System Setup and Software Installation Guide for Cisco NCS 1001

Cisco Network Convergence System 1001
Updated July 31, 2026

Full Cisco Trademarks with Software License

Full Cisco Trademarks with Software License

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)

Topics included

1 Cisco NCS 1001 Boot Sources

Topics:

Provides comprehensive procedures for booting, initializing, and configuring the Cisco NCS 1001, including details on boot sources, Zero Touch Provisioning, console management, and essential system network settings.

Cisco NCS 1001 checks local and remote boot sources in a defined order to locate an available system image.

Cisco NCS 1001 checks these boot sources in order:

- Solid-state drive (SSD), which is the internal hard disk
- USB drive
- iPXE network boot

> **Note**
>
> If none of the boot sources contains a bootable image, reboot the system and verify that an image is available from one of the sources.

Cisco NCS 1001 bring-up processes

This section explains how the Cisco NCS 1001 starts after hardware installation and identifies the follow-on setup tasks needed to reach a manageable system state.

A Cisco NCS 1001 bring-up process is a system initialization procedure that

- starts the device through the XR console port after hardware installation,
- uses the preinstalled operating system image when one is available, and
- supports recovery boot methods, such as iPXE, USB boot, and Golden ISO, when no valid local image is available.

After the boot process, create the root username and password and use those credentials to log in to the XR console. From the XR console, access the System Admin console to configure system administration settings.

> **Note**
>
> The output in command examples might not be from the latest software release. Output can change when a release-specific value changes.

Boot sequence

This section explains the available boot options for Cisco NCS 1001 and their sequence.

A boot sequence is a system startup process that

- determines the order in which boot sources are selected,
- allows multiple options for local and remote boot methods, and
- ensures the system loads an available image for operation.

Cisco NCS 1001 supports three primary boot sources:

1. SSD (hard disk)
2. USB drive
3. iPXE

If no bootable image is available from any boot option, reboot the system and verify the available image source.

Boot NCS 1001

Use this procedure configure the Cisco NCS 1001 to start from the console port and verify that the system reaches the root-system username prompt.

Boot the Cisco NCS 1001 after hardware installation so the system can reach XR mode and accept initial credentials.

Use the console port to connect to the NCS 1001. By default, the console port connects to XR mode. After configuration, you can establish connections through the management port if needed.

Follow these steps to boot the Cisco NCS 1001:

Procedure

1. Connect a terminal to the console port of the RP.
2. Start the terminal emulation program on your workstation.

   Set the console settings to 115200 bps, 8 data bits, 1 stop bit, and no parity.
3. Power on the NCS 1001.

   To turn on the power shelves, press the power switch up. As NCS 1001 boots up, the boot process details are displayed at the console of the terminal emulation program.
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 the NCS 1001 more time to complete the initial boot procedure; then press **Enter**.

   > **Note**
   >
   > If the boot process fails, it may be because the pre-installed image on the NCS 1001 is corrupt. In this case, the NCS 1001 can be booted using an external bootable USB drive.

Cisco NCS 1001 reaches the root-system username prompt on the console terminal.

Boot NCS 1001 from a USB drive

Use this procedure set to prepare a bootable USB drive and recover or reimage Cisco NCS 1001 when the internal image cannot boot successfully.

Recover Cisco NCS 1001 from boot failure or reimage the device during upgrades by using a bootable USB drive.

The bootable USB drive is used to re-image the NCS 1001 for the purpose of system upgrade or to boot the NCS 1001 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.

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

Before you begin

- Use a USB drive with at least 4 GB of storage.
- Download the Cisco NCS 1001 software image from the Cisco Software Download page.
- Copy the compressed boot file from the software download page to your local machine. The compressed boot file uses the *ncs1001-usb-boot-<release_number>.zip* filename format.

Follow these steps to boot your Cisco NCS 1001 using a USB drive:

Procedure

1. Prepare the USB drive. For details, see Prepare a USB drive for Cisco NCS 1001 boot.

2. Boot Cisco NCS 1001 from the prepared USB drive. For details, see Boot Cisco NCS 1001 from a prepared USB drive.

Cisco NCS 1001 boots from the USB drive, reimages successfully, and restarts from the new installation.

