : Reload debug
 script successfully spawned
--
--
[438336.926067] reboot: Restarting system


..         *** Cisco ***
System Initializing..
..


Initializing Devices

Booting from Primary Flash
MI: Skipping reprogram
Version 2.19.1266. Copyright (C) 2024 American Megatrends, Inc.
BIOS Date: 10/28/2024 12:47:04 Ver: 0ACHI0500
Press <DEL> or <ESC> to enter setup.
TAM: Chip DB Verified

Software Boot OK, Validated

iPXE initialising devices...ok

iPXE 1.0.0+ (8b3e3) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP HTTPS TFTP VLAN EFI ISO9660 ISO9660_grub Menu
Trying net0-2051 and net0-2052...
net0-2051: 9c:38:18:88:a4:1d using NII on NII-0000:06:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
  [Link status: Unknown (http://ipxe.org/1a086194)]
Configuring (net0-2051 9c:38:18:88:a4:1d)...... ok
--
--
net0-2052: fe80::9e38:18ff:fe88:a41e/64
Filename: http://4.25.30.34/Kepler/25_1_1_Images/ncs1010-x64.iso
http://4.25.30.34/Kepler/25_1_1_Images/ncs1010-x64.iso... ok
ncs1010-x64.iso : 1810485248 bytes
Booting /EFI/BOOT/bootx64.efi
Welcome to GRUB!
```

Step 2 (Alternative method) Initiate the boot process using the **BIOS** interface.

a) Reboot NCS 1014 using power cycle or console.
b) Press **Esc** to enter BIOS.
c) Select the **Save & Exit** tab of BIOS.
d) Choose **UEFI: iPXE Network Boot**.

Example:

```
TAM: Chip DB Verified
Software Boot OK, Validated
iPXE initialising devices...ok
iPXE 1.0.0+ (8b3e3) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP HTTPS TFTP VLAN EFI ISO9660 ISO9660_grub Menu
Trying net0-2051 and net0-2052...
net0-2051: 9c:38:18:88:a4:1d using NII on NII-0000:06:00.0 (open)
[Link:down, TX:0 TXE:0 RX:0 RXE:0]
[Link status: Unknown (http://ipxe.org/1a086194)]
Configuring (net0-2051 9c:38:18:88:a4:1d)...... ok
net0: fe80::9e38:18ff:fe88:a41d/64
net1: fe80::9e38:18ff:fe88:a420/64 (inaccessible)
net2: fe80::9e38:18ff:fe88:a421/64 (inaccessible)
net3: fe80::9e38:18ff:fe88:a422/64 (inaccessible)
net0-2051: 4.25.9.3/255.255.0.0 gw 4.25.0.1
net0-2051: fe80::9e38:18ff:fe88:a41d/64
net0-2052: fe80::9e38:18ff:fe88:a41e/64
Filename: http://4.25.30.34/Kepler/25_1_1_Images/ncs1010-x64.iso
http://4.25.30.34/Kepler/25_1_1_Images/ncs1010-x64.iso... ok
ncs1010-x64.iso : 1810485248 bytes
Booting /EFI/BOOT/bootx64.efi
Welcome to GRUB!
/EndEntire
error: no such device: ((cd0)/EFI/BOOT)/EFI/BOOT/grub.cfg.
Verifying (cd0)/EFI/BOOT/grub.cfg...
(cd0)/EFI/BOOT/grub.cfg verified using Pkcs7 signature.
```

Bring-Up Line Card

Use the following procedure to bring-up the NCS1014 line cards:

Procedure

Step 1 Insert the line card into the slot.

Step 2 Wait until the line card LED turns Green.

Step 3 Check the PID is in **OPERATIONAL** status using the **show platform** command.

Example:

CCMD-16-C and CCM-16-L line cards

```
RP/0/RP0/CPU0:ios#show platform
Fri Sep 22 06:56:28.653 UTC
Node               Type                       State                 Config state
--------------------------------------------------------------------------------
0/RP0/CPU0         NCS1K14-CNTLR-K9(Active)   IOS XR RUN            NSHUT,NMON
0/PM0              NCS1K4-AC-PSU              OPERATIONAL           NSHUT,NMON
0/FT0              NCS1K14-FAN                OPERATIONAL           NSHUT,NMON
0/FT1              NCS1K14-FAN                OPERATIONAL           NSHUT,NMON
0/FT2              NCS1K14-FAN                OPERATIONAL           NSHUT,NMON
0/0/NXR0           NCS1K14-CCMD-16-L          OPERATIONAL           NSHUT,NMON
0/2/NXR0           NCS1K14-CCMD-16-C          OPERATIONAL           NSHUT,NMON
0/3/NXR0           NCS1K14-CCMD-16-C          OPERATIONAL           NSHUT,NMON
```

Example:

2.4T line card

```
RP/0/RP0/CPU0:ios#show platform
Fri Sep 22 06:56:28.653 UTC
Node              Type                       State                     Config state
--------------------------------------------------------------------------------
0/RP0/CPU0        NCS1K14-CNTLR-K9(Active) IOS XR RUN                NSHUT,NMON
0/PM0             NCS1K4-AC-PSU            OPERATIONAL               NSHUT,NMON
0/FT0             NCS1K14-FAN              OPERATIONAL               NSHUT,NMON
0/FT1             NCS1K14-FAN              OPERATIONAL               NSHUT,NMON
0/FT2             NCS1K14-FAN              OPERATIONAL               NSHUT,NMON
0/2/NXR0          NCS1K14-2.4T-K9          OPERATIONAL               NSHUT,NMON
0/3/NXR0          NCS1K14-BLANK             PRESENT                  NSHUT,NMON
```

Example:

2.4TX line card

```
RP/0/RP0/CPU0:ios#show platform
Node              Type                       State                     Config state
--------------------------------------------------------------------------------
0/RP0/CPU0        NCS1K14-CNTLR-K9(Active) IOS XR RUN                NSHUT,NMON
0/PM0             NCS1K4-AC-PSU-2           OPERATIONAL              NSHUT,NMON
0/FT0             NCS1K14-FAN              OPERATIONAL               NSHUT,NMON
0/FT1             NCS1K14-FAN              OPERATIONAL               NSHUT,NMON
0/FT2             NCS1K14-FAN              OPERATIONAL               NSHUT,NMON
0/0/NXR0          NCS1K14-2.4T-X-K9        OPERATIONAL               NSHUT,NMON
0/1/NXR0          NCS1K14-2.4T-X-K9        OPERATIONAL               NSHUT,NMON
0/2/NXR0          NCS1K14-2.4T-X-K9        OPERATIONAL               NSHUT,NMON
0/3/NXR0          NCS1K14-2.4T-X-K9        OPERATIONAL               NSHUT,NMON
```

Step 4 Check the line card environment parameters using the command **show environment [power | voltage | current | temperature] [location |** *location* **]**.

Example:

```
RP/0/RP0/CPU0:ios#show environment power
================================================================================
CHASSIS LEVEL POWER INFO: 0
================================================================================
   Total output power capacity (Group 0 + Group 1) :    2500W +     0W
   Total output power required                     :    1636W
   Total power input                               :     637W
   Total power output                              :     568W

Power Group 0:
================================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type            Volts     Amps    Volts     Amps
================================================================================
   0/PM1       NCS1K4-AC-PSU-2 227.5      2.8     12.1      47.0     OK

Total of Group 0:              637W/2.8A         568W/47.0A


================================================================================
Location     Card Type        Power               Power             Status
                              Allocated           Used
                              Watts               Watts
================================================================================
0/RP0/CPU0  NCS1K14-CNTLR-K9   73                   20                ON
0/FT0       NCS1K14-FAN        170                  167               ON
0/FT1       NCS1K14-FAN        170                  85                ON
```

```
0/FT2       NCS1K14-FAN        170                 159               ON
0/0/NXR0    NCS1K14-2.4T-L-K9  460                 38                ON
0/2/NXR0    NCS1K14-2.4T-X-K9  410                 0                 ON
0/3/NXR0    NCS1K14-CCMD-16-C  110                 16                ON
0/Rack      NCS1014            73                  14                ON
```

Note

- When a slot is not in use, we recommend inserting a filler to allow proper airflow across the line cards to maintain an optimal system temperature.
- When a port is not in use, we recommend inserting a clip to maintain an optimal card temperature.
- Ensure to secure the line card in the chassis by tightening the top and bottom screws.

Step 5 Upgrade the FPDs of the line card depending on the output of **show hw-module location 0**/*line-card-slot* **fpd** command.

Configure NTP Server

Understand NTP

Table 1: Feature History

Feature Name	Release Information	Feature Description
NTP Support	Cisco IOS XR Release 7.11.1	Network Time Protocol (NTP) allows devices to synchronize clocks with the NTP servers, maintaining the most accurate time. NCS 1010 now supports time synchronization. In modern and large networks, time synchronization is critical because every aspect of managing, securing, planning, and debugging a network depends on the time of occurrence of events. Commands added: • **ntp server** • **show ntp associations** • **show ntp status**

NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communication uses Coordinated Universal Time (UTC). An NTP network usually receives its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP distributes this time across the network.

NTP uses the concept of a “stratum” to describe how many NTP hops away a machine is from an authoritative time source. A “stratum 1” time server typically has an authoritative time source (such as a radio or atomic clock, or a GPS time source) directly attached, a “stratum 2” time server receives its time through NTP from a “stratum 1” time server, and so on.

The communications between machines running NTP (known as associations) are usually statically configured; each machine is given the IP address of all machines with which it should form associations. Accurate timekeeping is made possible by exchanging NTP messages between each pair of machines with an association.

An NTP broadcast client listens for broadcast messages sent by an NTP broadcast server at a designated IPv4 address. The client synchronizes the local clock using the first received broadcast message.

Synchronize Clock with NTP Server

IOS XR has an independent system clock that must be synchronized with an NTP server to prevent deviation from true time.

To sync the clock with an NTP server, perform these steps:

Before you begin

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:ios# configure
```

Enters the configuration mode.

Step 2 **ntp server** *ntp-server-ip-address* **version** *version-number* **prefer iburst**

Example:

IPv4:

```
RP/0/RP0/CPU0:ios(config)#ntp server 10.127.60.137 version 4 prefer iburst
```

IPv6:

```
RP/0/RP0/CPU0:ios(config)#ntp server 2001:DB8::1 version 4 prefer iburst
```

Synchronizes the console clock with the specified NTP server.

Note
The NTP server can also be reached through a VRF if the management interface is in a VRF.

Step 3 **commit**

Example:

```
RP/0/RP0/CPU0:ios(config-ntp)#commit
```

Commits the configuration.

Step 4 **show ntp associations**

Example:

```
RP/0/RP0/CPU0:ios#show ntp associations

Tue Oct 17 06:32:54.389 UTC

      address         ref clock     st  when  poll reach  delay  offset    disp
 ~10.0.0.1       10.64.58.50       2    15    64    1    1.95  -0.062  1937.7
 ~10.127.60.137    .STEP.           16     -    64    0    0.00   0.000   15937
 * sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

Displays the status of NTP associations.

Step 5 **show ntp status**

Example:

```
RP/0/RP0/CPU0:ios#show ntp status

Tue Oct 17 06:33:41.125 UTC
Clock is synchronized, stratum 3, reference is 10.0.0.1
nominal freq is 1000000000.0000 Hz, actual freq is 31762818.6272 Hz, precision is 2**24
reference time is E8D8A944.A4F2AEF1 (06:33:40.644 UTC Tue Oct 17 2023)
clock offset is -0.592 msec, root delay is 2.434 msec
root dispersion is 939.64 msec, peer dispersion is 938.03 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.0000304834 s/s
system poll interval is 64, last update was 1 sec ago
authenticate is disabled, panic handling is disabled,
hostname resolution retry interval is 1440 minutes.
```

Verifies that the clock is synchronized with the NTP server.

Verify the Status of the External Reference Clock

This task explains how to verify the status of NTP components.

Note The commands can be entered in any order.

Procedure

Step 1 **show ntp associations [detail] [location** *node-id***]**

Example:

```
RP/0/RP0/CPU0:ios#show ntp associations
Sun Nov 5 15:14:44.128 UTC

address ref clock st when poll reach delay offset disp
*~10.0.0.1 10.64.58.50 2 81 128 377 1.84 7.802 2.129
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

Displays the status of NTP associations.

Example:

```
RP/0/RP0/CPU0:ios#show ntp associations detail
Sun Nov 5 15:14:48.763 UTC

10.0.0.1 configured, our_master, stratum 2
ref ID 10.64.58.50, time E8F22BB9.79D4A841 (14:56:57.475 UTC Sun Nov 5 2023)
our mode client, peer mode server, our poll intvl 128, peer poll intvl 128
root delay 0.6866 msec, root disp 1.04, reach 377, sync dist 6.2590
delay 1.84 msec, offset 7.802 msec, dispersion 2.129
precision 2**23, version 4
org time E8F22F92.B647E8FC (15:13:22.712 UTC Sun Nov 5 2023)
rcv time E8F22F92.B88F303C (15:13:22.720 UTC Sun Nov 5 2023)
xmt time E8F22F92.B88F303C (15:13:22.720 UTC Sun Nov 5 2023)
filtdelay = 1.844 1.772 1.983 1.954 1.945 2.000 1.902 1.778
filtoffset = 7.857 7.802 8.065 8.063 8.332 8.397 8.664 8.684
filterror = 0.000 0.060 1.995 2.055 4.050 4.110 6.060 6.120
```

Example:

```
RP/0/RP0/CPU0:ios#show ntp associations detail location 0/RP0/CPU0
Sun Nov 5 15:38:15.744 UTC

10.0.0.1 configured, our_master, stratum 2
ref ID 10.64.58.50, time E8F233C0.5606A159 (15:31:12.336 UTC Sun Nov 5 2023)
our mode client, peer mode server, our poll intvl 128, peer poll intvl 128
root delay 0.7019 msec, root disp 0.47, reach 377, sync dist 5.6762
delay 2.01 msec, offset 7.226 msec, dispersion 3.856
precision 2**23, version 4
org time E8F23563.DE5D42D5 (15:38:11.868 UTC Sun Nov 5 2023)
rcv time E8F23563.E07C296D (15:38:11.876 UTC Sun Nov 5 2023)
xmt time E8F23563.E07C296D (15:38:11.876 UTC Sun Nov 5 2023)
filtdelay = 2.006 1.865 1.936 1.762 1.932 1.875 1.881 2.011
filtoffset = 7.210 7.305 7.372 7.226 7.298 7.258 7.251 7.224
filterror = 0.000 2.025 2.085 4.035 4.095 6.060 6.120 8.070
```

Step 2 **show ntp status [location** *node-id***]**

Example:

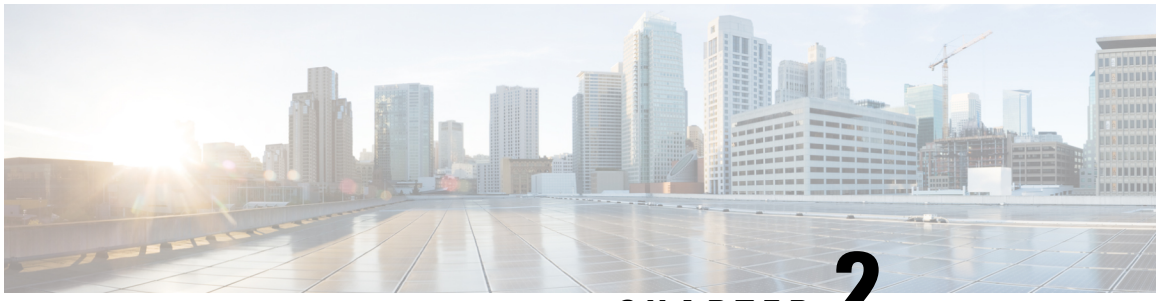
```
RP/0/RP0/CPU0:ios#show ntp status
Sun Nov 5 15:14:36.949 UTC

Clock is synchronized, stratum 3, reference is 10.0.0.1
nominal freq is 1000000000.0000 Hz, actual freq is 44881851.3383 Hz, precision is 2**24
reference time is E8F22D7A.AB020D97 (15:04:26.668 UTC Sun Nov 5 2023)
clock offset is 9.690 msec, root delay is 2.553 msec
root dispersion is 24.15 msec, peer dispersion is 2.13 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.0000212807 s/s
system poll interval is 128, last update was 610 sec ago
authenticate is disabled, panic handling is disabled,
hostname resolution retry interval is 1440 minutes.
```

Verifies that the clock is synchronized with the NTP server.

NTP Troubleshooting Information

For NTP troubleshooting information, see here.



CHAPTER 2

Perform Preliminary Checks

After successfully logging into the console, you must perform some preliminary checks to verify the default setup. If any setup issue is detected, take corrective action before making further configurations.

Note The output of the examples in the procedures is not from the latest software release. The output will change for any explicit references to the current release.

Note

- Refer to System Health Check for monitoring systems in a network to proactively prevent potential issues and take preventative steps.

- Inventory Support in NCS 1014, on page 29
- Verify Status of Hardware Components, on page 36
- Verify Software Version, on page 38
- Verify Environmental Parameters, on page 39
- Verify Management Interface Status, on page 42
- Verify Firmware Version, on page 44
- Verify Alarms, on page 46
- Verify Context, on page 48
- Verify Core Files, on page 53

Inventory Support in NCS 1014

Table 2: Feature History

Feature Name	Release Information	Description
Inventory Support	Cisco IOS XR Release 7.11.1	Inventory support and pluggable optics support for QSFP28,QSFP-DD and Coherent Interface Module (CIM 8) are enabled in NCS 1014 system.

The NCS 1014 inventory model consists of the following components.

- One NCS 1014 controller card.
- NCS 1014 chassis.
- Two AC or DC power supply units (PSU) of 2KW and 2.5KW.
- Three FAN trays.
- Four line cards.

The components are connected to a 2RU chassis. NCS 1014 can support upto four line cards at once at any given point in time. The line cards supported are 1.2T, NCS1K4-2.4T-K9, CCMD-16-C and CCMD-16-L cards.

Note **CCMD-16-C**: refers to the NCS1K14-CCMD-16-C card.

CCMD-16-L: refers to the NCS1K14-CCMD-16-L card.

1.2T : refers to the NCS1K4-1.2T-K9 C-band card.

2.4T: refers to the NCS1K4-2.4T-K9 C-band card.

The show inventory command retrieves and displays the inventory information about each Cisco product in the form of a Unique Device Identifier (UDI). The UDI is a combination of three separate data elements: a product identifier (PID), a version identifier (VID), and the serial number (SN). The PID is the name by which the product is ordered. It is also known as product name or part number.

The VID is the version of the product. Whenever a product is upgraded the VID gets incremented according to the changes added. The SN is the vendor based unique serial number assigned to any product. It is used to identifying any specific product.

Verify Inventory

The **show inventory** command displays the details of the hardware inventory of NCS 1014.

To verify the inventory information for all the physical entities, use the following command.

show inventory [all | details | fan | power | vendor-type | raw | chasis | word] [location | *location* **]**.

Note The various options available under **show inventory** command are listed below.

- **Word**: Partially qualified location specification
- **All**: Inventory information for all the physical entities
- **Chassis**: Inventory information about chassis
- **Details**: Detailed entity information
- **Fan**: Inventory information about fan
- **Location**: Location of node for inventory
- **Power**: Inventory information about power
- **Raw**: Raw information
- **Vendor-type**: Vendor type information

Example

```
RP/0/RP0/CPU0:ios#show inventory ?
  WORD         Partially qualified location specification
  all          Inventory information for all the physical entities
  chassis      Inventory information about chassis
  details      detailed entity information
  fan          Inventory information about fan
  location     Location of node for inventory
  power        Inventory information about power
  raw          raw information
  vendor-type  vendor-type information
  |            Output Modifiers
  <cr>
```

Procedure

show inventory

When you execute this command in the Cisco IOS XR EXEC mode, it displays the summary of NCS 1014 inventory based on different card and optics pluggables on all the slots or ports.

Example:

```
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios# show inventory
Thu Oct  5 02:32:14.231 UTC

NAME: "Rack 0", DESCR: "Network Convergence System 1014 chassis with timing support"
PID: NCS1014           , VID: V00, SN: FCB2717B151

NAME: "0/RP0/CPU0", DESCR: "Network Convergence System 1014 Controller"
PID: NCS1K14-CNTLR-K9  , VID: V00, SN: FCB2718B1AX

NAME: "0/0/NXR0", DESCR: "NCS1K4 12x QSFP28 2 Trunk C-Band DWDM card"
PID: NCS1K4-1.2T-K9    , VID: V00, SN: CAT2250B0B9
```

```
NAME: "0/1/NXR0", DESCR: "Network Convergence System 1014 2.4T Line Card"
PID: NCS1K14-2.4T-K9   , VID: V00, SN: FCB2710B0L5

NAME: "Optics0/1/0/0", DESCR: "Cisco CIM8 C K9 Pluggable Optics Module"
PID: CIM8-C-K9         , VID: VES1, SN: SIM-AX12-SW

NAME: "Optics0/1/0/1", DESCR: "Cisco 100G QSFP28 SR4-S Pluggable Optics Module"
PID: QSFP-100G-SR4-S   , VID: ES1 , SN: AVF1933G18C

NAME: "Optics0/1/0/2", DESCR: "Non-Cisco UNKNOWN TYPE Pluggable Optics Module"
PID: TR-IQ13L-N00      , VID: 1B, SN: INFBH1940242

NAME: "Optics0/1/0/3", DESCR: "Cisco UNKNOWN TYPE Pluggable Optics Module"
PID: ONS-QSFP-4X10-MLR , VID: V01 , SN: INL21010375

NAME: "Optics0/1/0/4", DESCR: "Cisco 100G QSFP28 SR4-S Pluggable Optics Module"
PID: QSFP-100G-SR4-S   , VID: ES1 , SN: AVF1933G16A

NAME: "Optics0/1/0/6", DESCR: "Cisco QSFP DD 400G FR4 S Pluggable Optics Module"
PID: QDD-400G-FR4-S    , VID: V01 , SN: FIW250504DL

NAME: "Optics0/1/0/7", DESCR: "Cisco CIM8 C K9 Pluggable Optics Module"
PID: CIM8-C-K9         , VID: VES1, SN: ACA27370055

NAME: "0/2/NXR0", DESCR: "NCS1K4 12x QSFP28 2 Trunk C-Band DWDM card"
PID: NCS1K4-1.2T-K9    , VID: V03, SN: CAT2329B32K

NAME: "Optics0/2/0/10", DESCR: "Cisco QSFP28 100G CU1M Pluggable Optics Module"
PID: QSFP-100G-CU1M    , VID: V01 , SN: LCC2402GKJ3-B

NAME: "Optics0/2/0/11", DESCR: "Cisco 100G QSFP28 LR-S Pluggable Optics Module"
PID: QSFP-100G-LR-S    , VID: ES0 , SN: FBN2321A013

NAME: "Optics0/2/0/12", DESCR: "Cisco 100G QSFP28 AOC Pluggable Optics Module"
PID: QSFP-100G-AOC3M   , VID: V03 , SN: INL23302076-B

NAME: "Optics0/2/0/13", DESCR: "Cisco 100G QSFP28 LR-S Pluggable Optics Module"
PID: QSFP-100G-LR-S    , VID: ES0 , SN: FBN2321A024

NAME: "Optics0/2/0/3", DESCR: "Cisco QSFP28 100G CU1M Pluggable Optics Module"
PID: QSFP-100G-CU1M    , VID: V01 , SN: LCC2402GKJ3-A

NAME: "Optics0/2/0/4", DESCR: "Cisco 100G QSFP28 CWDM4 Pluggable Optics Module"
PID: QSFP-100G-CWDM4-S , VID: V02 , SN: JFQ2210800T

NAME: "Optics0/2/0/5", DESCR: "Cisco 100G QSFP28 AOC Pluggable Optics Module"
PID: QSFP-100G-AOC3M   , VID: V03 , SN: INL23302076-A

NAME: "Optics0/2/0/6", DESCR: "Non-Cisco 100G QSFP28 CWDM4 Pluggable Optics Module"
PID: FTLC1152RGPL2-G2  , VID: A0, SN: UYL0AL9

NAME: "Optics0/2/0/7", DESCR: "Non-Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: FIM37700/171      , VID: 01, SN: 37700171ZZ00PK

NAME: "Optics0/2/0/8", DESCR: "Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: ONS-QSFP28-LR4    , VID: V01 , SN: FNS20520RM6

NAME: "0/3/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A

NAME: "0/FT0", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B15J

NAME: "0/FT1", DESCR: "Network Convergence System 1014 FAN Module"
```

```
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B15L

NAME: "0/FT2", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B15E

NAME: "0/PM0", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V00, SN: POG2221CL0Z

NAME: "0/PM1", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V01, SN: POG2505CL53
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#


RP/0/RP0/CPU0:ios#show inventory all
Mon Nov 27 11:01:53.452 UTC

NAME: "Rack 0", DESCR: "Network Convergence System 1014 chassis with timing support"
PID: NCS1014           , VID: V00, SN: FCB2717B13C

NAME: "0/RP0/CPU0", DESCR: "Network Convergence System 1014 Controller"
PID: NCS1K14-CNTLR-K9  , VID: V00, SN: FCB2723B0CX

NAME: "0/0/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A

NAME: "0/1/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A

NAME: "0/2/NXR0", DESCR: "Network Convergence System 1014 2.4T Line Card"
PID: NCS1K14-2.4T-K9   , VID: V00, SN: FCB2726B067

NAME: "Optics0/2/0/4", DESCR: "Cisco QSFP DD 400G DR4 S Pluggable Optics Module"
PID: QDD-400G-DR4-S    , VID: V01 , SN: CGC25512003

NAME: "Optics0/2/0/5", DESCR: "Cisco QSFP DD 400G FR4 S Pluggable Optics Module"
PID: QDD-400G-FR4-S    , VID: V01 , SN: CGC26371408

NAME: "Optics0/2/0/7", DESCR: "Cisco CIM8 C K9 Pluggable Optics Module"
PID: CIM8-C-K9         , VID: VES1, SN: ACA274500CR

NAME: "0/3/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A

NAME: "0/FT0", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B192

NAME: "0/FT1", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B197

NAME: "0/FT2", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B19U

NAME: "0/PM0", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V01, SN: POG2727CLP6

NAME: "0/PM1", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V01, SN: POG2727CLKS
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
```