Prepare a USB drive for Cisco NCS 1001 boot

Use this procedure to format the USB drive, copy the compressed boot file, and extract the boot contents before inserting the drive into Cisco NCS 1001.

Create a bootable USB drive that Cisco NCS 1001 can use for recovery or reimaging.

Follow these steps to prepare the USB drive for Cisco NCS 1001 boot:

Procedure

1. Connect the USB drive to your local machine and format it with the FAT32 file system.
2. Copy the compressed boot file to the USB drive.
3. Verify that the copy operation is successful.

   Compare the file size at the source and destination. Also verify the MD5 checksum value.
4. Extract the contents of the compressed boot file directly into the root folder of the USB drive.

> **Note**
>
> The content of the zipped file ("EFI" and "boot" directories) must be extracted 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.

The USB drive contains the extracted boot directories at the root and is ready for the device boot procedure.

Boot Cisco NCS 1001 from a prepared USB drive

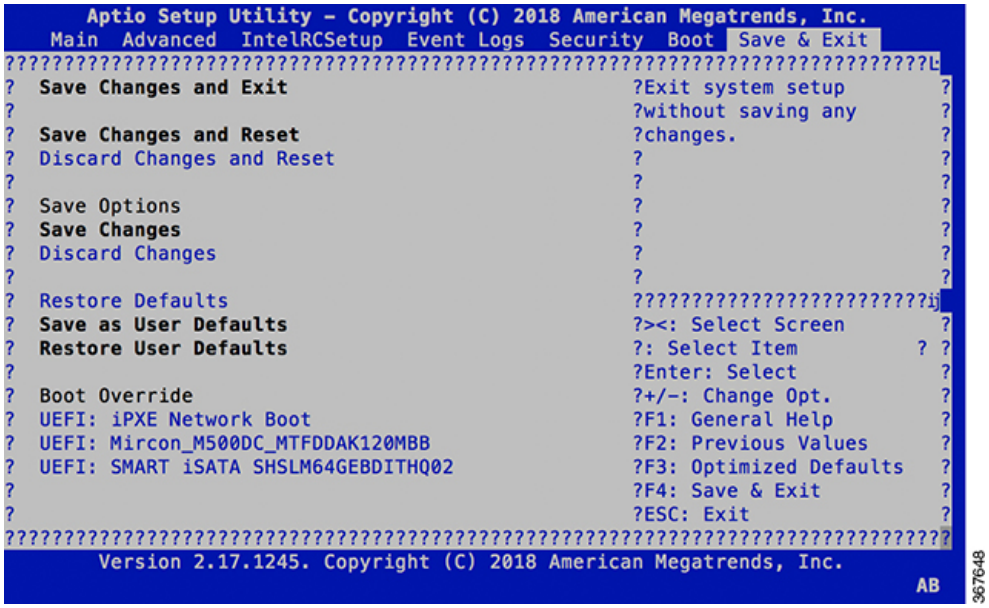
Use this procedure to boot the Cisco NCS 1001 system from a prepared USB drive to reinstall the image.

Start Cisco NCS 1001 from a prepared USB drive after the boot files are extracted to the drive root.

Follow these steps to boot Cisco NCS 1001 from the prepared USB drive:

Procedure

1. Insert the USB drive in one of the four Cisco NCS 1001 USB ports after removing any other USB drive.
2. Reboot Cisco NCS 1001 using a power cycle or the console.
3. Press **Esc** to enter BIOS.
4. Select the **Save & Exit** tab in BIOS.



5. Choose **UEFI based USB device**.

   The system detects USB and boots the image from USB.

```
Admin Console:
GNU GRUB version 2.00
Press F2 to goto grub Menu..
Booting from USB..
Loading Kernel..

Validating End Entity Certificate...

Validating SubCA Certificate...

Validating Root Certificate...
Loading initrd..

Validating End Entity Certificate...

Validating SubCA Certificate...

Validating Root Certificate...
CiscoSec: Image signature verification completed.
XR Console:
CiscoSec: Image signature verified.
[    9.957281] i8042: No controller found
Starting udev
udevd[972]: failed to execute '/etc/udev/scripts/network.sh'
'/etc/udev/scripts/network.sh': No such file or directory
Populating dev cache
Running postinst /etc/rpm-postinsts/100-dnsmasq...
update-rc.d: /etc/init.d/run-postinsts exists during rc.d purge (continuing)

 Removing any system startup links for run-postinsts ...
  /etc/rcS.d/S99run-postinsts
Configuring network interfaces... done.
```

   The system detects the USB drive and boots the image from USB. The console displays image validation and network interface configuration progress.

6. Remove the USB drive. The NCS 1001 reboots automatically.

```
Setting maximal mount count to -1
Setting interval between checks to 0 seconds
Fri Dec 11 20:35:56 UTC 2015: Install EFI on /dev/mb_disk4
Fri Dec 11 20:35:57 UTC 2015: Install finished on mb_disk
Rebooting system after installation ...
[  116.973666] reboot: Restarting system

Version 2.17.1245. Copyright (C) 2015 American Megatrends, Inc.

BIOS Date: 11/29/2015 12:02:45 Ver: 0ACBZ1110

Press <DEL> or <ESC> to enter setup.

CiscoSec: Image signature verified.


GNU GRUB version 2.00
Press F2 to goto grub Menu..
Booting from Disk..
Loading Kernel..

Validating End Entity Certificate...

Validating SubCA Certificate...

Validating Root Certificate...
Loading initrd..

Validating End Entity Certificate...

Validating SubCA Certificate...

Validating Root Certificate...
CiscoSec: Image signature verification completed.
Initrd, addr=0xff69a000, size=0x955cb0
[    1.745686] i8042: No controller found
```

Cisco NCS 1001 completes the USB-based reimage process and boots from the installed image.

iPXE network boot

This section explains how iPXE network boot enables routers to recover from boot failures by downloading and installing a system image from a network server.

An iPXE network boot is a preboot execution environment included in the network card of management interfaces that

- operates at the Unified Extensible Firmware Interface (UEFI) level of the chassis,
- reimages and boots the system after a boot failure or when no valid bootable partition is available, and
- downloads an ISO image, installs it, and bootstraps the chassis inside the new installation.

iPXE image download requirements

> **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 boot loader 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. iPXE must be defined in the DHCP server configuration file.

> **Note**
>
> For IPv6 configuration, see Configure a DHCP server for iPXE boot on page 11.

Configure a DHCP server for iPXE boot

This procedure explains how to configure a Dynamic Host Configuration Protocol (DHCP) server with chassis identifiers and image locations required by iPXE for IPv4 or IPv6 network boot.

Set up a DHCP server to provide network settings and an ISO image location to a chassis for iPXE booting.

The DHCP configuration can support IPv4, IPv6, or both protocols, and can select a boot image based on the chassis MAC address or serial number.

Before you begin

If using DHCPv6, configure the Router Advertisement Daemon (radvd) so router advertisement (RA) messages instruct nodes to obtain their IPv6 addresses through DHCP.

Install radvd by entering this command:

```
yum install radvd
```

Configure the interface in the radvd configuration. This example advertises a documentation-safe IPv6 prefix:

```
interface eth3
{
        AdvSendAdvert on;
        MinRtrAdvInterval 60;
        MaxRtrAdvInterval 180;
        AdvManagedFlag on;
        AdvOtherConfigFlag on;
        prefix 2001:DB8:C622:1::/64
        {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr off;
        };
};
```

Follow these steps to configure the DHCP server for iPXE boot:

Procedure

1. Create the required DHCP configuration files in the `/etc/` directory.

   Create `dhcpd.conf` for IPv4, IPv6, or both protocols. Create `dhcpv6.conf` for IPv6.

   Store the network information in these files, including the script path, ISO image location, provisioning configuration location, chassis serial number, and chassis MAC address.

2. For IPv4 identification by MAC address, add a host declaration to `dhcpd.conf`.

   Example

   ```
   host ncs1001
   {
   hardware ethernet ab:cd:ef:01:23:45;
   fixed-address <ip-address>;
   filename "http://<http-server-address>/<path-to-image>/ncs1001-mini-x.iso";
   }
   ```

   The server can match the chassis MAC address to a fixed address and boot-image location.

3. For IPv4 identification by serial number, add a host declaration that uses the DHCP client identifier.

   The chassis derives its serial number from the BIOS and uses the value as an identifier.