To display location based inventory details use this command.

```
RP/0/RP0/CPU0:ios#show inventory location 0/2/NXR0
Thu Oct  5 02:35:30.251 UTC

NAME: "0/2/NXR0", DESCR: "NCS1K4 12x QSFP28 2 Trunk C-Band DWDM card"
PID: NCS1K4-1.2T-K9    , VID: V03, SN: CAT2329B32K

NAME: "Optics0/2/0/10", DESCR: "Cisco QSFP28 100G CU1M Pluggable Optics Module"
PID: QSFP-100G-CU1M    , VID: V01 , SN: LCC2402GKJ3-B

NAME: "Optics0/2/0/11", DESCR: "Cisco 100G QSFP28 LR-S Pluggable Optics Module"
PID: QSFP-100G-LR-S    , VID: ES0 , SN: FBN2321A013

NAME: "Optics0/2/0/12", DESCR: "Cisco 100G QSFP28 AOC Pluggable Optics Module"
PID: QSFP-100G-AOC3M   , VID: V03 , SN: INL23302076-B

NAME: "Optics0/2/0/13", DESCR: "Cisco 100G QSFP28 LR-S Pluggable Optics Module"
PID: QSFP-100G-LR-S    , VID: ES0 , SN: FBN2321A024

NAME: "Optics0/2/0/3", DESCR: "Cisco QSFP28 100G CU1M Pluggable Optics Module"
PID: QSFP-100G-CU1M    , VID: V01 , SN: LCC2402GKJ3-A

NAME: "Optics0/2/0/4", DESCR: "Cisco 100G QSFP28 CWDM4 Pluggable Optics Module"
PID: QSFP-100G-CWDM4-S , VID: V02 , SN: JFQ2210800T

NAME: "Optics0/2/0/5", DESCR: "Cisco 100G QSFP28 AOC Pluggable Optics Module"
PID: QSFP-100G-AOC3M   , VID: V03 , SN: INL23302076-A

NAME: "Optics0/2/0/6", DESCR: "Non-Cisco 100G QSFP28 CWDM4 Pluggable Optics Module"
PID: FTLC1152RGPL2-G2  , VID: A0, SN: UYL0AL9

NAME: "Optics0/2/0/7", DESCR: "Non-Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: FIM37700/171      , VID: 01, SN: 37700171ZZ00PK

NAME: "Optics0/2/0/8", DESCR: "Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: ONS-QSFP28-LR4    , VID: V01 , SN: FNS20520RM6
RP/0/RP0/CPU0:ios#


RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#show inventory location 0/1/NXR0
Thu Oct  5 02:38:13.791 UTC

NAME: "0/1/NXR0", DESCR: "Network Convergence System 1014 2.4T Line Card"
PID: NCS1K14-2.4T-K9   , VID: V00, SN: FCB2710B0L5

NAME: "Optics0/1/0/0", DESCR: "Cisco CIM8 C K9 Pluggable Optics Module"
PID: CIM8-C-K9         , VID: VES1, SN: SIM-AX12-SW

NAME: "Optics0/1/0/1", DESCR: "Cisco 100G QSFP28 SR4-S Pluggable Optics Module"
PID: QSFP-100G-SR4-S   , VID: ES1 , SN: AVF1933G18C

NAME: "Optics0/1/0/2", DESCR: "Non-Cisco UNKNOWN TYPE Pluggable Optics Module"
PID: TR-IQ13L-N00      , VID: 1B, SN: INFBH1940242

NAME: "Optics0/1/0/3", DESCR: "Cisco UNKNOWN TYPE Pluggable Optics Module"
PID: ONS-QSFP-4X10-MLR , VID: V01 , SN: INL21010375

NAME: "Optics0/1/0/4", DESCR: "Cisco 100G QSFP28 SR4-S Pluggable Optics Module"
PID: QSFP-100G-SR4-S   , VID: ES1 , SN: AVF1933G16A

NAME: "Optics0/1/0/6", DESCR: "Cisco QSFP DD 400G FR4 S Pluggable Optics Module"
PID: QDD-400G-FR4-S    , VID: V01 , SN: FIW250504DL

NAME: "Optics0/1/0/7", DESCR: "Cisco CIM8 C K9 Pluggable Optics Module"
```

```
PID: CIM8-C-K9         , VID: VES1, SN: ACA27370055
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
```

To display chassis based inventory details use this command.

```
RP/0/RP0/CPU0:ios#show inventory chassis
Mon Nov 27 11:02:05.083 UTC

NAME: "Rack 0", DESCR: "Network Convergence System 1014 chassis with timing support"
PID: NCS1014           , VID: V00, SN: FCB2717B13C
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#sh inventory details
Mon Nov 27 11:02:23.095 UTC

NAME: "Rack 0", DESCR: "Network Convergence System 1014 chassis with timing support"
PID: NCS1014           , VID: V00, SN: FCB2717B13C
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 8384513   , Type: Rack
PN: 800-111211-01, HW Ver: 0.1

NAME: "0/RP0/CPU0", DESCR: "Network Convergence System 1014 Controller"
PID: NCS1K14-CNTLR-K9  , VID: V00, SN: FCB2723B0CX
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 16385     , Type: Module
PN: 800-111209-01, HW Ver: 0.2

NAME: "0/0/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 1         , Type: Module
PN: 800-39505-01, HW Ver: 0.1

NAME: "0/1/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 4097      , Type: Module
PN: 800-39505-01, HW Ver: 0.1

NAME: "0/2/NXR0", DESCR: "Network Convergence System 1014 2.4T Line Card"
PID: NCS1K14-2.4T-K9   , VID: V00, SN: FCB2726B067
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 8193      , Type: Module
PN: 800-51107-01, HW Ver: 0.1

NAME: "Optics0/2/0/4", DESCR: "Cisco QSFP DD 400G DR4 S Pluggable Optics Module"
PID: QDD-400G-DR4-S    , VID: V01 , SN: CGC25512003
MFG_NAME: CISCO-CIG          , SNMP_IDX: 3129345   , Type: Module
PN: 10-3320-01, HW Ver: 0.0

NAME: "Optics0/2/0/5", DESCR: "Cisco QSFP DD 400G FR4 S Pluggable Optics Module"
PID: QDD-400G-FR4-S    , VID: V01 , SN: CGC26371408
MFG_NAME: CISCO-CIG          , SNMP_IDX: 3133441   , Type: Module
PN: 10-3321-01, HW Ver: 0.0

NAME: "Optics0/2/0/7", DESCR: "Cisco CIM8 C K9 Pluggable Optics Module"
PID: CIM8-C-K9         , VID: VES1, SN: ACA274500CR
MFG_NAME: CISCO-ACACIA       , SNMP_IDX: 3141633   , Type: Module
PN: 10-100471-01, HW Ver: 0.0

NAME: "0/3/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 12289     , Type: Module
PN: 800-39505-01, HW Ver: 0.1

NAME: "0/FT0", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B192
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 28673     , Type: Fantray
```

```
PN: 800-111210-01, HW Ver: 0.1

NAME: "0/FT1", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B197
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 32769     , Type: Fantray
PN: 800-111210-01, HW Ver: 0.1

NAME: "0/FT2", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B19U
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 36865     , Type: Fantray
PN: 800-111210-01, HW Ver: 0.1

NAME: "0/PM0", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V01, SN: POG2727CLP6
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 20481     , Type: Power Supply
PN: 341-100825-01, HW Ver: 0.1

NAME: "0/PM1", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V01, SN: POG2727CLKS
MFG_NAME: Cisco Systems, Inc., SNMP_IDX: 24577     , Type: Power Supply
PN: 341-100825-01, HW Ver: 0.1
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
```

To display fan based inventory details use this command.

```
RP/0/RP0/CPU0:ios#show inventory fan
Mon Nov 27 11:02:39.811 UTC

NAME: "0/FT0", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B192

NAME: "0/FT1", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B197

NAME: "0/FT2", DESCR: "Network Convergence System 1014 FAN Module"
PID: NCS1K14-FAN       , VID: V00, SN: FCB2720B19U
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#sh inventory location 0/3/NXR0
Mon Nov 27 11:02:46.903 UTC

NAME: "0/3/NXR0", DESCR: "Network Convergence System 1014 Filler"
PID: NCS1K14-BLANK     , VID: V01, SN: N/A
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#sh inventory power
Mon Nov 27 11:03:04.566 UTC

NAME: "0/PM0", DESCR: "Network Convergence System 1004 AC Power Supply Unit"
PID: NCS1K4-AC-PSU     , VID: V01, SN: POG2727CLP6
```

Verify Status of Hardware Components

To verify the status of all the hardware components installed on NCS 1014, perform the following procedure.

Before you begin

Ensure that all the required hardware components are installed on NCS 1014. For installation details, see *Cisco Network Convergence System 1014 Hardware Installation Guide*.

Procedure

Step 1 **show platform**

When you execute this command from the Cisco IOS XR EXEC mode, the status of Cisco IOS XR is displayed.

Example:

```
RP/0/RP0/CPU0:ios#show platform
Node              Type                      State                     Config state
--------------------------------------------------------------------------------
0/RP0/CPU0        NCS1K14-CNTLR-K9(Active)  IOS XR RUN                NSHUT,NMON
0/PM0             NCS1K4-AC-PSU             OPERATIONAL               NSHUT,NMON
0/PM1             NCS1K4-AC-PSU             OPERATIONAL               NSHUT,NMON
0/FT0             NCS1K14-FAN               OPERATIONAL               NSHUT,NMON
0/FT1             NCS1K14-FAN               OPERATIONAL               NSHUT,NMON
0/FT2             NCS1K14-FAN               OPERATIONAL               NSHUT,NMON
0/0/NXR0          NCS1K4-1.2T-K9            OPERATIONAL               NSHUT,NMON
0/1/NXR0          NCS1K14-2.4T-K9           OPERATIONAL               NSHUT,NMON
0/2/NXR0          NCS1K4-1.2T-K9            OPERATIONAL               NSHUT,NMON
0/3/NXR0          NCS1K4-1.2T-K9            OPERATIONAL               NSHUT,NMON
```

Step 2 **show inventory**

Displays details of the physical entities of NCS 1014 along with the details of QSFPs when you execute this command in Cisco IOS XR EXEC mode.

Example:

```
RP/0/RP0/CPU0:ios#show platform
Node              Type                      State                     Config state
--------------------------------------------------------------------------------
0/RP0/CPU0        NCS1K14-CNTLR-K9(Active)  IOS XR RUN                NSHUT,NMON
0/PM0             NCS1K4-AC-PSU             OPERATIONAL               NSHUT,NMON
0/PM1             NCS1K4-AC-PSU             OPERATIONAL               NSHUT,NMON
0/FT0             NCS1K14-FAN               OPERATIONAL               NSHUT,NMON
0/FT1             NCS1K14-FAN               OPERATIONAL               NSHUT,NMON
0/FT2             NCS1K14-FAN               OPERATIONAL               NSHUT,NMON
0/0/NXR0          NCS1K4-1.2T-K9            OPERATIONAL               NSHUT,NMON
0/1/NXR0          NCS1K14-2.4T-K9           OPERATIONAL               NSHUT,NMON
0/2/NXR0          NCS1K4-1.2T-K9            OPERATIONAL               NSHUT,NMON
0/3/NXR0          NCS1K4-1.2T-K9            OPERATIONAL               NSHUT,NMON


Step 2 show inventory

RP/0/RP0/CPU0:ios#show inventory

NAME: "Rack 0", DESCR: "Network Convergence System 1014 chassis with timing support"
PID: NCS1014           , VID: V00, SN: FCB2726B0AR

NAME: "0/RP0/CPU0", DESCR: "Network Convergence System 1014 Controller"
PID: NCS1K14-CNTLR-K9  , VID: V00, SN: FCB2726B0LR

NAME: "0/0/NXR0", DESCR: "NCS1K4 12x QSFP28 2 Trunk C-Band DWDM card"
PID: NCS1K4-1.2T-K9    , VID: V00, SN: CAT2250B0C4
```

```
NAME: "Optics0/0/0/9", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2638A153

NAME: "Optics0/0/0/10", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2638A6J0

NAME: "Optics0/0/0/11", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2638A73V

NAME: "Optics0/0/0/12", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2544A687

NAME: "Optics0/0/0/13", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2638A2D4

NAME: "Optics0/0/0/2", DESCR: "Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: ONS-QSFP28-LR4    , VID: V01 , SN: FNS2333080H

NAME: "Optics0/0/0/3", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2638A6J2

NAME: "Optics0/0/0/4", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V01 , SN: INL23243050

NAME: "Optics0/0/0/5", DESCR: "Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: ONS-QSFP28-LR4    , VID: V01 , SN: FNS23320DZF

NAME: "Optics0/0/0/6", DESCR: "Cisco 100G QSFP28 LR4 Pluggable Optics Module"
PID: ONS-QSFP28-LR4    , VID: V01 , SN: FNS26140JRK

NAME: "Optics0/0/0/7", DESCR: "Cisco 100G QSFP28 LR4-S Pluggable Optics Module"
PID: QSFP-100G-LR4-S   , VID: V02 , SN: FNS263509QL

NAME: "Optics0/0/0/8", DESCR: "Cisco 100G QSFP28 FR-S Pluggable Optics Module"
PID: QSFP-100G-FR-S    , VID: V02 , SN: FBN2638A5G5

NAME: "0/1/NXR0", DESCR: "Network Convergence System 1014 2.4T Line Card"
PID: NCS1K14-2.4T-K9   , VID: V00, SN: FCB2726B072
```

Verify Software Version

NCS 1014 is shipped with the Cisco IOS XR Software preinstalled. Verify that the latest version of the software is installed.

Procedure

show version

Displays the software version and details such as system uptime.

Example:

```
RP/0/RP0/CPU0:ios#sh version
Cisco IOS XR Software, Version 7.11.1.49I LNT
Copyright (c) 2013-2023 by Cisco Systems, Inc.
```

```
Build Information:
 Built By     : sajshah
 Built On     : Sun Nov 19 20:31:06 UTC 2023
 Build Host   : iox-ucs-077
 Workspace    : /auto/ioxdepot6/GISO/giso_build_lindt/giso_eng_create/yshivapp_2023-11-20_04-28-49_UTC

 Version      : 7.11.1.49I
 Label        : 7.11.1.49I-Weekly

cisco NCS1010 (C3758R @ 2.40GHz)
cisco NCS1014 (C3758R @ 2.40GHz) processor with 32GB of memory
KEPLER_PF6 uptime is 1 hour, 40 minutes
NCS 1014 - Chassis
```

Verify Environmental Parameters

The **show environment** command displays the environmental parameters of NCS 1014.

To verify the environmental parameters use the following commands **show environment [all | altitude | fan | power | voltage | current | temperature] [location |** *location* **]**.

The following example shows sample output of the **show environment** command with the **fan** keyword.

```
RP/0/RP0/CPU0:ios#show environment fan
=============================================================
                                                Fan speed (rpm)
Location      FRU Type                          FAN_0    FAN_1
-------------------------------------------------------------

0/PM0         NCS1K4-DC-PSU-2                    11520   11216
0/PM1         NCS1K4-DC-PSU-2                    12256   12128
0/FT0         NCS1K14-FAN                        11400    9960
0/FT1         NCS1K14-FAN                        11340    9960
0/FT2         NCS1K14-FAN                        11400    9960
```

The following example shows sample output of the **show environment** command with the **power** keyword.

```
RP/0/RP0/CPU0:ios#sh environment power
Tue Nov 28 14:14:52.169 UTC
================================================================================
CHASSIS LEVEL POWER INFO: 0
================================================================================
   Total output power capacity (Group 0 + Group 1) :    2000W +     2000W
   Total output power required                     :    1896W
   Total power input                               :     741W
   Total power output                              :     653W

Power Group 0:
================================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type            Volts     Amps    Volts     Amps
================================================================================
   0/PM0       NCS1K4-AC-PSU   224.0     1.8     12.1      28.9     OK

Total of Group 0:              403W/1.8A         349W/28.9A

Power Group 1:
```

```
==============================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type            Volts     Amps    Volts     Amps
==============================================================================
   0/PM1       NCS1K4-AC-PSU   225.2     1.5     12.1      25.2     OK

Total of Group 1:              337W/1.5A         304W/25.2A

==============================================================================
   Location     Card Type               Power       Power        Status
                                        Allocated   Used
                                        Watts       Watts
==============================================================================
   0/FT0        NCS1K14-FAN             170         27           ON
   0/FT1        NCS1K14-FAN             170         27           ON
   0/FT2        NCS1K14-FAN             170         28           ON
   0/0/NXR0     NCS1K4-1.2T-K9          260         220          ON
   0/1/NXR0     NCS1K4-1.2T-K9          260         221          ON
   0/2/NXR0     NCS1K4-1.2T-K9          260         54           ON
   0/3/NXR0     NCS1K14-2.4T-K9         460         15           ON
   0/Rack       NCS1014                 73          14           ON
```

The following example shows sample output of the **show environment** command with the **temperature** keyword.

```
RP/0/RP0/CPU0:ios#show environment temperature location 0/rp0/CPU0
==============================================================================

          ==============================
Location  TEMPERATURE                          Value     Crit    Major    Minor

            Minor    Major    Crit
          Sensor                             (deg C)     (Lo)     (Lo)     (Lo)

             (Hi)     (Hi)     (Hi)
------------------------------------------------------------------------------

          ------------------------------
0/RP0/CPU0
          RP_TEMP_PCB                          38      -10       -5        0

              80       85       90
          RP_TEMP_HOT_SPOT                     38      -10       -5        0

              80       85       90
          RP_TEMP_LTM4638_0                    38      -10       -5        0

              85       90       95
          RP_TEMP_LTM4644_0                    37      -10       -5        0

              85       90       95
          RP_TEMP_LTM4644_1                    38      -10       -5        0

              85       90       95
          RP_TEMP_LTM4638_1                    37      -10       -5        0

              80       90       95
          RP_TEMP_LTM4644_2                    38      -10       -5        0

              85       90       95
          RP_TEMP_LTM4638_2                    38      -10       -5        0

              80       90       95
          TEMP_CPU_DIE                         39      -10       -5        0
```

```
    80       85       90
TEMP_DDR_DIMM                          39      -10       -5        0

    80       85       90
TEMP_CPU_SSD                           48      -10       -5        0

    70       75       80
TEMP_EITU_SSD                          39      -10       -5        0

    70       75       80
```

The following example shows sample output of the **show environment** command with the **voltage** keyword.

```
RP/0/RP0/CPU0:ios#show environment voltage location 0/rp0/cpu0
```

Location	VOLTAGE Sensor	Value (mV)	Crit (Lo)	Minor (Lo)	Minor (Hi)	Crit (Hi)
0/RP0/CPU0						
	RP_ADM1266_12V0	12035	10800	11280	12720	13200
	RP_ADM1266_1V8_CPU	1801	1670	1750	1850	1930
	RP_ADM1266_1V24_VCCREF	1238	1150	1200	1280	1330
	RP_ADM1266_1V05_CPU	1053	980	1020	1080	1120
	RP_ADM1266_1V2_DDR_VDDQ	1205	1120	1160	1240	1280
	RP_ADM1266_1V0_VCC_RAM	1123	650	700	1250	1300
	RP_ADM1266_1V0_VNN	946	550	600	1250	1300
	RP_ADM1266_1V0_VCCP	704	450	500	1250	1300
	RP_ADM1266_0V6_DDR_VTT	600	560	580	620	640
	RP_ADM1266_12V0_DB	12028	10800	11280	12720	13200
	RP_ADM1266_3V3_STAND_BY_DB	3302	3069	3201	3399	3531
	RP_ADM1266_3V3_STAND_BY	3306	3070	3200	3400	3530
	RP_ADM1266_5V0_DB	5000	4650	4850	5150	5350
	RP_ADM1266_3V3_DB	3328	3069	3201	3399	3531
	RP_ADM1266_2V5_DB	2507	2325	2425	2575	2675
	RP_ADM1266_1V8_DB	1804	1674	1746	1854	1926
	RP_ADM1266_1V0_PHY	997	930	970	1030	1070
	RP_ADM1266_5V0	5048	4650	4850	5150	5350
	RP_ADM1266_3V3	3330	3070	3200	3400	3530
	RP_ADM1266_2V5_PLL	2516	2330	2430	2580	2680
	RP_ADM1266_2V5_FPGA	2505	2330	2430	2580	2680
	RP_ADM1266_1V2_FPGA	1196	1120	1160	1240	1280
	RP_ADM1266_3V3_CPU	3332	3070	3200	3400	3530
	RP_ADM1266_2V5_CPU	2498	2330	2430	2580	2680

The following example shows sample output of the **show environment** command with the **current** keyword.

```
RP/0/RP0/CPU0:ios#show environment current
```

Location	CURRENT Sensor	Value (mA)
0/RP0/CPU0		
	RP_JMAC_1V0_VCCP_IMON	0
	RP_JMAC_1V0_VNN_IMON	93
	RP_JMAC_1V0_VCC_RAM_IMON	0
	RP_JMAC_1V2_DDR_VDDQ_IMON	156
	RP_CURRMON_LTM4638_0	345
	RP_CURRMON_LTM4644_0	145
	RP_CURRMON_LTM4644_1	250
	RP_CURRMON_LTM4638_1	199

```
          RP_CURRMON_DB                          455
0/0/NXR0
          IMON_CLI                              2979
          IMON_CTLPL                             974
          IMON_MODULE                          11270
          IMON_CDR                              3357
          SA_ADM1275_12V_IMON_LC               18624
0/1/NXR0
          IMON_CTLPL                             887
          IMON_CLI                              4587
          IMON_META0_IN0                         807
          IMON_META0_CORE_IOUT0                 5648
          IMON_META0_CORE_IOUT1                 4570
          IMON_META0_IN2                         669
          IMON_META0_CORE_IOUT2                 3726
          IMON_META0_AVD_IOUT                   5085
          IMON_META1_IN0                         326
          IMON_META1_CORE_IOUT0                 2566
          IMON_META1_CORE_IOUT1                 1578
          IMON_META1_IN2                         650
          IMON_META1_CORE_IOUT2                 3718
          IMON_META1_AVD_IOUT                   4593
          SA_ADM1275_12V_IMON_LC                9433
0/2/NXR0
          IMON_OPTM                              867
          IMON_CTLPL                             512
          SA_ADM1275_12V_IMON_LC                1209
0/3/NXR0
          IMON_CLI                              2867
          IMON_CTLPL                            1017
          IMON_MODULE                          11153
          IMON_CDR                              3457
          SA_ADM1275_12V_IMON_LC               17582
0/Rack
          SA_ADM1275_12V_IMON_CPU               1843
 --More--
```

Verify Management Interface Status

To verify the management interface status, perform the following procedure.

Procedure

show interfaces mgmtEth *instance*

Displays the management interface configuration.

Example:

```
RP/0/RP0/CPU0:ios#show interfaces MgmtEth 0/RP0/CPU0/0
MgmtEth0/RP0/CPU0/0 is up, line protocol is up
  Interface state transitions: 3
  Hardware is Management Ethernet, address is 4014.82ba.d26e (bia 4014.82ba.d26e)
  Internet address is 10.105.57.37/25
  MTU 1514 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 1000Mb/s, CX, link type is autonegotiation
```

```
loopback not set,
Last link flapped 00:09:12
ARP type ARPA, ARP timeout 04:00:00
Last input 00:00:00, output 00:00:00
Last clearing of "show interface" counters never
5 minute input rate 1000 bits/sec, 2 packets/sec
5 minute output rate 5000 bits/sec, 1 packets/sec
   6715 packets input, 640515 bytes, 0 total input drops
   0 drops for unrecognized upper-level protocol
   Received 2213 broadcast packets, 4430 multicast packets
            0 runts, 0 giants, 0 throttles, 0 parity
   0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
   944 packets output, 355004 bytes, 0 total output drops
   Output 94 broadcast packets, 114 multicast packets
   0 output errors, 0 underruns, 0 applique, 0 resets
   0 output buffer failures, 0 output buffers swapped out
   3 carrier transitions
```

In the previous output, the management interface is administratively down.

You can also use the **show interfaces summary** and **show interfaces brief** commands in the Cisco IOS XR EXEC mode to verify the management interface status.

The following example shows sample output from the **show interfaces summary** command.

```
RP/0/RP0/CPU0:ios#show interfaces summary
Interface Type          Total    UP       Down     Admin Down
--------------          -----    --       ----     ----------
ALL TYPES               5        2        0        3
--------------
IFT_ETHERNET            2        1        0        1
IFT_NULL                1        1        0        0
IFT_PTP_ETHERNET        2        0        0        2
```

The following example shows sample output from the **show interfaces brief** command.

```
RP/0/RP0/CPU0:KEPLER_PF6#show interfaces brief

               Intf       Intf        LineP              Encap  MTU        BW
               Name       State       State               Type (byte)    (Kbps)
-------------------------------------------------------------------------------
                Nu0          up          up               Null  1500          0
     Mg0/RP0/CPU0/0          up          up               ARPA  1514    1000000
     Mg0/RP0/CPU0/1  admin-down  admin-down               ARPA  1514    1000000
     PT0/RP0/CPU0/0  admin-down  admin-down               ARPA  1514    1000000
     PT0/RP0/CPU0/1  admin-down  admin-down               ARPA  1514    1000000
```

What to do next

If the management interface is administratively down, perform the following steps:

- Check the Ethernet cable connection.
- Verify the IP configuration of the management interface. For details on configuring the management interface, see Configure Management Interface.
- Verify whether the management interface is in the no shut state using the **show running-config interface mgmtEth** command.

  The following example shows sample output from the **show running-config interface mgmtEth** command.

```
RP/0/RP0/CPU0:ios#show running-config interface mgmtEth 0/RP0/CPU0/0
interface MgmtEth0/RP0/CPU0/0
 ipv4 address 10.105.57.37 255.255.255.128
!
```

Verify Firmware Version

The firmware on various hardware components of NCS 1014 must be compatible with the installed Cisco IOS XR image. Incompatibility may cause the NCS 1014 to malfunction.

To verify the firmware version, perform the following procedure.

Before you begin

Procedure

Step 1 **show hw-module fpd**

Example:

```
RP/0/RP0/CPU0:ios#show hw-module fpdAuto-upgrade:Enabled
Attribute codes: B golden, P protect, S secure, A Anti Theft aware
                                                                          FPD Versions
                                                                         ==============
Location   Card type            HWver FPD device       ATR Status   Running Programd  Reload Loc
-----------------------------------------------------------------------------------------------
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   ADM-DB               CURRENT    2.10    2.10       NOT REQ
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   ADM-MB               CURRENT    2.30    2.30       NOT REQ
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   BIOS             S   CURRENT    4.70    4.70         0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   BIOS-Golden      BS  CURRENT            4.70         0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   CpuFpga          S   CURRENT    1.09    1.09         0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   CpuFpgaGolden    BS  NEED UPGD          1.03         0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   SsdMicron5300    S   CURRENT    0.01    0.01         0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   TamFw            S   CURRENT    9.04    9.04         0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9      0.2   TamFwGolden      BS  CURRENT            9.04         0/RP0
0/PM0      NCS1K4-AC-PSU         0.1   PO-PriMCU            CURRENT    2.04    2.04       NOT REQ
0/PM0      NCS1K4-AC-PSU         0.1   PO-SecMCU            CURRENT    2.06    2.06       NOT REQ
0/PM1      NCS1K4-AC-PSU         0.1   PO-PriMCU            CURRENT    2.04    2.04       NOT REQ
0/PM1      NCS1K4-AC-PSU         0.1   PO-SecMCU            CURRENT    2.06    2.06       NOT REQ
0/0/NXR0   NCS1K4-1.2T-K9        0.1   CpuModFw         S   CURRENT  234.10  234.10       NOT REQ
0/0/NXR0   NCS1K4-1.2T-K9        0.1   OptModFw         S   CURRENT    1.38    1.38       NOT REQ
0/1/NXR0   NCS1K14-2.4T-K9       0.1   CpuModFw         S   CURRENT  234.10  234.10       NOT REQ
0/2/NXR0   NCS1K4-1.2T-K9        0.1   CpuModFw         S   CURRENT  234.10  234.10       NOT REQ
0/2/NXR0   NCS1K4-1.2T-K9        0.1   OptModFw         S   CURRENT    1.38    1.38       NOT REQ
0/3/NXR0   NCS1K4-1.2T-K9        0.1   CpuModFw         S   CURRENT  234.10  234.10       NOT REQ
0/3/NXR0   NCS1K4-1.2T-K9        0.1   OptModFw         S   CURRENT    1.38    1.38       NOT REQ
0/Rack     NCS1014               0.1   ADM-CHASSIS          CURRENT    0.21    0.21       NOT REQ
0/Rack     NCS1014               0.1   IoFpga           S   CURRENT    1.10    1.10       NOT REQ
0/Rack     NCS1014               0.1   IoFpgaGolden     BS  CURRENT            1.05       NOT REQ
0/Rack     NCS1014               0.1   SsdIntelSC2KB    S   CURRENT    1.20    1.20        0/Rack
```

- Status—Upgrade status of the firmware. The different states are:
  - CURRENT—The firmware version is the latest version.
  - NOT READY—The firmware of the FPD is not ready for upgrade.

- NEED UPGD—A newer firmware version is available in the installed image. We recommended that upgrade be performed.
- UPGD PREP—The firmware of the FPD is preparing for upgrade.
- RLOAD REQ—The upgrade is completed, and the card requires a reload.
- UPGD DONE—The firmware upgrade is successful.
- UPGD FAIL—The firmware upgrade has failed.
- UPGD SKIP—The upgrade is skipped because the installed firmware version is higher than the version available in the image.
- Running—Current version of the firmware running on the FPD.

Step 2 **show fpd package**

Use the **show fpd package** command to display the FPD image version available with this software release for each hardware component.

Example:

```
RP/0/RP0/CPU0:ios#show fpd package

================================ ============================================
                                        Field Programmable Device Package
                                 ============================================
                                           Req     SW      Min Req   Min Req
Card Type            FPD Description       Reload  Ver     SW Ver  Board Ver
================== ========================= ====== ======= ======== =========
--------------------------------------------------------------------------------
NCS1014-SA           ADM-CHASSIS               NO      0.21      0.21      0.0
                     IoFpga                    NO      1.10      1.10      0.0
                     IoFpgaGolden              NO      1.05      1.05      0.0
                     SsdIntelSC2KB             YES     1.20      1.20      0.0
--------------------------------------------------------------------------------
NCS1K14-2.4T-K9      CpuModFw                  NO    234.10    234.10      0.0
--------------------------------------------------------------------------------
NCS1K14-2.4T-L-K9    CpuModFw                  NO    234.10    234.10      0.0
--------------------------------------------------------------------------------
NCS1K14-CCMD-16-C    CpuModFw                  NO    234.10    234.10      0.0
                     OptModFw                  NO     15.01     15.01      0.0
--------------------------------------------------------------------------------
NCS1K14-CCMD-16-L    CpuModFw                  NO    234.10    234.10      0.0
                     OptModFw                  NO     15.01     15.01      0.0
--------------------------------------------------------------------------------
NCS1K14-CNTLR-K9     ADM-DB                    NO      2.10      2.10      0.2
                     ADM-MB                    NO      2.30      2.30      0.2
                     BIOS                      YES     4.70      4.70      0.0
                     BIOS-Golden               YES     4.70      0.01      0.0
                     CpuFpga                   YES     1.09      1.09      0.0
                     CpuFpgaGolden             YES     1.09      1.09      0.0
                     SsdIntelS4510             YES    11.32     11.32      0.0
                     SsdIntelSC2KB             YES     1.20      1.20      0.0
                     SsdMicron5300             YES     0.01      0.01      0.0
                     TamFw                     YES     9.04      9.04      0.0
                     TamFwGolden               YES     9.04      9.04      0.0
--------------------------------------------------------------------------------
NCS1K14-CTLR-B-K9    ADM-DB                    NO      2.10      2.10      0.2
                     ADM-MB                    NO      2.30      2.30      0.2
                     BIOS                      YES     4.70      4.70      0.0
```

```
                    BIOS-Golden                YES     4.70      0.01      0.0
                    CpuFpga                    YES     1.09      1.09      0.0
                    CpuFpgaGolden              YES     1.09      1.09      0.0
                    SsdIntelS4510              YES    11.32     11.32      0.0
                    SsdIntelSC2KB              YES     1.20      1.20      0.0
                    SsdMicron5300              YES     0.01      0.01      0.0
                    TamFw                      YES     9.04      9.04      0.0
                    TamFwGolden                YES     9.04      9.04      0.0
--------------------------------------------------------------------------------
NCS1K4-1.2T-K9      CpuModFw                   NO    234.10    234.10      0.0
                    OptModFw                   NO      1.38      1.38      0.0
--------------------------------------------------------------------------------
NCS1K4-AC-PSU       PO-PriMCU                  NO      2.04      2.04      0.1
                    PO-SecMCU                  NO      2.06      2.06      0.1
--------------------------------------------------------------------------------
NCS1K4-AC-PSU-2     PO-PriMCU                  NO      1.03      1.03      0.1
                    PO-SecMCU                  NO      1.05      1.05      0.1
```

What to do next

Upgrade all the FPDs using the **upgrade hw-module location all fpd all** command in the Cisco IOS XR EXEC mode. After upgrade is completed, the Status column shows RLOAD REQ if the software requires reload.

If Reload is required

If the FPGA location is 0/RP0, use the **admin hw-module location 0/RP0 reload** command. This command reboots only the CPU. As a result, traffic is not impacted. If the FPGA location is 0/0, use the **admin hw-module location all reload** command. This command reboots the chassis. As a result, traffic is impacted. After the reload is completed, the new FPGA runs the current version.

If Firmware Upgrade Fails

If firmware upgrade fails, use the **show logging** command to view the details and upgrade the firmware again using the above commands.

Note You can upgrade the firmware version of power modules, only when both the power modules are present and powered on.

Verify Alarms

You can view the alarm information using the **show alarms** command.

Procedure

show alarms [**brief** [**card | rack | system**] [**location** *location*] [**active | history**] | **detail** [**card | rack | system**] [**location** *location*] [**active | clients | history | stats**]]

Displays alarms in brief or detail.

Example:

```
RP/0/RP0/CPU0:ios#show alarms brief card location 0/RP0/CPU0 active

------------------------------------------------------------------------------------
Active Alarms
------------------------------------------------------------------------------------
Location        Severity     Group           Set Time                    Description


------------------------------------------------------------------------------------
0/0             Major        Ethernet        11/21/2023 11:11:35 UTC    HundredGigECtrlr0/0/0/2 -
Carrier Loss On The LAN

0/3             Major        Ethernet        11/21/2023 11:11:37 UTC    HundredGigECtrlr0/3/0/2 -
Remote Fault

0/0             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/0/0/4 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/3/0/3 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/3/0/4 -
Local Fault

0/0             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/0/0/5 -
Local Fault

0/0             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/0/0/6 -
Local Fault

0/0             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/0/0/7 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:39 UTC    HundredGigECtrlr0/3/0/5 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:39 UTC    HundredGigECtrlr0/3/0/6 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:38 UTC    HundredGigECtrlr0/3/0/7 -
Local Fault

0/0             Major        Ethernet        11/21/2023 11:11:43 UTC    HundredGigECtrlr0/0/0/3 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:57 UTC    HundredGigECtrlr0/3/0/8 -
Local Fault

0/3             Major        Ethernet        11/21/2023 11:11:57 UTC    HundredGigECtrlr0/3/0/9 -
```

```
Local Fault

0/3             Major        Ethernet         11/21/2023 11:11:59 UTC    HundredGigECtrlr0/3/0/12 -
 Local Fault

0/3             Major        Ethernet         11/21/2023 11:11:59 UTC    HundredGigECtrlr0/3/0/13 -
 Local Fault

0/0             Major        Ethernet         11/21/2023 11:12:03 UTC    HundredGigECtrlr0/0/0/9 -
Local Fault

0/0             Major        Ethernet         11/21/2023 11:12:04 UTC    HundredGigECtrlr0/0/0/8 -
Local Fault

0/0             Major        Ethernet         11/21/2023 11:12:04 UTC    HundredGigECtrlr0/0/0/10 -
 Local Fault

0/0             Major        Ethernet         11/21/2023 11:12:04 UTC    HundredGigECtrlr0/0/0/11 -
 Local Fault

0/0             Major        Ethernet         11/21/2023 11:12:04 UTC    HundredGigECtrlr0/0/0/12 -
 Local Fault

0/0             Major        Ethernet         11/21/2023 11:12:04 UTC    HundredGigECtrlr0/0/0/13 -
 Local Fault

0/3             Major        Ethernet         11/21/2023 11:12:04 UTC    HundredGigECtrlr0/3/0/11 -
 Local Fault

0/3             Major        Ethernet         11/21/2023 11:12:05 UTC    HundredGigECtrlr0/3/0/10 -
 Local Fault
```

Note
In the maintenance mode, all the alarms are suppressed and the **show alarms** command will not show the alarms details. Use the **show controllers** *controllertype R/S/I/P* command to view the client and trunk alarms.

Verify Context

The **show context** command displays core dump context information of NCS 1014.

Procedure

show context

When you excecute the **show context** command in Cisco IOS XR EXEC mode, the output displays the core dump context information of any process on the NCS 1014 as well as up to 10 last instances.

Example:

```
RP/0/RP0/CPU0:ios#sh context

node: node0_RP0_CPU0
Context number: 1
--------------------------------------------------------------
Core location: 0/RP0/CPU0:/misc/disk1

Core for pid = 6232 (Terminal_Device)
Core for process: opt_terminal_device_6232.by.11.20231204-170249.node0_RP0_CPU0.877b9.core.gz
Core dump time: 2023-12-04 17:02:50.144240146 +0000

Process:
Core was generated by `opt_terminal_device'.

Build information:
### XR Information

User = deenayak
Host = iox-ucs-061
Workspace = /auto/iox-ucs-061-san1/prod/24.1.1.32I.SIT_IMAGE/ncs1010/ws/
Built on = Fri Nov 17 17:17:31 UTC 2023
Lineup = r241x.lu%EFR-00000453356
XR version = 24.1.1.32I

### Leaba Information


Platform information:
card_product_id: NCS1014
platform: ncs1010

Signal information:
Program terminated with signal 11, Segmentation fault.

Faulting thread: 6232

Registers for Thread 6232
rax:    0x7f2d0d7be000
rbx:    0x0
rcx:    0x0
rdx:    0x7f
rsi:    0x0
rdi:    0x7fff2f3d3420
rbp:    0x0
rsp:    0x7fff2f3d3410
r8:     0x0
r9:     0x7fff2f3d3510
r10:    0xffffffffffffff80
r11:    0x5287
r12:    0x7f
r13:    0x7fff2f3d3420
r14:    0x7f2d0dac4684
r15:    0x7fff2f3d3598
rip:    0x7f2d0d6701ca
eflags: 0x10206
cs:     0x33
ss:     0x2b
ds:     0x0
es:     0x0
```

```
fs:     0x0
gs:     0x0


Backtrace for Thread 6232
#0  0x00007f2d0d6701ca in ?? () from /lib64/libc-2.31.so
#1  0x00007f2d0d64b4d5 in snprintf+0x85 from /lib64/libc-2.31.so
#2  0x00007f2d0dadf5cf in ?? () from /opt/cisco/install-iosxr/base/lib/libopenconfig_cmn.so
#3  0x00007f2d0dade7a2 in ?? () from /opt/cisco/install-iosxr/base/lib/libopenconfig_cmn.so
#4  0x00007f2d0dfd6b35 in ?? () from
/opt/cisco/install-iosxr/base/lib/libinfra_sysdb_combine_82eb6a4d2fa15d0e.so
#5  0x00007f2d0dfd4b0e in sysdb_process_pending_pulse+0x512 from
/opt/cisco/install-iosxr/base/lib/libinfra_sysdb_combine_82eb6a4d2fa15d0e.so
#6  0x00007f2d0e11e3bd in ?? () from
/opt/cisco/install-iosxr/base/lib/libinfra_combine_82eb6a4d2fa15d0e.so
#7  0x00007f2d0e12831e in xr_event_dispatch+0x48 from
/opt/cisco/install-iosxr/base/lib/libinfra_combine_82eb6a4d2fa15d0e.so
#8  0x00005607924be6a9 in ?? ()
#9  0x00007f2d0d61cd1b in __libc_start_main+0xeb from /lib64/libc-2.31.so
#10 0x00005607924be31a in ?? ()

------------------------------------------------------------------

node: node0_RP0_CPU0
Context number: 2
------------------------------------------------------------------
Core location: 0/RP0/CPU0:/misc/disk1

Core for pid = 5155 (sh_proc_mem_edm)
Core for process: sh_proc_mem_edm_5155.by.user.20231204-105935.node0_RP0_CPU0.4b884.core.gz
Core Dump time: Mon Dec  4 10:59:35 2023

Process:
Core was generated by: user requested dump of pid 5155

Build information:
### XR Information

User = deenayak
Host = iox-ucs-061
Workspace = /auto/iox-ucs-061-san1/prod/24.1.1.32I.SIT_IMAGE/ncs1010/ws/
Built on = Fri Nov 17 17:17:31 UTC 2023
Lineup = r241x.lu%EFR-00000453356
XR version = 24.1.1.32I

### Leaba Information


Registers for Thread (LWP 5155)
rax:    0xfffffffffffffffc
rbx:    0x5570ec6edd60
rcx:    0x7f0239431cd6
rdx:    0x20
rsi:    0x5570ec6ee060
rdi:    0x1b
rbp:    0x7ffca2a8b690
rsp:    0x7ffca2a8b4e0
r8:     0x0
r9:     0x436
r10:    0xffffffff
r11:    0x293
r12:    0x5570ec6ee020
r13:    0x5570ec6ede70
r14:    0x5570ec6ee060
```

```
r15:    0x5570ec6edd60
rip:    0x7f0239431cd6
eflags: 0x293
cs:     0x33
ss:     0x2b
ds:     0x0
es:     0x0
fs:     0x0
gs:     0x0


Backtrace for Thread (LWP 5155)
#0  0x00007f0239431cd6 in ?? () from /lib64/libc-2.31.so
#1  0x00007f0238fefd2a in event_del_nolock_+0x3a from /usr/lib64/libevent-2.1.so.7.0.0
#2  0x00007f0238fe3dbe in ?? () from /usr/lib64/libevent-2.1.so.7.0.0
#3  0x00007f0239799034 in event_block+0x204 from
/opt/cisco/install-iosxr/base/lib/libinfra_combine_82eb6a4d2fa15d0e.so
#4  0x00005570ec41fba8 in ?? () from /opt/cisco/install-iosxr/base/bin/sh_proc_mem_edm
#5  0x00007f023935ed1b in ?? () from /lib64/libc-2.31.so
#6  0x00005570ec41f8fa in ?? () from /opt/cisco/install-iosxr/base/bin/sh_proc_mem_edm

------------------------------------------------------------------

node: node0_RP0_CPU0
Context number: 3
------------------------------------------------------------------
Core location: 0/RP0/CPU0:/misc/disk1

Core for pid = 4316 (sysdb_mc_main)
Core for process: sysdb_mc_4316.by.user.20231203-161922.node0_RP0_CPU0.3f09d.core.gz
Core Dump time: Sun Dec  3 16:19:22 2023

Process:
Core was generated by: user requested dump of pid 4316

Build information:
### XR Information

User = deenayak
Host = iox-ucs-061
Workspace = /auto/iox-ucs-061-san1/prod/24.1.1.32I.SIT_IMAGE/ncs1010/ws/
Built on = Fri Nov 17 17:17:31 UTC 2023
Lineup = r241x.lu%EFR-00000453356
XR version = 24.1.1.32I

### Leaba Information


Registers for Thread (LWP 4316)
rax:    0xfffffffffffffffc
rbx:    0x0
rcx:    0x7f46904eca92
rdx:    0x0
rsi:    0x0
rdi:    0x7ffd281477c0
rbp:    0x7ffd28147a00
rsp:    0x7ffd281476e0
r8:     0x0
r9:     0x0
r10:    0x8
r11:    0x293
r12:    0x1
r13:    0x0
r14:    0x0
```

```
r15:    0x7ffd281477c0
rip:    0x7f46904eca92
eflags: 0x293
cs:     0x33
ss:     0x2b
ds:     0x0
es:     0x0
fs:     0x0
gs:     0x0


Backtrace for Thread (LWP 4316)
#0  0x00007f46904eca92 in ?? () from /lib64/libc-2.31.so
#1  0x00005583860ea640 in ?? () from /opt/cisco/install-iosxr/base/sbin/sysdb_mc
#2  0x00005583860bbbd1 in ?? () from /opt/cisco/install-iosxr/base/sbin/sysdb_mc
#3  0x00007f46904d7d1b in ?? () from /lib64/libc-2.31.so
#4  0x00005583860bbada in ?? () from /opt/cisco/install-iosxr/base/sbin/sysdb_mc

----------------------------------------------------------------

node: node0_RP0_CPU0
Context number: 4
----------------------------------------------------------------
Core location: 0/RP0/CPU0:/misc/disk1

Core for pid = 4212 (sysdb_svr_local)
Core for process: sysdb_svr_local_4212.by.user.20231203-161920.node0_RP0_CPU0.bf2d1.core.gz
Core Dump time: Sun Dec  3 16:19:20 2023

Process:
Core was generated by: user requested dump of pid 4212

Build information:
### XR Information

User = deenayak
Host = iox-ucs-061
Workspace = /auto/iox-ucs-061-san1/prod/24.1.1.32I.SIT_IMAGE/ncs1010/ws/
Built on = Fri Nov 17 17:17:31 UTC 2023
Lineup = r241x.lu%EFR-00000453356
XR version = 24.1.1.32I

### Leaba Information


Registers for Thread (LWP 4212)
rax:    0xfffffffffffffffc
rbx:    0x0
rcx:    0x7f1fcb1c4a92
rdx:    0x0
rsi:    0x0
rdi:    0x7fff60339f50
rbp:    0x7fff6033a200
rsp:    0x7fff60339e80
r8:     0x0
r9:     0x0
r10:    0x8
r11:    0x293
r12:    0x0
r13:    0x0
r14:    0x7fff60339f50
r15:    0x7f1fcb50da9e
rip:    0x7f1fcb1c4a92
eflags: 0x293
```

```
cs:     0x33
ss:     0x2b
ds:     0x0
es:     0x0
fs:     0x0
gs:     0x0


Backtrace for Thread (LWP 4212)
#0  0x00007f1fcb1c4a92 in ?? () from /lib64/libc-2.31.so
#1  0x00007f1fcb5af472 in sysdb_svr_main+0xd15 from
/opt/cisco/install-iosxr/base/lib/libsysdbsvr_only.so
#2  0x000055e2fd529851 in ?? () from /opt/cisco/install-iosxr/base/sbin/sysdb_svr_local
#3  0x00007f1fcb1afd1b in ?? () from /lib64/libc-2.31.so
#4  0x000055e2fd52975a in ?? () from /opt/cisco/install-iosxr/base/sbin/sysdb_svr_local

----------------------------------------------------------------
```

Verify Core Files

The **dir harddisk:/*core.gz** command checks for core files of NCS 1014.