   Example

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

   This example selects an ISO image for an iPXE client and a provisioning configuration for another client:

   ```
   host 10.89.205.202 {
          hardware ethernet 40:55:39:56:0c:e8;
          if exists user-class and option user-class = "iPXE" {
             filename "http://10.89.205.127/box1/ncs1001-mini-x-7.1.1.iso";
          } else {
             filename "http://10.89.205.127/box1/StartupConfig.cfg";
          }
          fixed-address 10.89.205.202;
   }
   ```

4. For IPv6 identification by serial number, convert the serial number to a DHCP unique identifier (DUID) and add a DHCPv6 host declaration.

   Obtain the chassis serial number from the **`show inventory`** command. Convert the serial number to the IPv6 DUID format before you add it to the host declaration.

   Example

   ```
   host ncs1001 {
     host-identifier option dhcp6.client-id "<ipv6-duid>";
     filename
   "<http-or-tftp>://<ipv6-dhcp-server-address>/<hardware-platform>-mini-x.iso";
   }
   ```

This example uses a documentation-safe IPv6 address:

```
host mys-p1b-212 {

   # PID: NCS1001-K9, VID: V01, SN: CAT2018B02M

   host-identifier option dhcp6.client-id
00:02:00:00:00:09:43:41:54:32:30:31:38:42:30:32:4d:00;

   if exists dhcp6.user-class and substring(option dhcp6.user-class, 2, 4) =
 "iPXE" {
     option dhcp6.bootfile-url
"http://[2001:DB8:4491:2005::227:11]/rosco/mys-p1b-212/ncs1001-mini-mod.iso";

   }
   else {
     option dhcp6.bootfile-url
"http://[2001:DB8:4491:2005::227:11]/rosco/mys-p1b-212/ztp.ipv6.sh";
   }
}
```

5. Use a conversion script if you need to generate an enterprise-number-based DUID from the chassis serial number.

   Example

```
#! /bin/sh

# Comply with the IPv6 enterprise-number-based DUID format.
## 00:02 = Enterprise-number-based DUID
## 00:00:00:09 = Cisco enterprise number
## 00 = Terminator

SERIALNUMBER=$1
en_duid=0002
cisco_en=00000009

sn=`xxd -l11 -pu <<< ${SERIALNUMBER}`
null=00
DHCPv6_DUID=`sed 's/\(..\)/\1:/g;s/:$//' <<< ${en_duid}${cisco_en}${sn}${null}`

echo $DHCPv6_DUID
```

6. After the DHCP server is running, test the applicable host declaration with the chassis.

   A successful match provides the configured network address and boot-file location to the chassis.

The DHCP server provides an IPv4 or IPv6 address and the appropriate iPXE boot file to the chassis for network boot.

Boot the router through iPXE

Use this procedure to boot your router using iPXE network boot to reimage the chassis when it fails to start normally.

Start the iPXE network boot process to download an ISO image and restore the router's operating system.

Before you begin

- DHCP server is set and is running.
- You have logged in to the System Admin console using the **admin** command.

Follow these steps to boot your router through iPXE:

Procedure

1. Start the iPXE boot process.

   Example

   ```
   hw-module location all bootmedia network reload
   ```

   The router displays a confirmation prompt:

   ```
   sysadmin-vm:0_RP0# hw-module location all bootmedia network reload
   Wed Dec 23 15:29:57.376 UTC
   Reload hardware module ? [no,yes]
   ```

   This example shows the output of the command:

   ```
   iPXE 1.0.0+ (3e573) -- Open Source Network Boot Firmware -- http://ipxe.org
   Features: DNS HTTP TFTP VLAN EFI ISO9660 NBI Menu
   Trying net0...
   net0: c4:72:95:a6:14:e1 using dh8900cc on PCI01:00.1 (open)
   [Link:up, TX:0 TXE:0 RX:0 RXE:0]
   Configuring (net0 c4:72:95:a6:14:e1)................... Ok << Talking to
   DHCP/PXE server to obtain network information
   net0: 10.37.1.101/255.255.0.0 gw 10.37.1.0
   net0: fe80::c672:95ff:fea6:14e1/64
   net0: 2001:DB8:5000:1:C672:95FF:FEA6:14E1/64 gw fe80::20c:29ff:fefb:b9fe
   net1: fe80::c672:95ff:fea6:14e3/64 (inaccessible)
   Next server: 10.37.1.235
   Filename: http://10.37.1.235/ncs1001/ncs1001-mini-x.iso
   http://10.37.1.235/ncs1001/ncs1001-mini-x.iso ncs1000/ncs1001-mini-x.iso...
   58% << Downloading file as indicated by DHCP/PXE server to boot install image
   ```