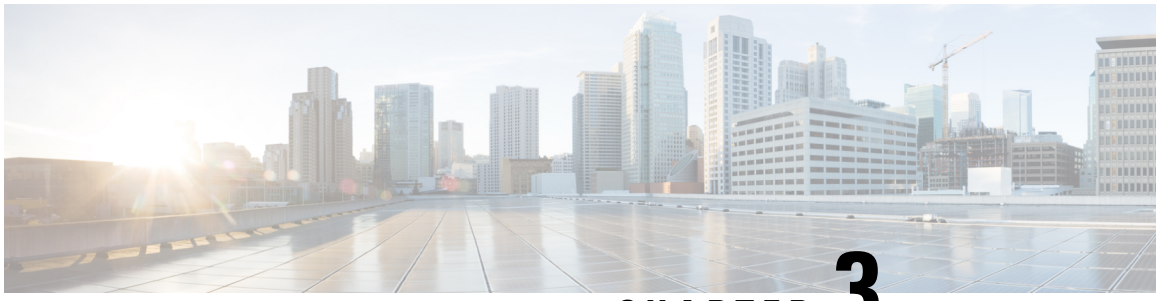
Procedure

dir harddisk:/*core.gz

Example:

```
RP/0/RP0/CPU0:ios#dir harddisk:/*core.gz
Wed Dec 6 04:54:16.336 UTC

Directory of harddisk:/*core.gz
2476 -rw-r--r--. 1 8120038 Oct 30 15:08
cma_server_41264.by.6.20231030-150817.node0_RP0_CPU0.502a7.core.gz
```



CHAPTER 3

Perform System Upgrade and Install Feature Packages

You can execute the system upgrade and package installation processes using the **install** commands on NCS 1014. The processes involve adding and activating the ISO images (*.iso*) and feature packages (*.rpm*) on NCS 1014. You can accesss these files from a network server and then activate on NCS 1014. If the installed package or SMU causes any issue, you can uninstall it.

Note The output of the examples in the procedures is not from the latest software release. The output will change for any explicit references to the current release.

Note We recommend that you collect the output of **show tech-support ncs1014** command before performing operations such as a reload or CPU OIR on NCS 1014. The command provides information about the state of the system before reload or before the CPU OIR operation is performed. This information is useful in debugging.

- Upgrade Software, on page 55
- View Supported Software Upgrade or Downgrade Versions, on page 58
- Software Upgrade and Downgrade Matrix , on page 61
- Install Packages and RPMs, on page 62
- Upgrade FPD, on page 66
- Verify if an FPD Upgrade is Required, on page 70
- Manual FPD Upgrade, on page 72
- Automatic FPD upgrade, on page 73

Upgrade Software

Upgrading the software is the process of installing a new version of the Cisco IOS XR operating system on NCS 1014. NCS 1014 is preinstalled with the Cisco IOS XR image. However, you can install a new version to keep features up to date. You can perform the software upgrade operation using an ISO image from the XR mode.

Note NCS1014 and NCS1014 platform uses the same IOS-XR packaging image. Nomenclature of ISO image of IOS-XR base image example: "ncs1010-x64-[sw-rel-ver].iso'.

Note Upgrading from R7.11.1 to either R24.2.1 or R24.3.1 will raise the *DISASTER_RECOVERY_UNAVAILABLE_ALARM*. Postupgrade, this alarm clears automatically. For more information on the alarm, see *Troubleshooting Guide for Cisco NCS 1014*.

Before you begin

- Configure Management Interface
- Copy the ISO image to be installed either on the NCS 1014 hard disk or on a network server to which NCS 1014 has access.

Procedure

Step 1 Execute one of these commands:

Installs the new ISO image from the harddisk or from the network server. The install operation takes 20–40 minutes to complete.

- **install replace /harddisk:/***iso-image-name*
- **install package replace** *<ftp or http or https protocol>/package_path/ filename1 filename2 ...*

Note
The **install package replace** command upgrades the ISO image but doesn't reload the RP automatically. But the **install replace** command upgrades the ISO image and reloads the RP.

Example:

```
RP/0/RP0/CPU0:ios#install replace /harddisk:/ncs1010-x64-7.11.1.iso
Wed Nov 15 09:44:44.491 UTC
Once the packaging dependencies have been determined, the install operation may have to reload the
system.
If you want to control the timing of system reload, you must not continue, but use the 'install
package replace' command instead, followed by 'install apply'.
Continue? [yes/no]:[yes]
Install replace operation 1.1 has started
Install operation will continue in the background
.....................
.....................
ios con0/RP0/CPU0 is now available
```

Note
Boot time FPD upgrade happens before XR boot. All the FPDs belonging to the RP location are upgraded during the boot time FPD upgrade.

Note

Automatic Field Programmable Device(FPD) upgrade is enabled by default.. When the automatic FPD upgrade is enabled, the install operation also upgrades the FPDs (except the Golden FPDs and Power modules) that need to be upgraded.

Step 2 **show install request**

Displays the status of the install operation.

Example:

```
RP/0/RP0/CPU0:ios#show install request
Wed Nov 15 10:00:35.713 UTC
User request: install replace /harddisk:/ncs1010-golden-x86_64-7.11.1.48I-Weekly.iso
Operation ID: 1.1
State:In progress since 2023-11-15 09:50:23 UTC
Current activity:    Package add or other package operation
Next activity:       Apply
Time started:        2023-11-15 09:55:24 UTC
Timeout in:          84m 43s
Locations responded: 0/1
Location          Packaging operation stage Notification Phase Clients responded
----------------- ------------------------- ------------------ -----------------
0/RP0/CPU0        Package operations        None in progress                N/A
```

When the install operation completes successfully, the device automatically reloads.

Note
In case of the **install package replace** command, you'll be prompted to enter the next command (**install apply reload** command).

Step 3 **install commit**

Commits the new ISO image.

Example:

```
RP/0/RP0/CPU0:ios#install commit
Wed Nov 15 10:38:00.592 UTC
Install commit operation 1 has started
Install operation will continue in the background
```

Note
It is the mandatory to commit the install successfully to upgrade the software, missing this step followed by any controller reload/restart/power cycle will result in rollback to previously installed committed software/RPM package version.

The *DISASTER_RECOVERY_UNAVAILABLE_ALARM* clears upon completion of the upgrade from R7.11.1 to R24.2.1 or R24.3.1.

Step 4 **show install committed**

Displays the committed package information.

Example:

```
RP/0/RP0/CPU0:ios#show install committed
Wed Nov 15 10:41:20.454 UTC
Software Hash: ec69dcceb81c0da69b297aa7de1d00f56b8aef52403c5e0ffe6e5db098bd83b8
Package                                                                      Version
-----------------------------------------------------------------------------------
xr-aaa                                                                 7.11.1.48Iv1.0.0-1
xr-acl                                                                 7.11.1.48Iv1.0.0-1
xr-apphosting                                                          7.11.1.48Iv1.0.0-1
xr-appmgr                                                              7.11.1.48Iv1.0.0-1
xr-bcdl                                                                7.11.1.48Iv1.0.0-1
xr-bfd                                                                 7.11.1.48Iv1.0.0-1
```

```
xr-bgp                                                    7.11.1.48Iv1.0.0-1
xr-bgputil                                                7.11.1.48Iv1.0.0-1
xr-bng-stubs                                              7.11.1.48Iv1.0.0-1
xr-bundles                                                7.11.1.48Iv1.0.0-1
```

View Supported Software Upgrade or Downgrade Versions

Your Cisco chassis comes preinstalled with IOS XR software. You either upgrade the software release to use new features and software fixes, or you downgrade the software. To leverage new features that are added or software fixes that are provided, it is important that you upgrade your software to a current version.

To help you select a Cisco IOS XR software release that aligns with Cisco-certified upgrade and downgrade paths, this feature provides answers to the following questions:

- What upgrade or downgrade releases are supported for the current release?
- I plan to upgrade from Release X to Release Y. Does my chassis support upgrade to Release Y?
- Are there any bridging SMUs that must be installed before I upgrade the software?

This feature provides a mechanism to determine whether the current release supports an upgrade to a target release. This task is run at the start of a software upgrade or downgrade through the **install replace** command. If the validation fails, the software upgrade is blocked, and the system notifies the reason for the failure. This feature allows you to proactively examine whether you can upgrade or downgrade to a certain release, saving time and effort involved in planning and upgrading the software.

The feature provides the following information to help you understand the prerequisites or limitations related to the specific software upgrade or downgrade:

- Required bridging SMU RPMs
- Blocking SMU RPMs
- Unsupported hardware
- Caveats or restrictions

You can overwrite the automatic validation using the **force** keyword in the **install replace** command. With this option, the system displays warning messages when the upgrade fails but does not block the software upgrade. Use the **force ?** keyword to understand any other impact to system functionalities apart from the disabling of this process that determines the supported releases for software upgrade or downgrade.

You can view the support information using the following **show** commands or through the operational data.

Command	Description
show install upgrade-matrix running	Displays all supported software upgrades from the current version according to the support data installed on the running system
show install upgrade-matrix iso *path-to-ISO*	Displays details about the software upgrade from the current version to the version of the target ISO

Command	Description
	according to the support data in both the running system and the ISO image
show install upgrade-matrix iso *path-to-ISO* **all**	Displays all supported software upgrades from any version according to the support data in the target ISO image
show install upgrade-matrix iso *path-to-ISO* **from-running**	Displays details about the software upgrade from the current version to the version of ISO according to the support matrices in both the running system and the target ISO image

View All Supported Software Upgrade from Running Version

The following example shows all supported releases for upgrade from the current version 24.1.1 on the chassis:

```
RP/0/RP0/CPU0:ios#show install upgrade-matrix running
Fri Mar 15 12:53:23.715 IST
Matrix: XR version: 24.1.1, File version: 1.0, Version: N/A

The upgrade matrix indicates that the following system upgrades are supported from the
current XR version:

From       To         Restrictions
---------- ---------- ------------
24.1.1     7.11.1     -

Add the from and to versions to the end of the CLI command, for data on versions with
additional restrictions

For example, to display restrictions for the 24.1.1->7.11.1 upgrade, use
        'show install upgrade-matrix running 24.1.1 7.11.1'
```

Pre and Post-Upgrade Installation Health Checks

This section describes about of the pre and postupgrade Installation health check for routers.

Existing client-server framework notifies the subscribed clients to perform the precheck functionality.

The System health check infrastructure that is plugged to the install pre and postchecks phase of the system upgrade. This includes other existing install pre or postchecks.

Upgrade precheck:

- If single command upgrade is triggered either with a force option or is configured to skip checks, then health check is bypassed and a syslog entry added.
- When single command upgrade is triggered, install infra performs install specific prechecks. If the install prechecks pass, the system health check infra plug-in is invoked to check the overall system health.
- The health check infrastructure returns the health status during the installation.
- Single command upgrade continues on if the prechecks completes with no errors.

- If any errors are detected, then single command upgrade continues or terminates depending on the option that is selected for abort-on-precheck-failure.
- Single command upgrade postchecks before autocommit triggers based on the user selected level information.

Upgrade post check:

- Post checks are bypassed if force or config option is selected for single command upgrade.
- If install specific postchecks are completed sucessfully, then the system health check infra plug-in is invoked. If no errors are reported then the autocommit triggers.
- If any errors are detected, the abort-on option that is saved before the upgrade reload is used to either abort the single command upgrade or continue. This depends on the severity of the errors that are detected during post check.
- Summary of the pre and posthealth check is appended to the single command upgrade operation log.

Installation Profile Creation

Installation Profile is created to choose and alternate installation behavior. One default profile is created involving pre and postchecks. You can edit the install behavior to choose cases like terminate installation if precheck fails or revert after post installation check. You can also choose to continue installation despite failure in pre checks.

You can configure “enable or disable” options to run pre or post installation checks or “abort-on-failure” for pre checks, or "warn-on-failure" and “restore-to-v1” on post checks. To configure the Install profile, use the following commands:

config

install profile *profile_name* **pre-check***metric-name* **[enable | disable] [abort-on-failure | continue-on-failure | revert-on-failure]**

end

Following is a sample to display metric settings in the install profile.

```
RP/0/RP0/CPU0:ios#show install profile default
Fri Mar 15 11:29:35.381 IST
Profile Name : default
State : Enabled

Prechecks : Enabled
        communication-timeout   : Enabled       [ warn-on-failure ]
        config-inconsistency    : Enabled       [ error-on-failure ]
        process-resource        : Enabled       [ warn-on-failure ]
        process-status          : Enabled       [ warn-on-failure ]
        system-clock            : Enabled       [ warn-on-failure ]
        hw-monitoring           : Enabled       [ warn-on-failure ]
        lc-monitoring           : Enabled       [ warn-on-failure ]
        pci-monitoring          : Enabled       [ warn-on-failure ]
        wd-monitoring           : Enabled       [ warn-on-failure ]
        disk-space              : Enabled       [ error-on-failure ]
        upgrade_matrix          : Enabled       [ error-on-failure ]
        core-cleanup            : Disabled      [ NA ]
        file-cleanup            : Disabled      [ NA ]

Postchecks : Enabled
```

```
communication-timeout   : Enabled       [ error-on-failure ]
config-inconsistency    : Enabled       [ error-on-failure ]
process-resource        : Enabled       [ error-on-failure ]
process-status          : Enabled       [ error-on-failure ]
system-clock            : Enabled       [ error-on-failure ]
hw-monitoring           : Enabled       [ error-on-failure ]
lc-monitoring           : Enabled       [ error-on-failure ]
pci-monitoring          : Enabled       [ error-on-failure ]
wd-monitoring           : Enabled       [ error-on-failure ]
```

Use the following configuration to report health check:

config

grpc local-connection

Netconf-yang agent

commit

The following is a sample to display health check states:

```
RP/0/RP0/CPU0:ios#show healthcheck internal states
Fri Mar 15 12:55:54.739 IST

 Internal Structure INFO

 Current state: Disabled

 Reason: Success

 Netconf Config State: Enabled

 Grpc Config State: Disabled

 Nosi state: Not ready

 Appmgr conn state: Invalid

 Nosi lib state: Not ready

 Nosi client: Valid client
```

Software Upgrade and Downgrade Matrix

The following table lists the upgrade and downgrade paths supported for Cisco NCS 1014.

Upgrade Path			Downgrade Path		
Source Release	**Destination Release**	**Bridge SMU**	**Source Release**	**Destination Release**	**Bridge SMU**
• R7.11.1 • R24.1.1 • R24.2.11 • R24.3.1	R24.4.1 R24.4.10	No	R24.4.1 R24.4.10	• R24.3.1 • R24.2.11 • R24.1.1 • R7.11.1	No

Note

- Downgrading the software from version 24.4.1 to a lower version with loopback enabled is not supported and will affect traffic if attempted.
- Before upgrading to R24.4.1 or a later version, you must manually configure the wavelength or frequency to ensure a non-traffic-impacting software upgrade.

Install Packages and RPMs

Complete this task to install additional packages or rpm files. The rpm files that need to be installed must be placed in a folder.

Note This task can be used to install SMUs as well.

Before you begin

- Configure and connect to the management interface. You can access the installable file through the management interface. For details about configuring the management interface, see Workflow for Install Process.
- Copy the package or rpm to be installed either on the NCS 1014 hard disk or on a network server to which NCS 1014 has access.

Procedure

Step 1 **install package add source /harddisk:/** *iso-image-name or rpm-folder-name*

Example:

```
RP/0/RP0/CPU0:ios#install package add source harddisk:/rpm
Wed Nov 15 18:10:14.784 UTC

Install add operation 2.1.2 has started
Install operation will continue in the background
```

```
RP/0/RP0/CPU0:ios#install package add source harddisk:/rpm/
Thu Apr 20 18:09:49.582 UTC
Install add operation 7.1.1 has started
Install operation will continue in the background
```

Ensure to add the respective packages or rpm files as appropriate. This operation may take time depending on the size of the files that are added. The operation takes place in an asynchronous mode. The **install package add source** command runs in the background, and the EXEC prompt is returned.

Step 2 **show install request**

Example:

```
RP/0/RP0/CPU0:ios#show install request

Thu Apr 20 18:13:00.720 UTC

User request: install package add source file:///harddisk:/rpm
Operation ID: 7.1.1
State:        Success since 2023-04-20 18:13:04 UTC

Current activity:    Await user input
Time started:        2023-04-20 18:13:04 UTC

The following actions are available:
    install package add
    install package remove
    install package upgrade
    install package downgrade
    install package abort latest
    install package abort all-since-apply
    install apply restart
    install apply reload
    install replace reimage

Least impactful apply method: install apply restart
```

Displays the current status of the install operation.

Step 3 **install apply reload**

Example:

```
RP/0/RP0/CPU0:ios#install apply

Thu Apr 20 18:13:18.514 UTC
Once the packaging dependencies have been determined, the install operation may have to reload the
system.
If you want more control of the operation, then explicitly use 'install apply restart' or 'install
apply reload' as reported by 'show install request'.
Continue? [yes/no]:[yes] yes
Install apply operation 7.1 has started
Install operation will continue in the background
```

Enables NCS 1014 to reload.

Step 4 **show install request**

Example:

```
RP/0/RP0/CPU0:ios#show install request
Thu Apr 20 18:15:06.876 UTC

User request: install apply restart
Operation ID: 7.1
State:        Success since 2023-04-20 18:14:41 UTC

Current activity:    Await user input
Time started:        2023-04-20 18:14:41 UTC

The following actions are available:
install package add
install package remove
install package upgrade
install package downgrade
install package replace
install package rollback
install replace
```

```
install rollback
install source
install commit
install replace reimage
```

Displays the current status of the install operation.

Step 5 **install commit**

Example:

```
RP/0/RP0/CPU0:ios#install commit
Thu Apr 20 18:15:17.620 UTC
Install commit operation 7 has started
Install operation will continue in the background
```

Commits the package or rpm files.

Step 6 **show install request**

Example:

```
RP/0/RP0/CPU0:ios#show install request

User request: install commit
Operation ID: 2
State:        In progress since 2022-07-04 11:48:48 UTC

Current activity:    Commit transaction
Next activity:       Transaction complete
Time started:        2022-07-04 11:48:48 UTC

No per-location information.
```

Displays the current status of the install operation. The above output indicates that the install operation is in progress.

Example:

```
RP/0/RP0/CPU0:ios#show install request

User request: install commit
Operation ID: 2
State:        Success since 2022-07-04 11:50:32 UTC

Current activity:    No install operation in progress

The following actions are available:
    install package add
    install package remove
    install package upgrade
    install package downgrade
    install package replace
    install package rollback
    install replace
    install rollback
    install source
```

Displays the current status of the install operation. The above output indicates that the install operation is complete.

Step 7 **show install request**

Example:

```
RP/0/RP0/CPU0:ios#show install request

User request: install commit
Operation ID: 2
```

```
State:        Success since 2022-07-04 11:50:32 UTC

Current activity:    No install operation in progress

The following actions are available:
    install package add
    install package remove
    install package upgrade
    install package downgrade
    install package replace
    install package rollback
    install replace
    install rollback
    install source
```

Displays the current status of the install operation. The above output indicates that the install operation is complete.

Step 8 **show install active summary**

Example:

```
RP/0/RP0/CPU0:ios#show install active summary
Wed Nov 15 18:20:38.783 UTC
Active Packages: XR: 160 All: 1318
Label: 7.11.1.48I-Weekly
Software Hash: ec69dcceb81c0da69b297aa7de1d00f56b8aef52403c5e0ffe6e5db098bd83b8

Optional Packages                                   Version
--------------------------------------------------- ----------------------------
xr-bgp 7.11.1.48I                                   v1.0.0-1
xr-cdp 7.11.1.48I                                   v1.0.0-1
xr-cosm 7.11.1.48I                                  v1.0.0-1
xr-dt-sit 7.11.1.48I                                v1.0.0-1
xr-eigrp 7.11.1.48I                                 v1.0.0-1
xr-healthcheck 7.11.1.48I                           v1.0.0-1
xr-ipsla 7.11.1.48I                                 v1.0.0-1
xr-is-is 7.11.1.48I                                 v1.0.0-1
xr-k9sec 7.11.1.48I                                 v1.0.0-1
xr-license-util 7.11.1.48I                          v1.0.0-1
xr-lldp 7.11.1.48I                                  v1.0.0-1
xr-mpls-oam 7.11.1.48I                              v1.0.0-1
xr-netsim 7.11.1.48I                                v1.0.0-1
xr-olc 7.11.1.48I                                   v1.0.0-1
xr-ospf 7.11.1.48I                                  v1.0.0-1
xr-perfmgmt 7.11.1.48I                              v1.0.0-1
xr-rip 7.11.1.48I                                   v1.0.0-1
xr-telnet 7.11.1.48I                                v1.0.0-1
xr-tftp 7.11.1.48I                                  v1.0.0-1
xr-track 7.11.1.48I                                 v1.0.0-1
```

Displays the list of active packages and rpm files.

Step 9 **show install committed summary**

Example:

```
RP/0/RP0/CPU0:ios#show install committed summary

Wed Nov 15 18:21:35.919 UTC
Committed Packages: XR: 160 All: 1318
Label: 7.11.1.48I-Weekly
Software Hash: ec69dcceb81c0da69b297aa7de1d00f56b8aef52403c5e0ffe6e5db098bd83b8

Optional Packages                    Version
---------------------------------------------------- ----------------------------
```

```
xr-bgp 7.11.1.48I                    v1.0.0-1
xr-cdp 7.11.1.48I                    v1.0.0-1
xr-cosm 7.11.1.48I                   v1.0.0-1
xr-dt-sit 7.11.1.48I                 v1.0.0-1
xr-eigrp 7.11.1.48I                  v1.0.0-1
xr-healthcheck 7.11.1.48I            v1.0.0-1
xr-ipsla 7.11.1.48I                  v1.0.0-1
xr-is-is 7.11.1.48I                  v1.0.0-1
xr-k9sec 7.11.1.48I                  v1.0.0-1
xr-license-util 7.11.1.48I           v1.0.0-1
xr-lldp 7.11.1.48I                   v1.0.0-1
xr-mpls-oam 7.11.1.48I               v1.0.0-1
xr-netsim 7.11.1.48I                 v1.0.0-1
xr-olc 7.11.1.48I                    v1.0.0-1
xr-ospf 7.11.1.48I                   v1.0.0-1
xr-perfmgmt 7.11.1.48I               v1.0.0-1
xr-rip 7.11.1.48I                    v1.0.0-1
xr-telnet 7.11.1.48I                 v1.0.0-1
xr-tftp 7.11.1.48I                   v1.0.0-1
xr-track 7.11.1.48I                  v1.0.0-1

-------------------------------------------------- --------------------------
```

Displays the list of committed packages and rpm files.

Related Commands

The following commands can be used to track the status of the install operation.

Related Commands	Purpose
show install active	Displays the list of active packages.
show install committed	Displays the list of committed packages.
show install log	Displays the log information for the install operation. This information is used for troubleshooting in case of installation failure.
show install package	Displays the details of the packages that are added to the repository. Use this command to identify individual components of a package.
show install request	Displays the current status of the install operation.
show install which	Displays the package information on an installed file.

Upgrade FPD

A Field Programmable Device (FPD) refers to any programmable hardware device on a system which includes a Field Programmable Gate Array (FPGA). You can use the following tasks to verify and upgrade the FPDs of line cards, which are critical for chassis operation.

Note During the software upgrade, when the SSD is upgraded, the FPD goes into the RELOAD_REQ state, as displayed by the **show hw-module fpd** command. This behavior is expected because the updated SSD firmware can only be activated after reloading the specific SSD location mentioned in the **show hw-module fpd** output.

The following table lists the NCS 1014 FPDs that are distributed across Route Processor (RP), Power Modules (PM), Line Cards (LC), and Rack.

Table 3: NCS 1014 FPDs

Location	FPDs
RP	• ADM-DB • ADM-MB • BIOS • BIOS-Golden • CpuFpga • CpuFpgaGolden • SsdIntelS4510 • SsdIntelSC2KB • SsdMicron5300 • TamFw • TamFwGolden
PM0 and PM1	• PO-PriMCU • PO-SecMCU
LC	• CpuModFw • OptModFw
Rack	• ADM-CHASSIS • IoFpga • IoFpgaGolden • SsdIntelSC2KB

Golden FPDs serve as backup FPDs for the primary FPDs. For example, **BIOS-Golden** is the backup Golden FPD for the **BIOS** primary FPD. If a primary FPD is corrupted, NCS 1014 boots with the corresponding Golden FPD. The Golden FPDs cannot be upgraded.

Retrieve FPD Information

There are multiple types of FPDs for each type of module. The **show hw-module fpd** command provides information about each FPD.

```
RP/0/RP0/CPU0:ios#show hw-module fpd
Wed Nov 15 19:29:37.061 UTC

Auto-upgrade:Enabled
Attribute codes: B golden, P protect, S secure, A Anti Theft aware
                                                                  FPD Versions
                                                                  ==============
Location    Card type       HWver   FPD device     ATR Status   Running Programd Reload Loc
-------------------------------------------------------------------------------------------
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   ADM-DB              CURRENT    2.10  2.10    NOT REQ
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   ADM-MB              CURRENT    2.30  2.30    NOT REQ
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   BIOS S              CURRENT    4.70  4.70    0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   BIOS-Golden      BS CURRENT          4.70    0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   CpuFpga           S CURRENT    1.09  1.09    0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   CpuFpgaGolden    BS CURRENT          1.09    0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   SsdMicron5300     S CURRENT    0.01  0.01    0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   TamFw             S CURRENT    9.04  9.04    0/RP0
0/RP0/CPU0 NCS1K14-CNTLR-K9  0.2   TamFwGolden      BS CURRENT          9.04    0/RP0
0/PM0      NCS1K4-AC-PSU     0.1   PO-PriMCU           CURRENT    2.04  2.04    NOT REQ
0/PM0      NCS1K4-AC-PSU     0.1   PO-SecMCU           CURRENT    2.06  2.06    NOT REQ
0/PM1      NCS1K4-AC-PSU     0.1   PO-PriMCU           CURRENT    2.04  2.04    NOT REQ
0/PM1      NCS1K4-AC-PSU     0.1   PO-SecMCU           CURRENT    2.06  2.06    NOT REQ
0/0/NXR0   NCS1K4-1.2T-K9    0.1   CpuModFw          S CURRENT  234.10 234.10  NOT REQ
0/0/NXR0   NCS1K4-1.2T-K9    0.1   OptModFw          S CURRENT    1.38  1.38    NOT REQ
0/1/NXR0   NCS1K14-2.4T-K9   0.1   CpuModFw          S CURRENT  234.10 234.10  NOT REQ
0/2/NXR0   NCS1K14-CCMD-16-C 0.1   CpuModFw          S CURRENT  234.10 234.10  NOT REQ
0/2/NXR0   NCS1K14-CCMD-16-C 0.1   OptModFw          S CURRENT    1.38 1.38    NOT REQ
0/3/NXR0   NCS1K4-1.2T-K9    0.1   CpuModFw          S CURRENT  234.10 234.10  NOT REQ
0/3/NXR0   NCS1K4-1.2T-K9    0.1   OptModFw          S CURRENT    1.38 1.38    NOT REQ
0/Rack     NCS1014           0.1   ADM-CHASSIS         CURRENT    0.21 0.21    NOT REQ
0/Rack     NCS1014           0.1   IoFpga            S CURRENT    1.10 1.10    NOT REQ
0/Rack     NCS1014           0.1   IoFpgaGolden     BS CURRENT         1.05    NOT REQ
0/Rack     NCS1014           0.1   SsdIntelSC2KB     S CURRENT    1.20 1.20    0/Rack
```

The following table describes the significant fields in the output of the **show hw-module fpd** command.

Table 4: Description of Fields in show hw-module fpd Command

Field	Description
Location	Location of the FPD.
Card type	PID of the modules such as chassis, card, CPU, and PSU.
HWver	Hardware version where the FPD resides.
FPD device	Name of the FPD.

Field	Description
ATR	Attribute codes. The possible values are: • B - Golden Image • S - Secure Image • P - Protect Image The attribute code of the primary FPDs is S and the Golden FPDs is BS.
Status	Status of the FPD. See Table 5: Description of FPD Status Values in show hw-module fpd Command Output, on page 69.
Running	FPD image version that has been activated and currently running in the FPD device.
Programd	FPD image version that has been programmed into the FPD device, but might not be activated.
Reload Loc	Indicates whether reload of the location is required or not.

The following table describes the possible values of the **Status** field in the output of the **show hw-module fpd** command.

Table 5: Description of FPD Status Values in show hw-module fpd Command Output

FPD Status	Description
NOT READY	The driver that owns the FPD device has not initialized the FPD client to handle this device.
CURRENT	FPD version is up-to-date and upgrade is not required.
NEED UPGD	Upgrade is required for this FPD. Check the output of the **show fpd package** command to determine the recommended FPD version.
UPGD PREP	FPD is preparing for upgrade.
IN QUEUE	Upgrade of this FPD is in queue.
UPGD SKIP	FPD upgrade is not required. For example, • FPD version is up-to-date and compatible. • FPD image is protected.
UPGRADING	FPD upgrade has started and the driver has not reported the upgrade progress information yet.
%UPGD	Percentage of FPD upgrade completion.

FPD Status	Description
RLOAD REQ	FPD upgrade is successful and the FPD must be reloaded for the new version to take effect.
UPGD FAIL	FPD upgrade has failed.Check the syslog for any timeout messages or any failure reported by the driver.
UPGD DONE	FPD upgrade is successfull.

Restriction The NCS 1014 does not support trunk FPD upgrade on the QXP card.

Verify if an FPD Upgrade is Required

Table 6: Feature History

Feature Name	Release Information	Feature Description
Automatic FPD Upgrade Support for Coherent Interconnect Module 8	Cisco IOS XR Release 24.3.1	The Coherent Interconnect Module 8 (CIM8) FPD is automatically upgraded to the latest qualified version when the line card FPD is upgraded. This ensures that both the line card and CIM8 operate with optimized performance and improved interoperability. Supported line cards are: • NCS1K14-2.4T-X-K9 • NCS1K4-2.4T-K9

Procedure

Step 1 Use the **show hw-module fpd** command to check whether all the FPDs are in the Current state.

If the status of any FPD is **NEED UPGD**, then the upgrade is required for that FPD.

Step 2 Use the **show fpd package** command to determine the FPDs that are supported with the current software release and the minimum hardware requirements for each FPD.

The Coherent Interconnect Module 8 (CIM8) FPD is automatically upgraded to the latest qualified version while the line card FPD is being upgraded. If the CIM 8 has a higher version than the line card's CIM 8 FPD version, the CIM 8 remains at the higher version and does not downgrade.

- NCS1K14-2.4T-X-K9

- NCS1K4-2.4T-K9

You can see the CIM 8 FPD version in the **show fpd package** command output.

```
RP/0/RP0/CPU0:ios#show fpd package
Wed Jul 17 11:44:07.258 IST

================================ ============================================
                                        Field Programmable Device Package
                                 ============================================
                                          Req      SW       Min Req   Min Req
Card Type            FPD Description     Reload   Ver      SW Ver    Board Ver
================== ========================== ====== ======= ======== =========
--------------------------------------------------------------------------------
NCS1014              ADM-CHASSIS               NO      0.21      0.21      0.0
                     IoFpga                    NO      1.19      1.19      0.0
                     IoFpgaGolden              NO      1.05      1.05      0.0
                     SsdIntelSC2KB             YES     1.20      1.20      0.0
                     SsdMicron5400             YES     0.02      0.02      0.0
--------------------------------------------------------------------------------
NCS1K-MD-32E-C       MD-32-ACC                 NO      2.18      2.18      0.0
                     MD-32-NEO                 NO      2.02      2.02      0.0
.
.
.
--------------------------------------------------------------------------------
NCS1K14-2.4T-K9      CIMFw                     NO    180.13019 180.13019   0.0
                     CpuModFw                  NO     43.27     43.27      0.0
--------------------------------------------------------------------------------
NCS1K14-2.4T-L-K9    CIMFw                     NO    180.13019 180.13019   0.0
                     CpuModFw                  NO     43.27     43.27      0.0
--------------------------------------------------------------------------------
NCS1K14-2.4T-X-K9    CIMFw                     NO    180.13019 180.13019   0.0
                     CpuModFw                  NO     43.27     43.27      0.0
--------------------------------------------------------------------------------
NCS1K14-2.4TXL-K9    CIMFw                     NO    180.13019 180.13019   0.0
                     CpuModFw                  NO     43.27     43.27      0.0
--------------------------------------------------------------------------------
.
.
NCS1K14-CNTLR-K9     ADM-DB                    NO      2.10      2.10      0.2
                     ADM-MB                    NO      2.30      2.30      0.2
                     BIOS                      YES     4.80      4.80      0.0
                     BIOS-Golden               YES     4.70      0.01      0.0
```

The following table describes the fields in the output of the **show fpd package** command.

Table 7: Description of Fields in show fpd package Command

Field	Description
Card Type	PID of the modules such as chassis, card, CPU, and PSU.
FPD Description	Description of the FPD.
Req Reload	Determines whether reload is required to activate the FPD image.
SW Ver	Recommended FPD software version for the associated module running the current Cisco IOS XR Software.

Field	Description
Min Req SW Ver	Minimum required FPD software version to operate the module.
Min Req Board Ver	Minimum required hardware version for the associated FPD. A minimum hardware requirement of version 0.0 indicates that all the hardware can support this FPD version.

FPD can be upgraded using two methods:

- Manual FPD Upgrade
- Automatic FPD upgrade

Manual FPD Upgrade

Use the following procedure to upgrade the FPDs manually.

Procedure

Step 1 Use the **upgrade hw-module location** *[location-id]* **fpd** *[fpd name]* command to upgrade a specific FPD.

Note
FPD upgrades are non-traffic affecting.

Example:

```
RP/0/RP0/CPU0:ios#upgrade hw-module location 0/0/NXR0 fpd CPUModFw
```

Step 2 Use the **show hw-module fpd** command to display information about the completed FPD upgrade.

Step 3 (Optional) Use the **upgrade hw-module location** *[location-id]* **fpd** *[fpd name]* **force** command toforcibly upgrade a specific FPD irrespective of whether the upgrade is required or not.

Example:

```
RP/0/RP0/CPU0:ios#upgrade hw-module location 0/0/NXR0 fpd CPUModFw force
```

Step 4 Use the **reload location** *location-id* to reload the FPDs belonging to a specific location with the new version.

The **Reload Loc** field in the output of **show hw-module fpd** command indicates whether the reload is required or not.

Example:

```
RP/0/RP0/CPU0:ios#reload location 0/RP0/CPU0
```

Step 5 (Optional) Use the **upgrade hw-module location all fpd all** command to upgrade all the FPDs concurrently.

Example:

```
RP/0/RP0/CPU0:ios#upgrade hw-module location all fpd all
```

Note

You cannot upgrade PSU FPD using **location all fpd all** command. You can excecute **Step 6** command to upgrade PSU FPD.

Step 6 (Optional) Use the **upgrade hw-module [location** [*location-id* | **all**]] **fpd** [*fpd name*] | **all**] command to upgrade a specific FPD, all the FPDs, or the FPDs belonging to a specific location.

Note
Until Release 24.2.1, you cannot forcefully upgrade FPDs of power modules and SSDs.

From Release 24.3.1, you can upgrade power module FPDs by using the command:

upgrade hw-module [location [*location-id* | **all**]] **[pm** [*pm-id* | **all**]] **fpd** [*fpd name*] | **all**].

Automatic FPD upgrade

Table 8: Feature History

Feature Name	Release Information	Feature Description
Automatic FPD Upgrade Support for Power Module	Cisco IOS XR Release 24.3.1	The FPD upgrade for power modules is now integrated with the NCS 1014 automatic FPD upgrade. You have the flexibility to include or exclude the power module FPD in the automatic upgrade according to your operational requirements. This option is disabled by default. Command added: • **fpd auto-upgrade {include | exclude} pm** You can also enable the automatic FPD upgrade for power modules using the OpenConfig data model Cisco-IOS-XR-openconfig-system-fpd-ext.

The automatic FPD upgrade process the firmware upgraded with the **NEED UPGD** status to **CURRENT** status automatically. Use the **show hw-module fpd** command to view the latest status after the automatic upgrade is completed.

In NCS 1014, automatic FPD upgrade is enabled by default.

Procedure

Step 1 Use the following commands to disable automatic FPD upgrade.

Example:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#fpd auto-upgrade disable
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Note

- CpuModFw is upgraded during the automated FPD upgrade for cards NCS1K14-2.4T-K9 and NCS1K14-2.4T-L-K9.
- OptModFw is upgraded first followed by CpuModFw during automated FPD upgrade for the cards NCS1K14-CCMD-16-C, NCS1K14-CCMD-16-L, and NCS1K4-1.2T-K9.
- Untill R24.2.1, you cannot do an automatic upgrade for the FPD power module.

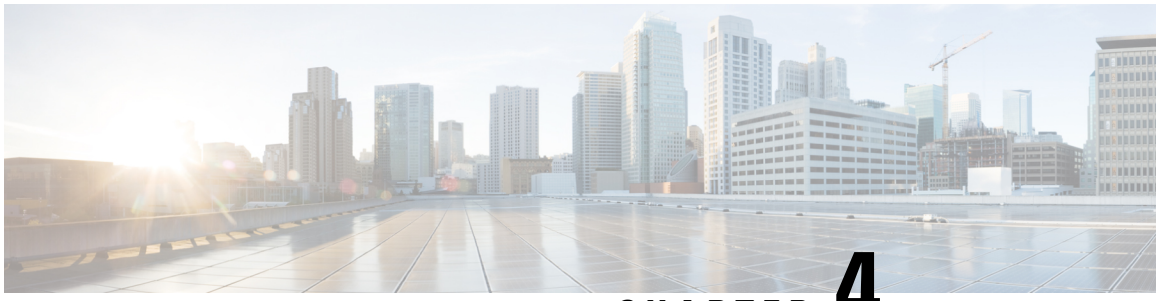
Step 2 (From R24.3.1), to include or exclude automatic upgrade of power modules, use the following commands:

Example:

```
RP/0/RP0/CPU0:ios#fpd auto-upgrade include pm
RP/0/RP0/CPU0:ios#commit
```

Example:

```
RP/0/RP0/CPU0:ios#fpd auto-upgrade exclude pm
RP/0/RP0/CPU0:ios#commit
```



CHAPTER 4

Disaster Recovery

This chapter describes the disaster recovery process and the health check feature.

- Overview, on page 75
- CPU Replacement Considerations, on page 75
- How the Node Recovers After a Chassis SSD Replacement, on page 76
- Health Check of Backup ISO Image, on page 76

Overview

There are two partitions in NCS 1014: RP SSD (CPU partition) and chassis SSD (Disaster Recovery partition). The Disaster Recovery partition contains all the backup configurations such as ISO images, RPMs, and system configuration files. When the node is corrupted, the Disaster Recovery feature allows the CPU to be replaced with the existing configuration. After replacing the CPU, the node reboots and comes up by restoring the software and configuration files from the chassis SSD without traffic loss.

Note When Chassis SSD is corrupted and replaced, chassis SSD takes backup of the running software and configuration files from the RP SSD without traffic loss.

CPU Replacement Considerations

You must consider the following points for CPU replacement.

- When the CPU is removed from the chassis, NCS 1014 chassis runs in headless mode which is non-traffic impacting.
- When the CPU is replaced with another CPU having the same software and RPMs as in the chassis SSD, the configuration is restored from the chassis SSD.
- When the CPU is replaced with another CPU having different software and RPMs as in the chassis SSD, the Disaster recovery process starts. In this case, the node boots with the software from the chassis SSD and the configuration is also restored from the chassis SSD.

How the Node Recovers After a Chassis SSD Replacement

The chassis SSD (NCS1K14-SSD) in NCS 1014 is a hot-swappable module, meaning you can replace it without shutting down the system.

Attention When you remove the chassis SSD while the controller is in operation, the system raises the *Disaster Recovery Unavailable* alarm. This alarm clears automatically after you install the new chassis SSD.

These scenarios describe how the node recovers after a chassis SSD replacement.

Table 9: Replacing a Chassis SSD in an Operational Node: How the Node Recovers in Live Network

When you replace the chassis SSD with…	Then the system loads only…	And then the new chassis SSD takes a backup of…
another that has the same software and RPMs as the controller SSD	the configuration from the controller SSD	the system configuration from the controller SSD.
another that has the different software and RPMs from the controller SSD	the software, RPMs, and the configuration from the controller SSD	all the contents from the controller SSD.
another received from Cisco manufacturing as a spare or through an RMA process, which comes with a blank configuration	the software, RPMs, and the configuration from the controller SSD, after inserting this SSD to this system	all controller SSD contents.

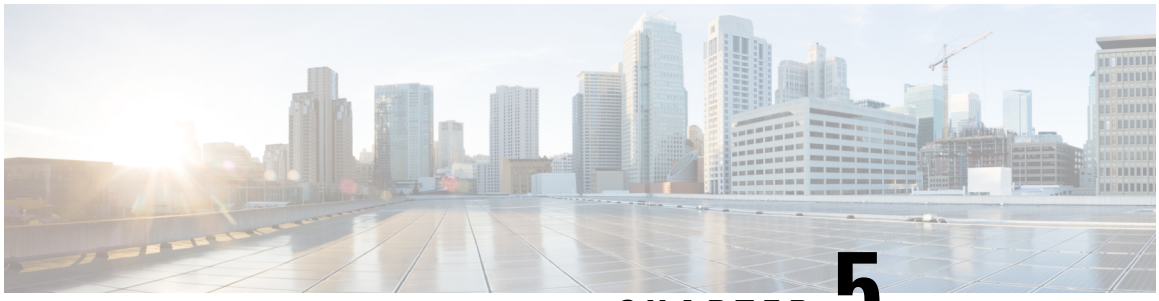
Table 10: Replacing a Chassis SSD in a Shutdown Node: How the Node Recovers after Boot Up

When you replace the chassis SSD with…	Then upon powering on…	And then the system…
another that has software and RPMs different from the contents of the controller SSD	the system triggers the disaster recovery workflow	installs the software, RPMs, and configurations from the chassis SSD.

Health Check of Backup ISO Image

The Health Check feature ensures error-free booting of NCS 1014 chassis during disaster recovery operations. NCS 1014 has a partition for disaster recovery where the backup ISO image is stored. The backup ISO image is stored in the chassis SSD.

The chassis SSD content is audited against the running software by the install process in the background every 12 hours to detect corruption. If the ISO image is corrupted, the software will recover it by copying from the backup location. If the software fails to synchronize with the chassis SSD, then the **Disaster Recovery ISO Image Corruption** alarm is raised. See the *Troubleshooting Guide for Cisco NCS 1014* to clear the alarm.



CHAPTER 5

Connection Verification

This chapter describes the tasks to verify connection between the OLT Line Card of NCS 1010 and NCS1K14-CCMD-16-C line card.

- Power Data Reading, on page 77
- Connection Verification, on page 77
- Verify Connection for CCMD-16 Line Card, on page 78

Power Data Reading

Photodiodes (PDs) are optical power monitors available on all input and aggregated output ports to monitor power levels. Tone detection is enabled on some PD monitors.

Table 11: NCS1K-CCMS-16 Calibrated Port References

Photodiode	Port Calibrated	Port Label (Direction)	Minimum Power (dBm)	Maximum Power (dBm)	Dynamic Range (dBm)
PD 21	MPO-16 input ports	(TX)	–50	30	80
PD 22	MPO-16 output ports	(RX)	–50	30	80

Connection Verification

Connection verification checks the connection between the OLT line card and the CCMD-16 line cards to avoid miscabling during the node installation. The dedicated Connection Verification Tunable Laser (CV-TL) available at the OLT card generates a specific probe signal at a given frequency and power. This signal is detected by the CCMD-16 line card that is connected to the OLT line card.

For more information on the connection verification process, see Cisco NCS 1010 Datapath Configuration guide.

CCMD-16 Connection Verification with OLT

The OLT line card generates the tone and connection verification is performed using the OOB channel with CV-TL tuned at 191.175 THz. To univocally identify the optical path under test, the CV-TL is modulated with a low-frequency pattern including the Cable ID of the connection.

For connection verification toward the CCMD-16 card, the CV-TL is routed to the PD21 inside the CCMD-16 card. The out-of-band (OOB) connection is verified at PD21 and the in-band (IB) connection is verified at PD22 on the CCMD-16 line cards.

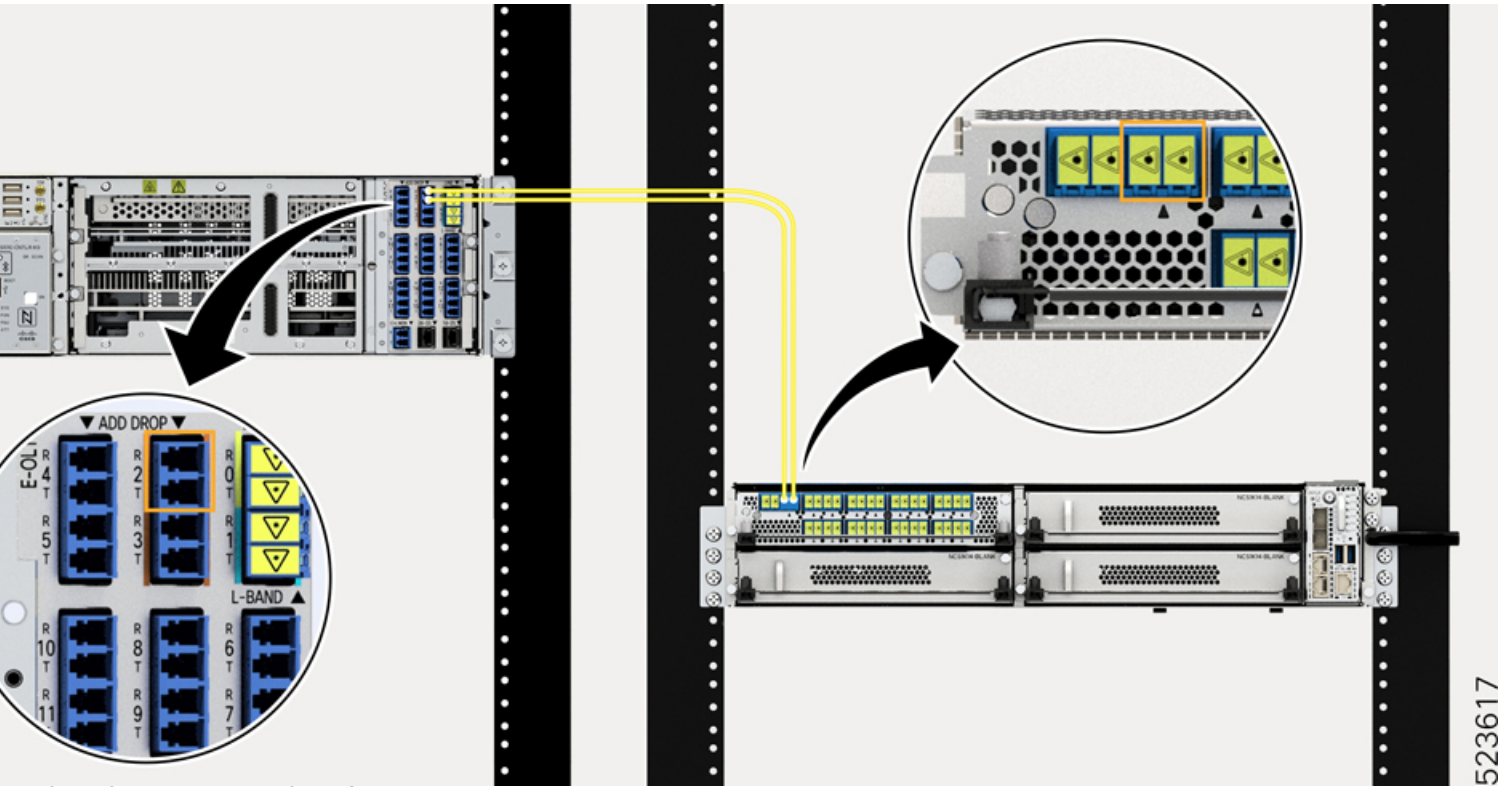
The PD monitors receiving a connection verification signal detect and buffer the Cable-ID pattern encoded in the tone to allow the connection verification process by the node controller.

Verify Connection for CCMD-16 Line Card

The connection verification procedure checks the connection between the OLT line card and CCMD-16 line cards to match the different instances with respect to the OLT LC connectors.

The OLT-C line card and the NCS1K-CCMD-16 line card are connected as shown in the following image:

Figure 1: NCS 1010 and NCS1K-CCMD-16



The OLT-C line card performs connection verification between the OLT-C line card and the NCS1KCCMD-16 line card panels as described in CCMD-16 Connection Verification with OLT, on page 78.

The identification/verification of the NCS1K-CCMD-16 line card is performed by checking the connection verification signal at the monitor present on the OOB and IB loops (PD21 and PD22 for the NCS1K-CCMD-16 line card respectively).

This task describes on how to verify the connection between the NCS 1010 OLT-C line card and NCS1K-CCMD-16 line card.

Before you begin

Configure the OTS controller in NCS 1010 to generate the tone for connection verification. See Cisco NCS 1010 Datapath Configuration Guide.

Procedure

Step 1 Configure the OMS controller to detect the tone for connection verification.

Example:

```
RP/0/RP0/CPU0:(config)#controller oms 0/1/0/0
RP/0/RP0/CPU0:(config-Oms)#tone-rate 2
RP/0/RP0/CPU0:(config-Oms)#tone-pattern-expected aabbccdd
RP/0/RP0/CPU0:(config-Oms)#tone-detect-oob
RP/0/RP0/CPU0:(config-Oms)#commit
```

tone-detect-oob must be configured on the OMS x/x/x/0 for NCS1K-CCMD-16.

Step 2 Use the **tone-pattern-detect** command to start the detection of tone pattern.