2. Enter **yes** at the confirmation prompt.

   The chassis reloads and starts the iPXE network boot process.

The router downloads and installs the ISO image specified by the DHCP/PXE server, re-imaging the chassis and restoring normal operation.

Zero Touch Provisioning boot for Cisco NCS 1001

This section explains how Zero Touch Provisioning automates day-zero device setup on Cisco NCS 1001 through DHCP, configuration files, scripts, and CLI-triggered workflows.

A Zero Touch Provisioning method is an automated network provisioning workflow that

- applies day-zero configurations to Cisco NCS 1001 devices at scale,
- downloads and installs Cisco IOS XR images, application packages, or third-party applications automatically, and
- deploys containers and enables flexible software upgrades or downgrades across large numbers of network devices without manual intervention.

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

Benefits of zero touch provisioning

ZTP facilitates effortless management of large-scale service provider infrastructures. Its principal benefits include:

- Remote provisioning of routers anywhere in the network, eliminating the need for onsite experts and reducing IT costs.
- Automated provisioning using ZTP that removes delays, increases accuracy, improves cost-effectiveness, and provides better customer experience. By automating repeated tasks, ZTP enables network administrators to focus on higher-priority activities.

  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 by hand, you can reset a system to well-known working status.

Prerequisites

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

> **Note**
>
> For details about IPv6 support for ZTP, see IPv6 support for Zero Touch Provisioning.

ZTP initiation methods

ZTP can be initiated by one of these methods:

- **Automated fresh boot:**

  Fresh boot: On device boot, ZTP initiates automatically if there is no prior configuration. The router receives configuration file details from the DHCP server. Use this method for devices lacking preloaded configuration. The configuration file or bootscript is downloaded as specified by the DHCP server.

  You must define the configuration file or the bootscript that is downloaded from the DHCP server:

- **Configuration file:** The first line of every configuration file must contain **!! IOS XR configuration"** , to be processed correctly. If bringing up multiple nodes, define a configuration file for each.
- **Manual invocation using CLI:** Use this method when you want to forcefully initiate ZTP on a fully configured device, using CLI.

How fresh boot with DHCP works for Zero Touch Provisioning

This section explains how Cisco NCS 1001 uses DHCP during a fresh boot to locate and run the configuration file or script needed for Zero Touch Provisioning.

When you boot the device, the ZTP process initiates automatically if the device does not have a prior configuration.