Example:

The following is a sample on starting the tone pattern detection on the OMS controller.

```
RP/0/RP0/CPU0:#tone-pattern-detect controller oms 0/1/0/0 start
Tue May 10 11:38:03.775 UTC
Tone pattern detect started
```

Step 3 Use the **tone-info** command to check for successful connection verification.

Example:

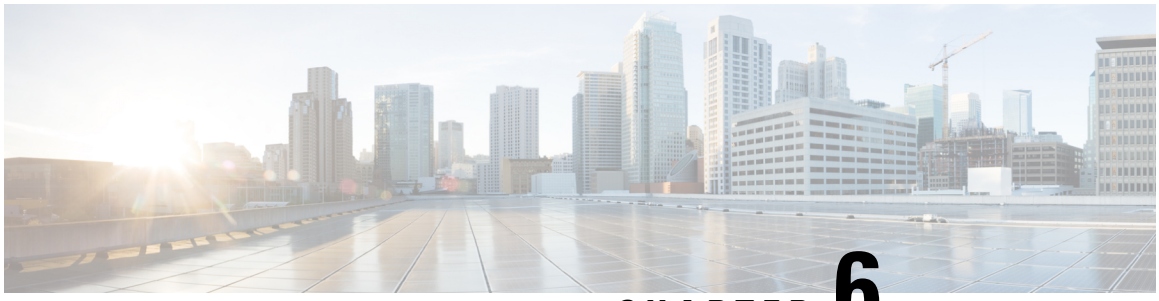
The following is a sample to view the Tone Info for successful connection verification on the OMS controller.

```
RP/0/RP0/CPU0:#show controllers oms 0/1/0/0 tone-info
Fri Sep 22 06:04:03.787 UTC
Tone Info:
Tone Rate : 2 bits/second
Tone Pattern Expected(Hex value) : aabbccdd
Tone Pattern Received(Hex value) : aabbccdd
Tone Detected OOB :  Enabled
Detection State: Success
```

Step 4 After successful connection verification, stop **tone-pattern-detect** on the OMS controller.

Example:

```
RP/0/RP0/CPU0:#tone-pattern-detect controller oms 0/1/0/0 stop
Fri Sep 22 06:23:15.165 UTC
Tone pattern detect stopped
```



CHAPTER 6

System Health Check

Monitoring systems in a network proactively helps prevent potential issues and take preventive actions. This chapter describes the tasks to configure and monitor system health check.

- System Health Check, on page 81
- Enable Health Check, on page 82
- Change Health Check Refresh Time, on page 83
- View Status of All Metrics, on page 83
- Change Threshold Value for a Metric, on page 86
- View Health Status of Individual Metric, on page 86
- Disable Health Check, on page 88

System Health Check

Proactive network monitoring systems play a pivotal role in averting any issues. NCS 1014 health check service lets you monitor physical characteristics, current processing status, and the curently utilized resources to quickly assess the condition of the device at any time. This service helps to analyze the system health by monitoring, tracking and analyzing metrics that are critical for functioning of the NCS 1014. The system health metrics are thresholds set on the device in order to monitor the usage of CPU and other system resources. The health check service is installed with the NCS 1014 RPM.

You can evaluate the system's health by examining the metric values. If these values cross or approach the set thresholds, it suggests potential problems. By default, metrics for system resources are configured with preset threshold values. You can customize the metrics to be monitored by disabling or enabling metrics of interest based on your requirement.

Each metric is tracked and compared with that of the configured threshold, and the state of the resource is classified accordingly.

The system resources metrics can be in one of these states:

- **Normal:** The resource usage is less than the threshold value.
- **Minor:** The resource usage is more than the minor threshold, but less than the severe threshold value.
- **Severe:** The resource usage is more than the severe threshold, but less than the critical threshold value.
- **Critical:** The resource usage is more than the critical threshold value.

The infrastructure services metrics can be in one of these states:

- **Normal:** The resource operation is as expected.
- **Warning:** The resource needs attention. For example, a warning is displayed when the FPD needs an upgrade.

Supported System Health Check Metrics

NCS 1014 supports the following system health check metrics:

- communication-timeout
- cpu
- filesystem
- fpd
- free-mem
- hw-monitoring
- lc-monitoring
- pci-monitoring
- platform
- process-resource
- process-status
- shared-mem
- wd-monitoring

Enable Health Check

To enable health check, perform the following steps:

Before you begin

Before enabling health check, ensure that:

- An IP address and subnet mask is assigned to the management interface.
- The IP address of the default gateway is configured with a static route.

For more details, see the Configure Management Interface section of the *Cisco NCS 1014 System Setup and Software Installation Guide*.

Procedure

Step 1 Enter into the configuration mode using the **configuration** command.

Step 2 Enable health check using the **healthcheck enable** command.

Example:

```
RP/0/RP0/CPU0:ios(config)# healthcheck enable
```

Step 3 Run the **netconf-yang agent ssh** command.

Example:

```
RP/0/RP0/CPU0:ios(config)# netconf-yang agent ssh
```

Step 4 Enable Google Remote Procedure Call (gRPC) using the **grpc local-connection** command.

Example:

```
RP/0/RP0/CPU0:ios(config)# grpc local-connection
```

Step 5 Commit the changes using the **commit** command.

Change Health Check Refresh Time

Cadence is the time interval, in seconds, at which the health check status is refreshed. By default, this time is 60 seconds which means that health check status is updated every 60 seconds. You can change this time using the **healthcheck cadence** *cadence-value* command.

The following example shows to change the health check cadence value to 50 seconds so that health check status is updated every 50 seconds.

```
RP/0/RP0/CPU0:ios(config)#healthcheck cadence 50
```

View Status of All Metrics

You can view the status of all the supported metrics with the associated threshold and configured parameters in the system. To check the status of all the metrics, perform these steps:

Procedure

Step 1 Run the **show healthcheck status** command.

Example:

```
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios#show healthcheck status
Sat Jun 12 02:00:25.204 UTC

Healthcheck status: Enabled
Time started: 12 Jun 02:00:22.392972

Collector Cadence: 30 seconds

METRICS STATS

System Resource metrics
   cpu
```

```
        Thresholds: Minor: 20%
                    Severe: 50%
                  Critical: 75%

        Tracked CPU utilization: 15 min avg utilization

    free-memory
        Thresholds: Minor: 10%
                    Severe: 8%
                  Critical: 5%

    filesystem
        Thresholds: Minor: 80%
                    Severe: 95%
                  Critical: 99%

    shared-memory
        Thresholds: Minor: 80%
                    Severe: 95%
                  Critical: 99%

Infra Services metrics
    platform

    fpd

Install Custom Metrics
    process-status

    process-resource

    communication-timeout

    pci-monitoring

    hw-monitoring

    wd-monitoring

    lc-monitoring

Use case
 Use cases are disabled
```

Step 2 To view the health state of the health check manager, use the **show healthcheck internal states** command.

Example:

```
RP/0/RP0/CPU0:ios#show healthcheck internal states
Sat Jun 12 02:00:55.425 UTC

 Internal Structure INFO

 Current state: Enabled

 Reason: Success

 Netconf Config State: Enabled

 Grpc Config State: Enabled

 Nosi state: Initialized

 Appmgr conn state: Connected
```

```
 Nosi lib state: Not ready

 Nosi client: Valid client
```

Step 3 To view the health state for each enabled metric, use the **show healthcheck report** command.

Example:

```
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios#show healthcheck report
Sat Jun 12 02:02:54.417 UTC

Healthcheck report
Last Update Time: 12 Jun 02:02:46.955241
METRICS REPORT

cpu
  State: Normal

free-memory
  State: Normal

filesystem
  State: Normal

shared-memory
  State: Normal

platform
  State: Warning
  Reason: One or more devices are not in operational state

fpd
  State: Warning
  Reason: One or more FPDs are not in CURRENT state

process-status
  State: Normal

process-resource
  State: Normal

communication-timeout
  State: Normal

pci-monitoring
  State: Normal

hw-monitoring
  State: Normal

wd-monitoring
  State: Normal

lc-monitoring
  State: Normal
```

In the above output, the state of the FPD shows a warning message that indicates an FPD upgrade is required.

Change Threshold Value for a Metric

You can customize the health check threshold value for a metric using the following command:

healthcheck metric *metric-name* **threshold** *threshold-value*

Example to Change Preset Metric Value

The following example shows to change the threshold value of CPU metric to 25%.

```
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios(config)#healthcheck metric cpu minor threshold 25%
```

View Health Status of Individual Metric

You can view the health status of a system resource or infrastructure service metric in the system.

Procedure

Run the **show healthcheck metric** *metric-name* command.

Example:

The following example shows how to obtain the health-check status for the *filesystem* metric:

```
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios#show healthcheck metric filesystem
Sat Jun 12 02:01:32.432 UTC
Filesystem Metric State: Normal
Last Update Time: 12 Jun 02:01:04.446619
Filesystem Service State: Enabled
Number of Active Nodes: 1
Configured Thresholds:
   Minor: 80%
   Severe: 95%
   Critical: 99%

Node Name: 0/RP0/CPU0
    Partition Count: 5

    Partition Name: tftp:
        Partition Access Attribute: rw
        Partition Type: network
        Partition Size: 0
        Partition Free Bytes: 0
        Partition Free Space in %: 0

    Partition Name: disk0:
        Partition Access Attribute: rw
        Partition Type: flash-disk
        Partition Size: 20024897536
        Partition Free Bytes: 19978481664
        Partition Free Space in %: 99

    Partition Name: /misc/config
```

```
        Partition Access Attribute: rw
        Partition Type: flash
        Partition Size: 151314698240
        Partition Free Bytes: 146903269376
        Partition Free Space in %: 97

    Partition Name: harddisk:
        Partition Access Attribute: rw
        Partition Type: harddisk
        Partition Size: 150114078720
        Partition Free Bytes: 144962641920
        Partition Free Space in %: 96

    Partition Name: ftp:
        Partition Access Attribute: rw
        Partition Type: network
        Partition Size: 0
        Partition Free Bytes: 0
        Partition Free Space in %: 0
```

Example:

The following example shows how to obtain the health-check status for the *platform* metric:

```
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios#show healthcheck metric platform
Sat Jun 12 02:01:51.922 UTC
Platform Metric State: Warning
Last Update Time: 12 Jun 02:01:38.650003
Platform Service State: Enabled
Number of Racks: 1

Rack Name: 0
    Number of Slots: 5

    Slot Name: RP0
        Number of Instances: 1

    Instance Name: CPU0
        Node Name 0/RP0/CPU0
        Card Type NCS1K14-CNTLR-K9
        Card Redundancy State Active
        Admin State NSHUT,NMON
        Oper State IOS XR RUN

    Slot Name: PM1
        Number of Instances: 0

        Node Name 0/PM1
        Card Type NCS1K4-AC-PSU-2
        Card Redundancy State None
        Admin State NSHUT,NMON
        Oper State OPERATIONAL

    Slot Name: FT1
        Number of Instances: 0

        Node Name 0/FT1
        Card Type NCS1K14-FAN
        Card Redundancy State None
        Admin State NSHUT,NMON
        Oper State OPERATIONAL

    Slot Name: FT2
        Number of Instances: 0
```

```
    Node Name 0/FT2
    Card Type NCS1K14-FAN
    Card Redundancy State None
    Admin State NSHUT,NMON
    Oper State OPERATIONAL

Slot Name: 2
    Number of Instances: 1

Instance Name: NXR0
    Node Name 0/2/NXR0
    Card Type NCS1K4-1.2T-K9
    Card Redundancy State None
    Admin State NSHUT,NMON
    Oper State CARD FAILED
```

Disable Health Check

You can disable health check service or disable health check for an individual metric. By default, health check of all the metrics is enabled.

Disable Health Check Service

To disable health check service, use the following command:

no healthcheck enable

Note When the health check service is enabled, other configuration changes are not permitted. Disable the service before committing configuration changes.

The following example shows to disable the health check service.

```
RP/0/RP0/CPU0:#configure
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios(config)#no healthcheck enable
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios(config)#commit
```

Disable Health Check for a Metric

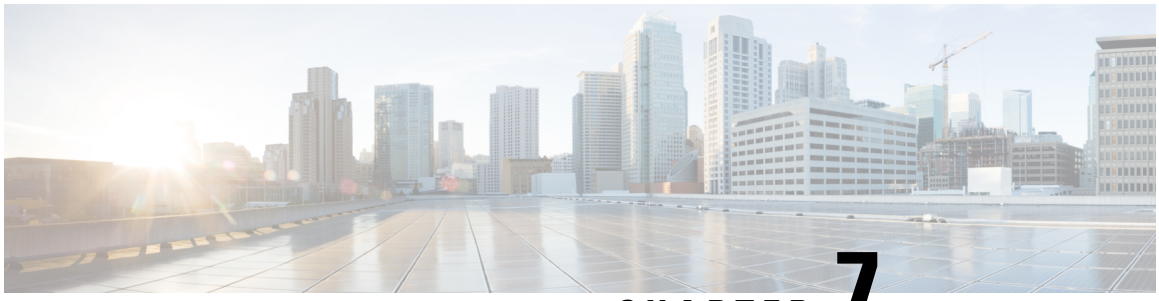
To disable health check for an individual metric, use the following command:

healthcheck metric *metric-name* **disable**

Example to Disable Health Check of a Metric

The following example shows to disable the free memory (*free-mem*) metric.

```
RP/0/RP0/CPU0:RP/0/RP0/CPU0:ios(config)#healthcheck metric free-mem disable
```



CHAPTER 7

Configure AAA

This chapter describes the implementation of the administrative model of *task-based authorization* used to control user access in the software system. The major tasks required to implement task-based authorization involve configuring TACACS+ and RADIUS servers and groups.

Note From Release 24.4.1, the AAA local database supports configuring up to 3000 usernames. Although you can configure more than 3000 users, it may impact the system's scale and performance, which are not assured beyond this limit.

- Deprecation of Type 7 password and Type 5 secret, on page 89
- About TACACS+, on page 94
- Configure TACACS+ Server, on page 95
- Configure TACACS+ Server Groups, on page 95
- About RADIUS, on page 97
- Configure RADIUS Server Groups, on page 97

Deprecation of Type 7 password and Type 5 secret

Password configuration options before Release 24.4.1

Until Release 24.4.1, there were two options for configuring a password:

- Password: Uses Type 7 encryption to store the password.
- Secret: Supports Type 5, 8, 9, or 10 hashing algorithms to store the password securely.

Deprecation notice

Starting from the Release 24.4.1, the use of Type 7 password and Type 5 secret are deprecated due to security concerns. The deprecation process commences from the Release 24.4.1. We expect the full deprecation in a future release. We recommend using the default option, which is Type 10 secret.

- password, on page 90
- masked-password, on page 90
- password-policy, on page 91

- aaa password-policy, on page 92
- secret, on page 92
- masked-secret, on page 93

password

The **password** options available in CLI from the Release 24.4.1:

```
RP/0/RP0/CPU0:ios(config-un)#password ?
LINE The type 7 password followed by '7 ' OR SHA512-based password (deprecated, use 'secret')
```

Changes:

- All the options that were present until the Release 24.4.1are removed except LINE (to accept cleartext).
- **During upgrade**: Any configuration using the Type 7 password configuration is automatically converted to Type 10 secret.

  Post-upgrade: You can still use the Type 7 password configurations option after new commits, but the password will be stored as Type 10 secret.
- New **syslog** has been added to indicate the deprecation process:

  ```
  %SECURITY-PSLIB-4-DEPRECATED_PASSWORD_TYPE : The password configuration is deprecated.

   Converting it to a Type 10 secret for user <user name>.
  ```
- **show running configuration** command output before upgrade:

  ```
  username example
  password 7 106D000A0618
  !
  ```

  show running configuration command output post-upgrade:

  ```
  username example
  Cisco Confidential
  secret 10
  $6$P53pb/FFxNIT4b/.$yVakako4fp9PZiIYYh1xS0.W6b/yPrSyC8j4gLs6xli57iClOryPXyN9y8yojRD2nhAWb9pjr/WAIhbXqq8st.
  !
  ```

masked-password

The **masked-password** options available in CLI from the Release 24.4.1:

```
RP/0/RP0/CPU0:ios(config-un)#masked-password ?
0 Specifies a cleartext password will follow
clear Config deprecated. Will be removed in 7.7.1. Specify '0' instead.
<cr> The cleartext user password
```

Changes:

- The options 7 and encrypted that were present until the Release 24.4.1 are removed.
- **During upgrade**: Any configuration using the Type 7 password configuration is automatically converted to Type 10 secret.
- **Post-upgrade**: Masked-password is an alternate method of configuring the password. You can still use the masked-password keyword with a clear string after new commits, but the password will be stored as Type 10 secret.

- New **syslog** has been added to indicate the deprecation process:

```
%SECURITY-PSLIB-4-DEPRECATED_PASSWORD_TYPE : The password configuration is deprecated.

 Converting it to a Type 10 secret for user <user name>.
```

- **show running configuration** command output before upgrade:

```
username example
password 7 106D000A0618
!
```

show running configuration command output post-upgrade:

```
username example
Cisco Confidential
secret 10
$6$P53pb/FFxNIT4b/.$yVakako4fp9PZiIYYh1xS0.W6b/yPrSyC8j4gLs6xli57iClOryPXyN9y8yojRD2nhAWb9pjr/WAIhbXqq8st.
!
```

password-policy

The **password-policy** options available in CLI from the Release 24.4.1:

```
RP/0/RP0/CPU0:ios(config-un)#password-policy ?
WORD Specify the password policy name

RP/0/RP0/CPU0:ios(config-un)#password-policy abcd password ?
0 Specifies an UNENCRYPTED password will follow
7 Specifies that an encrypted password will follow
LINE The UNENCRYPTED (cleartext) user password
clear Config deprecated. Will be removed in 7.7.1. Specify '0' instead.
encrypted Config deprecated. Will be removed in 7.7.1. Specify '7' instead.
```

Changes:

- All the options that were present until 24.4.1are removed except LINE (to accept cleartext).
- **During upgrade**: Any configuration using the Type 7 password configuration is automatically converted to Type 10 secret.

  Post-upgrade: You can still use the password-policy configurations option after new commits, but the it will be stored as Type 10 secret.

- New **syslog** has been added to indicate the deprecation process:

```
%SECURITY-PSLIB-4-DEPRECATED_PASSWORD_TYPE : The password configuration is deprecated.

 Converting it to a Type 10 secret for user <username>.
```

- **show running configuration** command output before upgrade:

```
username example
password-policy abcd password 7 106D000A0618
!
```

show running configuration command output post-upgrade:

```
username example
secret 10
$6$P53pb/FFxNIT4b/.$yVakako4fp9PZiIYYh1xS0.W6b/yPrSyC8j4gLs6xli57iClOryPXyN9y8yojRD2nhAWb9pjr/WAIhbXqq8st.
!
!
```

aaa password-policy

The **aaa password-policy** options available in CLI from the Release 24.4.1:

```
RP/0/RP0/CPU0:ios(config)#aaa password-policy abcd
RP/0/RP0/CPU0:ios(config-pp)#?
min-char-change Number of characters change required between old and new passwords
(deprecated, will be removed in 25.3.1)
restrict-password-advanced Advanced restrictions on new password (deprecated, will be removed
 in 25.3.1)
restrict-password-reverse Restricts the password to be same as reversed old password
(deprecated, will be removed in 25.3.1)
```

Changes:

- The options min-char-change, restrict-password-advanced, and restrict-password-reverse that were present until the Release 24.4.1 are deprecated.
- **During upgrade**: These deprecated configurations do not go through any change during upgrade.

  Post-upgrade: These deprecated keywords do not take effect when configured post-upgrade.
- New **syslog** have been added to indicate the deprecation process:
  - `%SECURITY-LOCALD-4-DEPRECATED_PASSWORD_POLICY_OPTION : The password policy option 'min-char-change' is deprecated. Password/Secret will not be checked against this option now.`
  - `%SECURITY-LOCALD-4-DEPRECATED_PASSWORD_POLICY_OPTION : The password policy option 'restrict-password-reverse' is deprecated. Password/Secret will not be checked against this option now.`
  - `%SECURITY-LOCALD-4-DEPRECATED_PASSWORD_POLICY_OPTION : The password policy option 'restrict-password-advanced' is deprecated. Password/Secret will not be checked against this option now.`
- **show running configuration** command output before upgrade:

  ```
  aaa password-policy abcd
  lower-case 3
  min-char-change 1
  restrict-password-reverse
  restrict-password-advanced
  !
  ```

  show running configuration command output post-upgrade:

  ```
  aaa password-policy abcd
  lower-case 3
  min-char-change 1
  restrict-password-reverse
  restrict-password-advanced
  !
  ```

secret

The **secret** options available in CLI from the Release 24.4.1:

```
RP/0/RP0/CPU0:ios(config-un)#secret ?
0 Specifies a cleartext password will follow
10 Specifies that SHA512-based password will follow
8 Specifies that SHA256-based password will follow
9 Specifies that Scrypt-based password will follow
LINE The cleartext user password
```

```
RP/0/RP0/CPU0:ios(config-un)#secret 0 enc-type ?
<8-10> Specifies which algorithm to use. Only 8,9,10 supported [Note: Option '5' is not
available to use from 24.4]
```

Changes:

- The options 5 and encrypted are removed.
- **During upgrade**: Configurations using Type 5 secret will remain unchanged.

  Post-upgrade: Though the keyword 5 has been deprecated, you can still apply the existing configurations using Type 5 secret.
- New **syslog** has been added to indicate the deprecation process:

  ```
  %SECURITY-LOCALD-2-DEPRECATED_SECRET_TYPE : Type 5 secret is deprecated.
  Please use the 'secret' keyword with option type 10 for user.
  ```
- **show running configuration** command output before upgrade:

  ```
  username example
  secret 5 $1$kACo$2RtpcwyiRuRB/DhWzabfU1
  !
  !
  ```

  show running configuration command output post-upgrade:

  ```
  username example
  secret 5 $1$kACo$2RtpcwyiRuRB/DhWzabfU1
  !
  !
  ```

masked-secret

The **masked-secret** options available in CLI from the Release 24.4.1:

```
RP/0/RP0/CPU0:ios(config-un)#masked-secret ?
0 Specifies a cleartext password will follow
Cisco Confidential
10 Specifies that SHA512-based password will follow
8 Specifies that SHA256-based password will follow
9 Specifies that Scrypt-based password will follow
clear Config deprecated. Will be removed in 7.7.1. Specify '0' instead.
<cr> The cleartext user password
```

Changes:

- The options 5 and encrypted are removed.
- **During upgrade**: Configurations using masked-secret with Type 5 will remain unchanged.

  Post-upgrade: Though the keyword 5 has been deprecated, you can still apply the existing configurations using Type 5 masked secret.
- New **syslog** has been added to indicate the deprecation process:

  ```
  %SECURITY-LOCALD-2-DEPRECATED_SECRET_TYPE : Type 5 secret is deprecated.
  Please use the 'secret' keyword with option type 10 for user.
  ```
- **show running configuration** command output before upgrade:

  ```
  username example
  secret 5 $1$kACo$2RtpcwyiRuRB/DhWzabfU1
  !
  !
  ```

show running configuration command output post-upgrade:

```
username example
secret 5 $1$kACo$2RtpcwyiRuRB/DhWzabfU1
!
!
```

Special use cases

Use case 1: Configurations using both Type 7 password and secret with 8, 9, or 10 hashing, for the same user

- **During upgrade**:
  - For the first 3000 username configurations, the password configuration will be rejected, and the secret configuration will remain unchanged.
  - For the rest of the username configurations, the original secret configuration will be rejected, and the password will be converted to Type 10 secret.
- **Post-upgrade**:
  - For a new username configured, or the username that is already present before the upgrade, the password configuration will be rejected.
  - New **syslog** has been added to indicate the deprecation process:

    ```
    %SECURITY-PSLIB-4-SECRET_CONFIG_PRESENT : The password configuration is deprecated.

    Once secret is configured, cannot use password config for user <user name> at index
     <x> now.
    ```

    where 'x' is a number representing the index.

Use case 2: Configurations using both Type 7 password and Type 5 secret, for the same user

- **During upgrade**:
  - For any username configuration, the original Type 5 secret configuration will be rejected, and the password will be converted to Type 10 secret.
- **Post-upgrade**:
  - For a new username configured, or the username that is already present before the upgrade, the password configuration will be converted to Type 10 secret.
  - New **syslog** has been added to indicate the deprecation process:

    ```
    %SECURITY-PSLIB-4-DEPRECATED_PASSWORD_TYPE : The password configuration is
    deprecated.
     Converting it to a Type 10 secret for user <username>.
    ```

About TACACS+

The Terminal Access Controller Access Control System Plus (TACACS+) application is designed to enhance the security of the NCS 1014 device by centralizing user validation. It uses AAA commands and can be

enabled and configured on NCS 1014 for improved security. TACACS+ provides detailed accounting information and flexible administrative control over user access.

When TACACS+ server is configured and protocol is enabled on the node, the user credentials are authenticated through TACACS+ server. When the user attempts to log into the node, the username and password is forwarded to the configured TACACS+ servers and get authentication status. If the authentication fails through TACACS+ server, the credentials are sent to the node and are authenticated against the node. If the authentication fails against the node, the user is not allowed to log into the node.

Configure TACACS+ Server

Enabling the AAA accounting feature on a switch allows it to track the network services that users are accessing and the amount of network resources they are using. The switch then sends this user activity data to the TACACS+ security server in the form of accounting records. Each record contains attribute-value pairs and is saved on the security server for analysis. This data can be used for network management, client billing, or auditing purposes.

To configure TACACS+ server, perform these steps:

Before you begin

Procedure

Step 1 Enter into the IOS XR configuration mode.

Example:

```
RP/0/RP0/CPU0:ios#configure
```

Step 2 Enable the TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.

Example:

```
RP/0/RP0/CPU0:ios(config)#aaa accounting exec default start-stop group TACACS_ALL
```

Step 3 Create a default command accounting method list for accounting services provided by a TACACS+ security server. This list is configured for privilege level commands and set with a stop-only restriction.

Example:

```
RP/0/RP0/CPU0:ios(config)#aaa accounting exec default start-stop group TACACS_ALL
```

Configure TACACS+ Server Groups

Configuring NCS 1014 to use AAA server groups provides a way to group existing server hosts. This allows you to select a subset of the configured server hosts and use them for a particular service. A server group is used in conjunction with a global server-host list. The server group lists the IP addresses of the selected server hosts.

You can enter one or more **server** commands. The **server** command specifies the hostname or IP address of an external TACACS+ server. Once configured, this server group can be referenced from the AAA method lists (used while configuring authentication, authorization, or accounting).

To configure TACACS+ server groups, perform these steps:

Before you begin

For successful configuration, the external server should be accessible at the time of configuration. When configuring the same IP address for global configuration, server-private parameters are required.

Procedure

Step 1 Enter into the IOS XR configuration mode.

Example:

```
RP/0/RP0/CPU0:ios# configure
```

Step 2 Create an AAA server-group and enter into the server group sub-configuration mode.

Example:

```
RP/0/RP0/CPU0:ios(config)# aaa group server tacacs+ tacgroup1
```

Step 3 Configure the IP address of the private TACACS+ server for the group server.

Example:

```
RP/0/RP0/CPU0:ios(config-sg-tacacs+)# server-private 10.1.1.1 port 49 key a_secret
```

Note

- You can configure a maximum of 10 TACACS+ private servers in a server group.
- If private server parameters are not specified, global configurations are used. If global configurations are not specified, default values are used.

Step 4 Configure the authentication and encryption key used between NCS 1014 and the TACACS+ daemon running on the TACACS+ server. If no key string is specified, the global value is used.

Example:

```
 RP/0/RP0/CPU0:ios(config-sg-tacacs+)# key 7 08984B1A4D0C19157A5F57
```

Step 5 Configure the timeout value that sets the length of time the authentication, authorization, and accounting (AAA) server waits to receive a response from the TACACS+ server.

Example:

```
RP/0/RP0/CPU0:ios(config-sg-tacacs-private)# timeout 4
```

Step 6 Repeat steps 3 to 5 for every private server to be added to the server group.

Step 7 Configure certificate-based authentication for users configured in the TACACS+ server or server groups.

Example:

```
RP/0/RP0/CPU0:ios(config-sg-tacacs-private)#aaa authorization exec default group TACACS_ALL local
```

Step 8 Set the default method list for authentication, and also enables authentication for console in global configuration mode.

Example:

```
RP/0/RP0/CPU0:ios(config-sg-tacacs-private)#aaa authentication login default group TACACS_ALL local
```

Step 9 Commit the changes and exit all the configuration modes.

commit

end

Step 10 Verify the TACACS+ server group configuration details.

Example:

```
RP/0/RP0/CPU0:ios# show tacacs server-groups
```

About RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a distributed client/server system that provides security against unauthorized access in distributed client/server networks. In Cisco's implementation, RADIUS clients operate on Cisco NCS 1014 and send requests for authentication and accounting to a central RADIUS server that contains all user authentication and network service access information.

Cisco's AAA security paradigm supports RADIUS, which can be used alongside other security protocols like TACACS+, Kerberos, and local username lookup.

Configure RADIUS Server Groups

You can enter one or more **server** commands. The **server** command specifies the hostname or IP address of an external RADIUS server along with port numbers. When configured, this server group can be referenced from the AAA method lists (used while configuring authentication, authorization, or accounting).

You can configure a maximum of 30 servers and private servers each per RADIUS server group. To configure RADIUS server groups, perform these tasks:

Before you begin

Ensure that the external server is accessible at the time of configuration.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:ios# configure
```

Enters mode.

Step 2 **aaa group server radius** *group-name*

Example:

```
RP/0/RP0/CPU0:ios(config)# aaa group server radius radgroup1
```

Groups different server hosts into distinct lists and enters the server group configuration mode.

Step 3 **radius-server** {*ip-address*}

Example:

```
RP/0/RP0/CPU0:ios(config)# radius-server host 192.168.20.0
```

Specifies the hostname or IP address of the RADIUS server host.

Step 4 **auth-port** *port-number*

Example:

```
RP/0/RP0/CPU0:ios(config)#auth-port 1812
```

Specifies the User Datagram Protocol (UDP) destination port for authentication requests; the host is not used for authentication if set to 0. If unspecified, the port number defaults to 1645.

Step 5 **acct-port** *port-number*

Example:

```
RP/0/RP0/CPU0:ios(config)# acct-port 1813
```

Specifies the UDP destination port for accounting requests; the host is not used for accounting if set to 0. If unspecified, the port number defaults to 1646.

Step 6 **key** *string*

Example:

```
RP/0/RP0/CPU0:ios(config-radius-host)#key 7 08984B1A4D0C19157A5F57
```

Specifies the authentication and encryption key used between NCS 1014 and the RADIUS server. This key overrides the global setting of the **radius-server key** command. If no key string is specified, the global value is used.

The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command syntax. This is because the leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

Step 7 Repeat steps 4 to 6 for every external radius server to be added to the server group.

—

Step 8 **aaa authentication** {**login**} {**default**} **group** *group-name* **local**

Example:

```
RP/0/RP0/CPU0:ios(config-radius-host)#aaa authentication login default group radius local
```

Specifies the default method list for authentication, and also enables authentication for console in global configuration mode.

Step 9 Use the **commit** or **end** command.

Step 10 **show radius server-groups** [*group-name* [**detail**]]

Example:

```
RP/0/RP0/CPU0:ios# show radius server-groups
```

(Optional) Displays information about each RADIUS server group that is configured in the system.

CHAPTER 8

Configure Access Control List

This chapter describes the Access Control List (ACL) and the procedures to configure ACL.

- Understand Access Control Lists, on page 102
- How an ACL Works, on page 103
- Apply ACLs, on page 105
- Configure an Ingress IPv4 ACL on Management Ethernet Interface, on page 105
- Configure an Egress IPv4 ACL on the Management Ethernet Interface, on page 106
- Configure an Ingress IPv6 ACL on the Management Ethernet Interface, on page 108
- Configure an Egress IPv6 ACL on the Management Ethernet Interface, on page 109
- Configure Extended Access Lists, on page 110
- Modify ACLs, on page 111

Understand Access Control Lists

Table 12: Feature History

Feature Name	Release Information	Feature Description
ACL on Management Port	Cisco IOS XR Release 7.11.1	Access Control List (ACL) feature enables you to permit or deny specific devices to connect to the management port and access NCS 1010 devices. This control enhances network security. Both IPv4 and IPv6 ACLs are supported on the management port. Commands added: • **ipv4-access-list** • **ipv4-access-group** • **show access-lists-ipv4** • **ipv6-access-list** • **ipv6-access-group** • **show access-lists-ipv6**

Access Control Lists (ACLs) perform packet filtering to control the packets that move through the network. These controls allow to limit the network traffic and restrict the access of users and devices to the network. ACLs have many uses, and therefore many commands accept a reference to an access list in their command syntax. An ACL consists of one or more access control entries (ACE) that collectively define the network traffic profile. Access control entries (ACE) are entries in an ACL that describe the access rights related to a particular security identifier or user.

There are 2 types of ACLs:

- Standard ACLs-Verifies only the source IP address of the packets. Traffic is controlled by the comparison of the address or prefix configured in the ACL, with the source address found in the packet.
- Extended ACLs-Verifies more than just the source address of the packets. Attributes such as destination address, specific IP protocols, User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) port numbers, Differentiated Services Code Point (DSCP), and so on are validated. Traffic is controlled by a comparison of the attributes stated in the ACL with those in the incoming or outgoing packets.

Purpose of ACLs

ACLs allow you to perform the following:

- Filter incoming or outgoing packets on an interface.
- Restrict the contents of routing updates.

- Limit debug output that is based on an address or protocol.
- Control vty access.

How an ACL Works

An ACL is a sequential list consisting of permit and deny statements that apply to IP addresses and upper-layer IP protocols. The ACL has a name by which it is referenced. Many software commands accept an ACL as part of their syntax.

An ACL can be configured and named; however, it does not take effect until the ACL is referenced by a command that accepts an ACL. Multiple commands can reference the same ACL. An ACL can control traffic arriving at the router or leaving the router, but not traffic originating at the router.

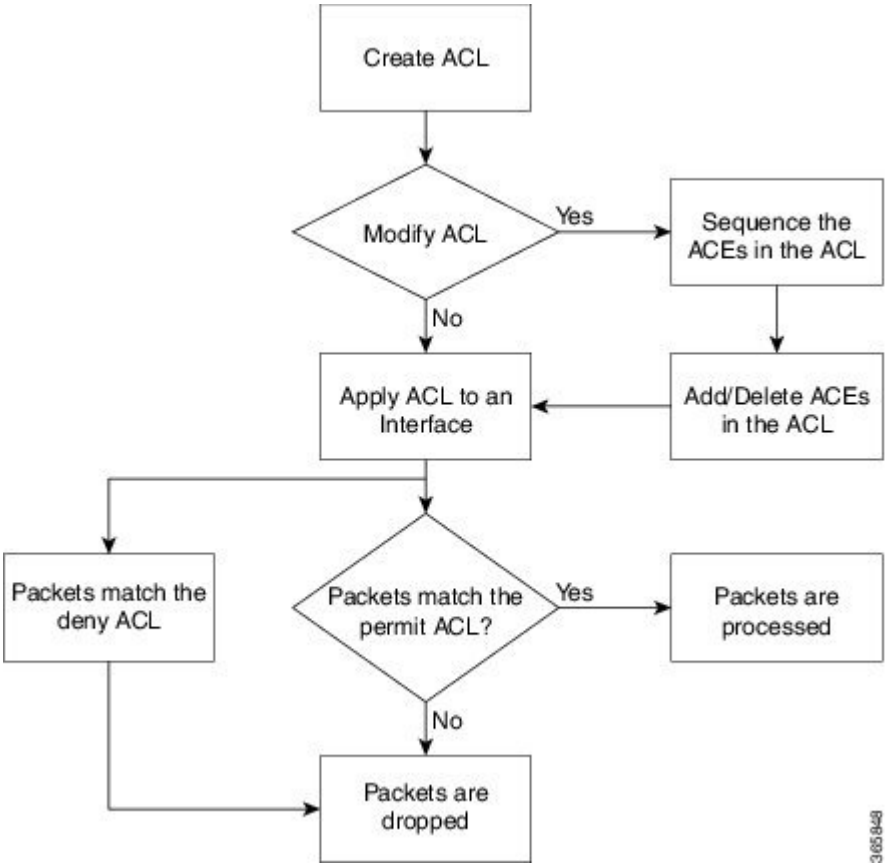
Source address and destination address are two of the most typical fields in an IP packet on which to base an ACL. Specify source addresses to control packets from certain networking devices or hosts. Specify destination addresses to control packets that are sent to certain networking devices or hosts.

You can also filter packets on the basis of transport layer information, such as whether the packet is a TCP, UDP, Internet Control Message Protocol (ICMP), or Internet Group Management Protocol (IGMP) packet.

ACL Workflow

The following image illustrates the workflow of an ACL.

Figure 2: ACL Workflow



Helpful Hints for Creating ACLs

Consider the following when creating ACLs:

- Create the access list before applying it to an interface.
- Organize your access list so that more specific references in a network or subnet appear before more general ones.
- To make the purpose of individual statements more easily understood at a glance, you can write a helpful remark before or after any statement.

Guidelines and Restrictions for Configuring ACLs

You must be aware of the following restrictions for configuring ACLs.

- Modifying an ACL when it is attached to the interface is supported.
- You can configure an ACL name with a maximum of 64 characters.
- You can configure an ACL name to comprise of only letters and numbers.

Apply ACLs

After you create an ACL, you must reference the ACL to make it work. ACL can be applied on *either* outbound or inbound interfaces. This section describes guidelines on how to accomplish this task for both terminal lines and network interfaces.

For inbound ACLs, after receiving a packet, Cisco IOS XR software checks the source address of the packet against the ACL. If the ACL permits the address, the software continues to process the packet. If the ACL rejects the address, the software discards the packet and returns an ICMP host unreachable message. The ICMP message is configurable.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the software checks the source address of the packet against the ACL. If the ACL permits the address, the software sends the packet. If the ACL rejects the address, the software discards the packet and returns an ICMP host unreachable message.

When you apply an ACL that has not yet been defined to an interface, the software acts as if the ACL has not been applied to the interface and accepts all packets. Note this behavior if you use undefined ACLs as a means of security in your network.

Configure an Ingress IPv4 ACL on Management Ethernet Interface

Use the following configuration to configure an ingress IPv4 ACL on mgmtEth interface.

```
/* Configure mgmtEth interface with an IPv4 address */
RP/0/RP0/CPU0:ios#configure
Thu Oct 19 17:30:23.719 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:ios(config-if)#interface mgmtEth 0/RP0/CPU0/2
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 198.51.100.247 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#no shutdown
RP/0/RP0/CPU0:ios(config-if)#commit
Thu Oct 19 17:31:25.127 UTC
RP/0/RP0/CPU0:ios(config-if)#exit


/* Verify if the interface is up */
RP/0/RP0/CPU0:ios(config)#do show ipv4 interface brief
Thu Oct 19 17:32:10.998 UTC

Interface                      IP-Address      Status          Protocol Vrf-Name
Loopback0                      192.0.2.89      Up              Up       default
GigabitEthernet0/0/0/0         198.51.100.1    Up              Up       default
GigabitEthernet0/0/0/2         198.51.100.10   Up              Up       default
MgmtEth0/RP0/CPU0/0            198.51.100.247  Up              Up       default
PTP0/RP0/CPU0/0                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/1            192.0.2.121     Up              Up       default
PTP0/RP0/CPU0/1                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/2            192.0.2.1       Down            Down     default


/* Configure an IPv4 ingress ACL */
RP/0/RP0/CPU0:ios(config)# ipv4 access-list V4-ACL-INGRESS
RP/0/RP0/CPU0:ios(config-ipv4-acl)# 10 permit tcp 192.0.2.2 255.255.255.0 any
```

```
RP/0/RP0/CPU0:ios(config-ipv4-acl)# 20 deny udp any any
RP/0/RP0/CPU0:ios(config-ipv4-acl)# 30 permit ipv4 192.0.2.64 255.255.255.0 any
RP/0/RP0/CPU0:ios(config-ipv4-acl)# commit
Thu Oct 19 18:31:25.127 UTC


/* Verify the ingress ACL creation */
RP/0/RP0/CPU0:ios(config)# do show access-lists ipv4
Thu Oct 19 18:32:25.127 UTC
...
ipv4 access-list V4-ACL-INGRESS
 10 permit tcp 192.0.2.2 255.255.255.0 any
 20 deny udp any any
 30 permit ipv4 192.0.2.64 255.255.255.0 any


/* Apply the ingress ACL to the mgmtEth interface */
RP/0/RP0/CPU0:ios(config)# interface mgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:ios(config-if)# ipv4 access-group V4-ACL-INGRESS ingress
RP/0/RP0/CPU0:ios(config-if)#commit
Thu Oct 19 18:34:25.127 UTC
RP/0/RP0/CPU0:ios(config-if)#exit


/* Verify if the ingress ACL has been successfully applied to the interface */

RP/0/RP0/CPU0:ios#show ipv4 interface mgmtEth 0/RP0/CPU0/0
Fri Oct 20 05:07:06.383 UTC
MgmtEth0/RP0/CPU0/0 is Up, ipv4 protocol is Up
  Vrf is default (vrfid 0x60000000)
  Internet address is 198.51.100.247/16
  MTU is 1514 (1500 is available to IP)
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  common access list is not set, access list is V4-ACL-INGRESS
  Proxy ARP is disabled
  ICMP redirects are never sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  Table Id is 0xe0000000
```

You have successfully configured an IPv4 ingress ACL on the mgmtEth interface.

Configure an Egress IPv4 ACL on the Management Ethernet Interface

Use the following configuration to configure an egress IPv4 ACL on the mgmtEth interface.

```
/* Configure mgmtEth interface with an IPv4 address */
RP/0/RP0/CPU0:ios#configure
Thu Oct 19 17:30:23.719 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 198.51.100.247 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#no shutdown
RP/0/RP0/CPU0:ios(config-if)#commit
Thu Oct 19 17:31:25.127 UTC
RP/0/RP0/CPU0:ios(config-if)#exit
```

```
/* Verify if the interface is up */
RP/0/RP0/CPU0:ios(config)#do show ipv4 interface brief
Thu Oct 19 17:32:10.998 UTC

Interface                      IP-Address      Status          Protocol Vrf-Name
Loopback0                      192.0.2.89      Up              Up       default
GigabitEthernet0/0/0/0         198.51.100.1    Up              Up       default
GigabitEthernet0/0/0/2         198.51.100.10   Up              Up       default
MgmtEth0/RP0/CPU0/0            198.51.100.247  Up              Up       default
PTP0/RP0/CPU0/0                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/1            192.0.2.121     Up              Up       default
PTP0/RP0/CPU0/1                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/2            192.0.2.1       Down            Down     default


/* Configure an IPv4 egress ACL */
RP/0/RP0/CPU0:ios(config)# ipv4 access-list V4-ACL-EGRESS
RP/0/RP0/CPU0:ios(config-ipv4-acl)# 10 permit ipv4 203.0.113.1 255.255.255.0 192.0.2.1
0.255.255.255
RP/0/RP0/CPU0:ios(config-ipv4-acl)# 20 deny udp any any
RP/0/RP0/CPU0:ios(config-ipv4-acl)# commit
Thu Oct 19 18:31:25.127 UTC


/* Verify the egress ACL creation */
RP/0/RP0/CPU0:ios(config)# do show access-lists ipv4
Thu Oct 19 18:32:25.127 UTC
ipv4 access-list V4-ACL-EGRESS
 10 permit ipv4 203.0.113.1 255.255.255.0 192.0.2.1 255.255.255.0
 20 deny ipv4 any any
...


/* Apply the egress ACL to the mgmtEth interface */
RP/0/RP0/CPU0:ios(config)# interface mgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:ios(config-if)# ipv4 access-group V4-ACL-EGRESS egress
RP/0/RP0/CPU0:ios(config-if)# commit
Thu Jul 11 09:19:49.569 UTC
RP/0/RP0/CPU0:ios(config-if)# exit


/* Verify if the egress ACL has been successfully applied to the interface */

RP/0/RP0/CPU0:ios#show ipv4 interface mgmtEth 0/RP0/CPU0/0
Fri Oct 20 05:07:06.383 UTC
MgmtEth0/RP0/CPU0/0 is Up, ipv4 protocol is Up
  Vrf is default (vrfid 0x60000000)
  Internet address is 198.51.100.247/16
  MTU is 1514 (1500 is available to IP)
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is V4-ACL-EGRESS
  Inbound  common access list is not set, access list is not set
  Proxy ARP is disabled
  ICMP redirects are never sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  Table Id is 0xe0000000
```

You have successfully configured an IPv4 egress ACL on the mgmtEth interface.

Configure an Ingress IPv6 ACL on the Management Ethernet Interface

Use the following configuration to configure an ingress IPv6 ACL on the mgmtEth interface.

```
/* Configure a mgmtEth interface with an IPv6 address */
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#configure
Fri Oct 20 05:25:58.785 UTC
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/1
RP/0/RP0/CPU0:ios(config-if)#ipv6 address 2001::1/64
RP/0/RP0/CPU0:ios(config-if)#no shutdown
RP/0/RP0/CPU0:ios(config-if)#commit
Fri Oct 20 05:26:13.669 UTC
RP/0/RP0/CPU0:ios(config-if)#exit


/* Verify if the interface is up */
RP/0/RP0/CPU0:ios(config)#do show ipv6 interface mgmtEth 0/RP0/CPU0/1 brief
Fri Oct 20 05:26:52.612 UTC
MgmtEth0/RP0/CPU0/1    [Up/Up]
    fe80::3afd:f8ff:fe66:872
    2001::1


/* Configure an IPv6 ingress ACL */
RP/0/RP0/CPU0:ios(config)#ipv6 access-list V6-INGRESS-ACL
RP/0/RP0/CPU0:ios(config-ipv6-acl)#10 permit ipv6 any any
RP/0/RP0/CPU0:ios(config-ipv6-acl)#20 deny udp any any
RP/0/RP0/CPU0:ios(config-ipv6-acl)#commit
Fri Oct 20 05:28:46.664 UTC
RP/0/RP0/CPU0:ios(config-ipv6-acl)#exit


/* Verify the ingress ACL creation */
RP/0/RP0/CPU0:ios(config)#do show access-lists ipv6
Fri Oct 20 05:29:01.125 UTC
ipv6 access-list V6-INGRESS-ACL
 10 permit ipv6 any any
 20 deny udp any any


/* Apply the ingress ACL to the HundredGigE interface */
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/1
RP/0/RP0/CPU0:ios(config-if)#ipv6 access-group V6-INGRESS-ACL ingress
RP/0/RP0/CPU0:ios(config-if)#commit
Fri Oct 20 05:37:32.738 UTC
RP/0/RP0/CPU0:ios(config-if)#exit


/* Verify if the ingress ACL has been successfully applied to the interface */
RP/0/RP0/CPU0:ios(config)#do show ipv6 interface mgmtEth 0/RP0/CPU0/1
Fri Oct 20 05:38:00.753 UTC
MgmtEth0/RP0/CPU0/1 is Up, ipv6 protocol is Up, Vrfid is default (0x60000000)
  IPv6 is enabled, link-local address is fe80::3afd:f8ff:fe66:872
  Global unicast address(es):
    2001::1, subnet is 2001::/64
  Joined group address(es): ff02::1:ff00:1 ff02::1:ff66:872 ff02::2
      ff02::1
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are disabled
```

```
ICMP unreachables are enabled
ND DAD is enabled, number of DAD attempts 1
ND reachable time is 0 milliseconds
ND cache entry limit is 1000000000
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 160 to 240 seconds
ND router advertisements live for 1800 seconds
Hosts use stateless autoconfig for addresses.
Outgoing access list is not set
Inbound  common access list is not set, access list is V6-INGRESS-ACL
Table Id is 0xe0800000
Complete protocol adjacency: 0
Complete glean adjacency: 0
Incomplete protocol adjacency: 0
Incomplete glean adjacency: 0
Dropped protocol request: 0
Dropped glean request: 0
RA DNS Server Address Count: 0
RA DNS Search list Count: 0
```

You have successfully configured an IPv6 ingress ACL on the mgmtEth interface.

Configure an Egress IPv6 ACL on the Management Ethernet Interface

Use the following configuration steps to configure an egress IPv6 ACL on the mgmtEth interface.

```
/* Configure a mgmtEth interface with an IPv6 address */
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/1
RP/0/RP0/CPU0:ios(config-if)#ipv6 address 2001::1/64
RP/0/RP0/CPU0:ios(config-if)#no shutdown
RP/0/RP0/CPU0:ios(config-if)#commit
Thu Jan 25 11:41:25.778 IST
Router(config-if)# exit


/* Verify if the interface is up */
Router(config)# do show ipv6 interface brief
Thu Jul 11 09:47:50.812 UTC
HundredGigE 0/0/0/0 [Up/Up]
    fe80::bd:b9ff:fea9:5606
    1001::1
HundredGigE 0/0/0/1 [Up/Up]
    fe80::23:e9ff:fea8:a44e
    2001::1


/* Configure an IPv6 egress ACL */
Router(config)# ipv6 access-list V6-EGRESS-ACL
Router(config-ipv6-acl)# 10 permit ipv6 any any
Router(config-ipv6-acl)# 20 deny udp any any
Router(config-ipv6-acl)# commit
Thu Jul 11 09:50:40.566 UTC
Router(config-ipv6-acl)# exit


/* Verify the egress ACL creation */
RP/0/RP0/CPU0:ios(config)#do show ipv6 interface mgmtEth 0/RP0/CPU0/1 brief
Fri Oct 20 05:26:52.612 UTC
MgmtEth0/RP0/CPU0/1    [Up/Up]
```

```
    fe80::3afd:f8ff:fe66:872
    2001::1
…

/* Apply the egress ACL to the mgmtEth interface */
Router(config)# interface mgmtEth 0/RP0/CPU0/1
Router(config-if)# ipv6 access-group V6-EGRESS-ACL egress
Router(config-if)# commit
Thu Jul 11 09:52:57.751 UTC
Router(config-if)# exit


/* Verify if the egress ACL has been successfully applied to the interface */
RP/0/RP0/CPU0:ios(config)#do show ipv6 interface mgmtEth 0/RP0/CPU0/1
Fri Oct 20 05:38:00.753 UTC
MgmtEth0/RP0/CPU0/1 is Up, ipv6 protocol is Up, Vrfid is default (0x60000000)
  IPv6 is enabled, link-local address is fe80::3afd:f8ff:fe66:872
  Global unicast address(es):
    2001::1, subnet is 2001::/64
  Joined group address(es): ff02::1:ff00:1 ff02::1:ff66:872 ff02::2
      ff02::1
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are disabled
  ICMP unreachables are enabled
  ND DAD is enabled, number of DAD attempts 1
  ND reachable time is 0 milliseconds
  ND cache entry limit is 1000000000
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 160 to 240 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
  Outgoing access list is V6-EGRESS-ACL
  Inbound  common access list is not set, access list is not set
  Table Id is 0xe0800000
  Complete protocol adjacency: 0
  Complete glean adjacency: 0
  Incomplete protocol adjacency: 0
  Incomplete glean adjacency: 0
  Dropped protocol request: 0
  Dropped glean request: 0
  RA DNS Server Address Count: 0
  RA DNS Search list Count: 0
...
```

You have successfully configured an IPv6 egress ACL on the mgmtEth interface.

Configure Extended Access Lists

Use Extended Access Lists to verify more than just the source address of the packets. Attributes such as destination address, specific IP protocols, UDP or TCP port numbers, DSCP, and so on are validated. Traffic is controlled by a comparison of the attributes stated in the ACL with those in the incoming or outgoing packets.

To configure Extended Access Lists, you must create an access list and specify the condition to allow or deny the network traffic.

```
/* Enter the global configuration mode and create the access list*/
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#ipv4 access-list acl_1
RP/0/RP0/CPU0:ios(config-ipv4-acl)#10 remark Do not allow user1 to telnet out
```

```
/*Specify the condition to allow or deny the network traffic.*/
RP/0/RP0/CPU0:ios(config-ipv4-acl)#10 permit 172.16.0.0 0.0.255.255
RP/0/RP0/CPU0:ios(config-ipv4-acl)#20 deny 192.168.34.0 0.0.0.255
RP/0/RP0/CPU0:ios(config-ipv4-acl)#commit
```

Running Configuration

```
RP/0/RP0/CPU0:ios#show running-config
Fri Oct 20 06:21:11.024 UTC
!! Building configuration...
!! IOS XR Configuration 24.1.1.23I
!! Last configuration change at Fri Oct 20 06:19:08 2023 by cisco

!
ipv4 access-list acl_1
 10 permit ipv4 172.16.0.0 0.0.255.255 any
 20 deny ipv4 192.168.34.0 0.0.0.255 any
!
```

Verification

```
RP/0/RP0/CPU0:ios#show access-lists ipv4 acl_1
Fri Oct 20 06:22:17.223 UTC
ipv4 access-list acl_1
 10 permit ipv4 172.16.0.0 0.0.255.255 any
 20 deny ipv4 192.168.34.0 0.0.0.255 any
```

Modify ACLs

This section describes a sample configuration to modify ACLs.

```
*/ Create an Access List*/
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#ipv4 access-list acl_1

*/Add entries (ACEs) to the ACL*/
RP/0/RP0/CPU0:ios(config-ipv4-acl)#10 permit ip host 10.3.3.3 host 172.16.5.34
RP/0/RP0/CPU0:ios(config-ipv4-acl)#20 permit icmp any any
RP/0/RP0/CPU0:ios(config-ipv4-acl)#30 permit tcp any host 10.3.3.3
RP/0/RP0/CPU0:ios(config-ipv4-acl)#end


*/Verify the entries of the ACL*/:
Router#show access-lists ipv4 acl_1
ipv4 access-list acl_1
10 permit ip host 10.3.3.3 host 172.16.5.34
20 permit icmp any any
30 permit tcp any host 10.3.3.3


*/Add new entries, one with a sequence number "15" and another without a sequence number
to the ACL. Delete an entry with the sequence number "30":*/
RP/0/RP0/CPU0:ios(config)#ipv4 access-list acl_1
RP/0/RP0/CPU0:ios(config-ipv4-acl)# 15 permit 10.5.5.5 0.0.0.255
RP/0/RP0/CPU0:ios(config-ipv4-acl)# no 30
RP/0/RP0/CPU0:ios(config-ipv4-acl)# permit 10.4.4.4 0.0.0.255
RP/0/RP0/CPU0:ios(config-ipv4-acl)# commit
```

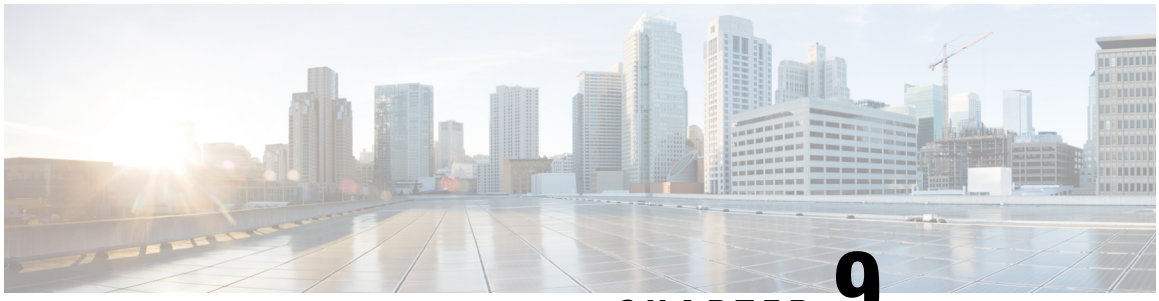
/When an entry is added without a sequence number, it is automatically given a sequence number that puts it at the end of the access list. Because the default increment is 10, the entry will have a sequence number 10 higher than the last entry in the existing access list/

```
*/Verify the entries of the ACL:*/
RP/0/RP0/CPU0:ios#show access-lists ipv4 acl_1
ipv4 access-list acl_1
 10 permit ipv4 host 10.3.3.3 host 172.16.5.34

15 permit 10.5.5.5 0.0.0.255---*/newly added ACL (with the sequence number)*/
20 permit icmp any any
30 permit ipv4 10.4.4.0 0.0.0.255 any ---*/newly added ACL (without the sequence number)*/

*/The entry with the sequence number 30, that is, "30 permit tcp any host 10.3.3.3" is
deleted from the ACL*/
```

You have successfully modified ACLs in operation.



CHAPTER 9

Smart Licensing

This chapter describes the smart licensing configuration on Cisco NCS 1014.

- Understanding Smart Licensing, on page 113
- Create an ID Token, on page 115
- Smart Licensing Transport Modes, on page 116
- Reserve Specific Licenses for NCS 1014, on page 120
- Smart Licensing for QXP Line Card, on page 121

Understanding Smart Licensing

Smart Licensing is a cloud-based, software license management solution that enables you to automate time-consuming, manual licensing tasks. The solution allows you to easily track the status of your license and software usage trends.

Smart Licensing helps you simplify three core functions:

- **Purchasing**: The software that you have installed in your network can be registered, without Product Activation Keys (PAKs).
- **Management**: You can automatically track activations against your license entitlements. Also, there is no need to install the license file on every node. You can create license pools (logical grouping of licenses) to reflect your organization structure. Smart Licensing offers you Cisco Smart Software Manager, a centralized portal that enables you to manage all your Cisco software licenses from one centralized website.
- **Reporting**: Through the portal, Smart Licensing offers an integrated view of the licenses you have purchased and what has been deployed in your network. You can use this data to make better purchasing decisions, based on your consumption.

Smart Licensing Features

- Your device initiates a call home and requests the licenses it needs.
- Pooled licenses - Licenses are company account-specific, and can be used with any compatible device in your company. You can activate or deactivate different types of licenses on the device without actually installing a license file on the device.
- Licenses are stored securely on Cisco servers.

- Licenses can be moved between product instances without license transfer. This greatly simplifies the reassignment of a software license as part of the Return Material Authorization (RMA) process.
- It provides a complete view of all the Smart Software Licenses used in the network using a consolidated usage report of software licenses and devices in one easy-to-use portal.

Cisco Smart Account

Cisco Smart Account is an account where all products enabled for Smart Licensing are deposited. Cisco Smart Account allows you to manage and activate your licenses to devices, monitor license use, and track Cisco license purchases. Through transparent access, you have a real-time view into your Smart Licensing products. IT administrators can manage licenses and account users within your organization's Smart Account through the Smart Software Manager.

When creating a Smart Account, you must have the authority to represent the requesting organization. After you submit the request, it goes through a brief approval process. Access http://software.cisco.com to learn about, set up, or manage Smart Accounts.

Cisco Smart Software Manager enables you to manage all your Cisco Smart software licenses from one centralized website. With Cisco Smart Software Manager, you organize and view your licenses in groups called virtual accounts (collections of licenses and product instances). Use the Cisco Smart Software Manager to do the following tasks:

- Create, manage, or view virtual accounts.
- Create and manage Product Instance ID Tokens.
- Transfer licenses between virtual accounts or view licenses.
- Transfer, remove, or view product instances.
- Run reports against your virtual accounts.
- Modify your email notification settings.
- View overall account information.

Virtual Accounts

A Virtual Account exists as a subaccount tithing the Smart Account. Virtual Accounts are a customer-defined structure based on organizational layout, business function, geography, or any defined hierarchy. They are created and maintained by the Smart Account administrator. Smart Licencing allows you to create multiple license pools or virtual accounts within the Smart Software Manager portal. Using the Virtual Accounts option that you can aggregate licenses into discrete bundles that are associated with a cost center so that one section of an organization cannot use the licenses of another section of the organization. For example, if you segregate your company into different geographic regions, you can create a virtual account for each region to hold the licenses and product instances for that region.

All new licenses and product instances are placed in the default virtual account in the Smart Software Manager, unless you specify a different one during the order process. After you access the default account, you may choose to transfer them to any other account, provided you have the required access permissions.

Use the Smart Software Manager portal to create license pools or transfer licenses.

Product Instance ID Tokens

ID tokens are stored in the Product Instance ID Token Table that is associated with your enterprise account. ID tokens can be valid 1–365 days.

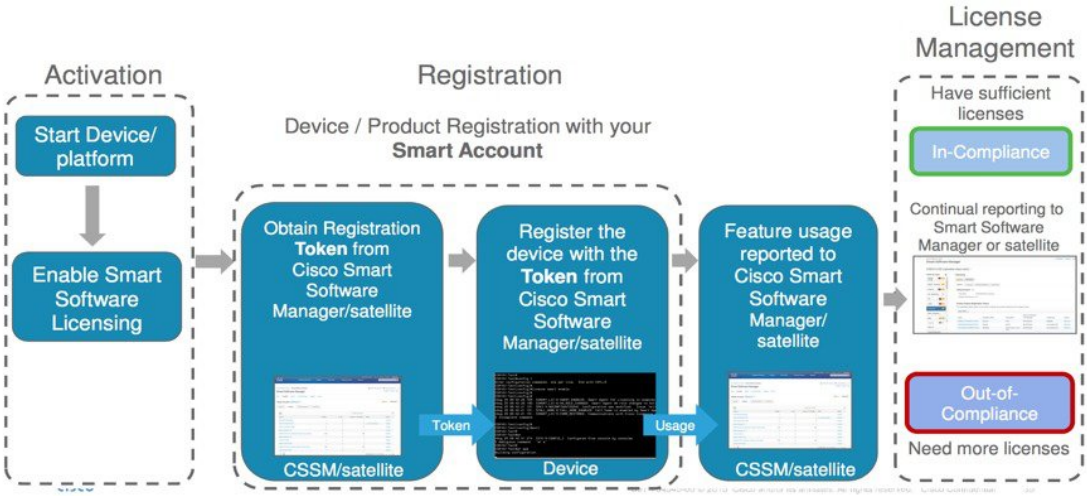
Product Instances

A product instance is an individual device with a unique device identifier (UDI) that is registered using a product instance ID token (or ID token). You can register any number of instances of a product with a single ID token. Each product instance can have one or more licenses residing in the same virtual account. Product instances must periodically connect to the Cisco Smart Software Manager servers during a specific renewal period. If you remove the product instance, its licenses are released and made available within the virtual account.

Smart Licensing Work Flow

The following figure depicts a working model of smart licensing that involves a three-step procedure.

Figure 3: Smart Licensing Work Flow



Create an ID Token

To create a new token using Cisco Smart Software Manager, perform the following tasks:

Before you begin

Procedure

Step 1 Log in to the Cisco Smart Software Manager.

https://software.cisco.com/software/csws/ws/platform/home#SmartLicensing-Inventory

Step 2 Click the **Inventory** tab, and select your virtual account from the **Virtual Account** drop-down list.

Step 3 Click the **General** tab, and click **New Token**.

The **Create ID Token** window is displayed.

Step 4 Enter the token description. Specify the number of days the token must be active.

Step 5 Check the **Allow export-controlled functionality on the products registered with this token** check box.

Step 6 Click **Create ID Token**.

Step 7 Copy the token and register NCS1014 with the same token ID.

An example of the token ID: YzY2ZjYyNjktY2NlOS00NTc4LWIxNTAtMjZkNmNiNzMxMTY1LTE2NjAzNjQ3%0ANzY4Njl8ZVJSckxKN2pFV2tIeHVoMUkxbGxTazFDVm9kc1B5MGlHQmlFWUJi%0Ac3VNRT0%3D%0A

Smart Licensing Transport Modes

Smart Licensing software management solution enables you to choose from one of the three transport modes, Cisco Smart Licensing Utility(CSLU), Smart Transport or Offline modes. This is in addition to the existing Call-Home mode. The default transport mode is CSLU, but you can change the mode to Call-Home, Smart Transport or Offline mode.

The following transport modes are available for you to choose now:

- Call-Home
- Smart
- CSLU
- Offline

Configure Callhome

You can use the Call Home to connect to the CSSM. To configure callhome in Cisco NCS 1014, perform the following steps:

Procedure

Step 1 Use this sample configuration to enable call home mode settings.

Example:

```
RP/0/RP0/CPU0:ios#call-home
RP/0/RP0/CPU0:ios(config-call-home)#service active
RP/0/RP0/CPU0:ios(config-call-home)#contact smart-licensing
RP/0/RP0/CPU0:ios(config-call-home)#profile CiscoTAC-1
RP/0/RP0/CPU0:ios(config-call-home-profile)#active
RP/0/RP0/CPU0:ios(config-call-home-profile)#destination address http
https://tools.cisco.com/its/service/oddce/services/DDCEService
RP/0/RP0/CPU0:ios(config-call-home-profile)#reporting smart-call-home-data
RP/0/RP0/CPU0:ios(config-call-home-profile)#reporting smart-licensing-data
RP/0/RP0/CPU0:ios(config-call-home-profile)#destination transport-method email disable
RP/0/RP0/CPU0:ios(config-call-home-profile)#destination transport-method http
RP/0/RP0/CPU0:ios(config-call-home-profile)#commit
RP/0/RP0/CPU0:ios(config-call-home-profile)#end
```

Step 2 Use this sample configuration to enable a domine name server.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#domain name cisco.com
RP/0/RP0/CPU0:ios(config)#domain name-server 64.102.6.247
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 3 Use this sample configuration to enable CRL Configuration.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 4 Use this sample configuration to enable Call Home as transport mode.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport callhome
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Verify whether the Callhome is Configured.

```
RP/0/RP0/CPU0:ios#show license all
Transport:  Type: Callhome
```

Step 5 Use this sample configuration to establish trust using id-token.

Example:

```
license smart trust idtoken Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n
```

Configure Smart

You can use the smart transport as an alternative option to Call Home, to connect to the CSSM. To configure smart transport in Cisco NCS 1014, perform the following steps:

Procedure

Step 1 Use this sample configure "Smart" proxy and "hostname"

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#license smart proxy port 80
RP/0/RP0/CPU0:ios(config)#license smart proxy hostname proxy.esl.cisco.com
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Use this sample configuration to enable CRL Configuration.

Example:

```
RP/0/RP0/CPU0:ios#config
```

```
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 3 Use this sample configuration to enable Call Home.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport smart
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Verify whether Smart Transport is Configured.

```
RP/0/RP0/CPU0:ios#show license all
Transport:
  Type: Smart
  URL: https://smartreceiver.cisco.com/licservice/license
  Proxy:
    Address: proxy.esl.cisco.com
    Port: 80
    Username: <empty>
    Password: <empty>
  VRF:
    Not Supported
```

Step 4 Use this sample configuration to establish trust using id-token.

Example:

```
license smart trust idtoken Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n
```

Configure CSLU

You can configure CSLU as one of the transport modes, CSLU is the default mode for software licensing policy. To configure CSLU in Cisco NCS 1014, perform the following steps:

Procedure

Step 1 Use this sample configuration to configure the CSLU URL.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#license smart url cslu http://10.127.60.58:8182/cslu/v1/pi
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Use this sample configuration to enable CRL Configuration.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 3 Use this sample configuration to enable CSLU.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport cslu
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Verify whether CSLU is Configured.

```
RP/0/RP0/CPU0:ios#show license all
Transport:
  Type: cslu
  Cslu address: http://10.127.60.58:8182/cslu/v1/pi
  Proxy:
    Not Configured
  VRF:
    Not Supported
```

Step 4 Use this sample configuration to establish trust using id-token.

Example:

```
license smart trust idtoken Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n
```

Configure Offline

You can configure Offline as one of the options. To configure Offline in Cisco NCS 1014, perform the following steps:

Procedure

Step 1 Use this sample configuration to disable transport.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport off
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Use this sample configuration to save the report.

Example:

```
RP/0/RP0/CPU0:ios#license smart save usage unreported /misc1/disk1/usage.txt
```

Step 3 Use this sample configuration to import the acknowledgment report.

Example:

```
RP/0/RP0/CPU0:ios#license smart import /misc/disk1/ACK_usage.txt
```

Reserve Specific Licenses for NCS 1014

Specific License Reservation (SLR) lets you reserve a license for your product instance from the CSSM. To reserve specific licenses for NCS 1014, perform the following steps:

Procedure

Step 1 Generate the request code using the **license smart mfg reservation request** local command.

Example:

```
RP/0/RP0/CPU0:iso#license smart mfg reservation request local
Thu Jul 19 13:33:47.241 UTC

Enter this request code in the Cisco Smart Software Manager portal:
CB-ZNCS1014-SA:FCB2546B08T-BBTQDthRu-BA
```

Step 2 Use the generated code and generate the authorization code through Cisco Smart Software Manager.

Step 3 Enter the **run** command to launch the iso XR Linux bash shell.

Example:

```
RP/0/RP0/CPU0:iso#run

RP/0/RP0/CPU0:Jul 19 13:35:20.236: run_cmd[67213]: %INFRA-INFRA_MSG.5-RUN_LOGIN : User Cisco logged
 into shell from con0/RP0/CP0
```

Step 4 Create a file using the **vim** *file name* command.

Example:

```
[node0_RP0_CPU0:~]$vim smart1
```

Step 5 Copy the authorization code in the file and type **:wq** to save and exit the file.

Step 6 Use the **exit** command to exit the shell.

Example:

```
[node0_RP0_CPU0:~]$exit
logout
RP/0/RP0/CPU0:Jul 19 13:45:21.146 UTC run-cmd[67213] %INFRA_MSG-5-LOGOUT : User cisco logged out of
 shell from con0/RP0/CPU0
```

Step 7 Install the authorization code using the **license smart reservation install file** command.

Example:

```
RP/0/RP0/CPU0:iso#license smart reservation install file /disk0:/smart1
Thu Jul 19 13:46:22.877 UTC
Last Confirmation code 8572aa81
```

Note
You can verify the number of reservations in the Cisco smart software manger portal and can view the product instance name changed to a UDI.

Step 8 Verify the udi using the **show license udi** command.

Example:

```
RP/0/RP0/CPU0:iso#show license udi
Thu Jul 19 13:43:19.731 UTC
UDI: PID:NCS1014-SA,SN:FCB2546B08T
```

Step 9 Verify the license reservation using the command **show license status**.

Example:

```
RP/0/RP0/CPU0:P2A_DT_08#show license status
Thu Jul 19 15:45:27.137 UTC

Smart Licensing is ENABLED

Utility:
  Status: DISABLED
License Reservation is ENABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Transport Off


License Authorization:
  Status: AUTHORIZED - RESERVED on Jul 19 2022 15:21:24 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Miscellaneous:
  Custom Id: <empty>
```

Smart Licensing for QXP Line Card

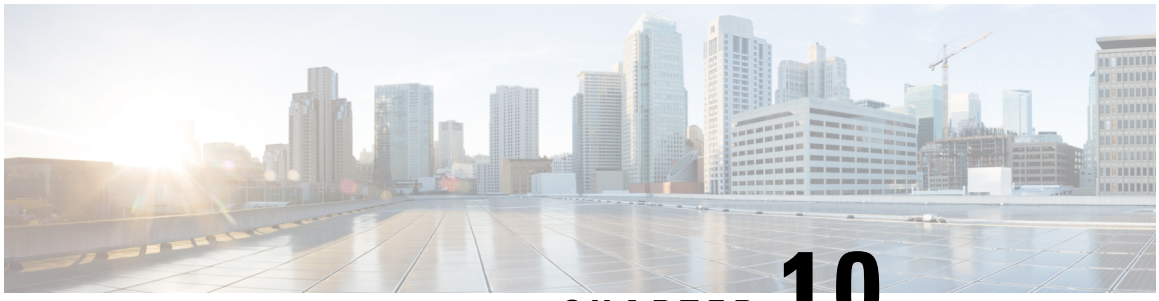
Table 13: Feature History

Feature Name	Release Information	Feature Description
Smart Licensing for QXP Line Card	Cisco IOS XR Release 24.3.1	Now the NCS1K4-QXP-L-K9 supports the smart licensing feature. It enables you to automate the time-consuming manual licensing tasks and allows you to easily track the status of your license and software usage trends.

The NCS1K4-QXP-L-K9 now supports Smart Licensing, allowing you to manage licenses and monitor usage trends with ease.

Table 14: License Entitlements for NCS1K4-QXP-L-K9

Display Name in CSSM Server	Description
S_N1K4_LIC_TRK	NCS 1K4 Smart License QDDTXP Trunk
NCS1014_ESS_TXP_RTU	Essential Tier Transponder RTU
NCS1014_ESS_TXP_SIA	Essentials Subscription Transponder SIA



CHAPTER 10

Automated File Management

- Automated File Management System, on page 123

Automated File Management System

Table 15: Feature History

Feature Name	Release Information	Feature Description
Automated File Management System	Cisco IOS XR Release 24.4.1	The new Automated File Management System is designed for efficient file handling on each node. This system automatically archives older files and removes them from local nodes to free up valuable SSD space. It manages the following types of files: • System-generated log files • Showtech-related residual files

The automated file management system archives older files to free up valuable SSD space by deleting them from the local nodes.

Types of files

The SSD stores two types of files that are generated by

- **User**: creates and owns files for requirement purposes and deletes the files when are no longer needed.
- **System**: organizes files automatically based on the file content and the application that created the content such as .log and showtech-related residual files.

  Automated file management is applicable for the system-generated files.

How the automated file management system works

These stages describe how the automated file management works for various files.

Log files

The NCS 1014 system uses the log rotation configurations to manage log files as required.

1. The system checks for the *.log* files exceeding 10 MB file size.

Note This threshold is applicable for `/tmp` folder files only. For files in other folders, the system uses a different threshold and follows the same process.

2. After locating the file, the system
   a. archives that file with .gz extension, and
   b. creates a new .log file with the same name.
3. The system then maintains one active log file and two archived files.
4. When new files are archived, the system deletes the oldest archived file to make sure that only the two most recently archived files and the current log file are retained.

showtech-related residual log files

1. The system generates the residual files during the collection of logs when **Showtech logs** is in operation.
2. After collecting the logs, the system automatically removes these residual files to conserve space.