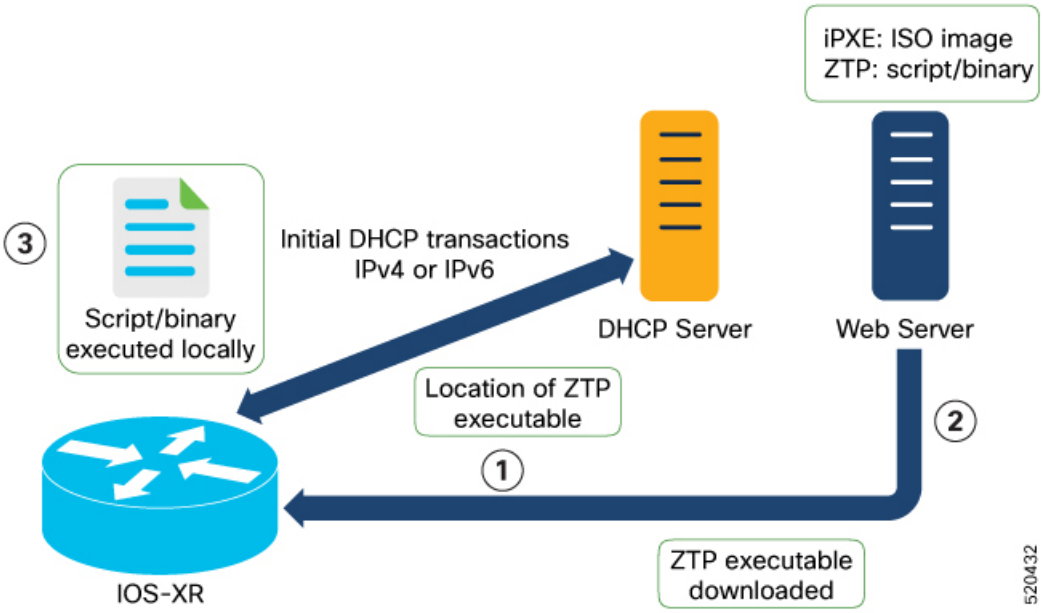
Summary

During the process, the router receives the details of the configuration file from the DHCP server.

This image depicts the high-level work flow of the ZTP process:

Workflow

Figure 1: Fresh boot workflow for Zero Touch Provisioning



These stages describe fresh boot with DHCP for ZTP:

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:
   - ZTP sends IPv4 DHCP request first on all the management port. In case there is a failure, then ZTP sends IPv6 DHCP request on all the management port.
   - ZTP sends IPv4 DHCP request first on all the data port. In case there is a failure, then ZTP sends IPv6 DHCP request on all the data port.

     The default sequential flow is defined in configuration file and you can modify the sequence using the configuration file.

2. DHCP server identifies the device and responds with DHCP response using one of the following options:

   DHCP server should be configured to respond with the DHCP options.

   DHCP server should be configured to respond with the DHCP options.
   - DHCPv4 using BOOTP filename to supply script/config location.
   - DHCPv4 using Option 67 (bootfile-name) to supply script/config location.
   - DHCPv6 using Option 59 (OPT_BOOTFILE_URL) to supply script/config location
   - For example, for IPv4:

```
host mys-p1b-209-Ztp {
option dhcp-client-identifier
```

```
00:02:00:00:00:09:43:41:54:32:31:30:37:42:30:4d:44:00;
  filename "http://192.0.2.10/ncs1000/mys-p1b-209-Ztp/-mini-x.iso";

}
```

For example, for IPv6:

```
host mys-p1b-209-Ztp {
                # PID: NCS1001-K9        , VID: V01, SN: CAT2107B0MD
                # needed for ztp
                host-identifier option dhcp6.client-id
00:02:00:00:00:09:43:41:54:32:31:30:37:42:30:4d:44:00;


                if exists dhcp6.user-class and substring(option
dhcp6.user-class, 2, 4) = "iPXE" {
                        option dhcp6.bootfile-url
"http://[2001:DB8::4491:2005::227:11]/ncs1001/mys-p1b-209/ncs1001-mini-mod.iso";

                }
                else {
                        option dhcp6.bootfile-url
"http://[2001:DB8::4491:2005::227:11]/ncs1001/mys-p1b-209/ztp.ipv6.sh";
                }
        }
```

3. The network device downloads the file from the web server using the URI location that is 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.

Result

The network device is successfully provisioned and starts operating with the automatically applied configuration or script, completing Zero Touch Provisioning without manual intervention.

Zero Touch Provisioning configuration files

This section provides IPv4 and IPv6 configuration file examples and syntax requirements for starting Zero Touch Provisioning on Cisco NCS 1001.

Use a configuration file or script file to initiate ZTP based on your deployment requirement. A configuration file must start with **!! IOS XR**.

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**.

Sample configuration file for IPv4

This is the sample configuration file for IPv4. You can automate all the configurations.

```
!! IOS XR Configuration version = 6.3.2
!
telnet vrf default ipv4 server max-servers 20
!
vty-pool default 0 20 line-template default
!
interface MgmtEth0/RP0/CPU0/0
 ipv4 address dhcp
 no shutdown
!
router static
 address-family ipv4 unicast
  0.0.0.0/0 10.77.132.1
!
end
```

Sample configuration file for IPv6

This is the sample configuration file for IPv6.

```
!! IOS XR Configuration version = 7.10.1
!
telnet vrf default ipv6 server max-servers 20
!
vty-pool default 0 20 line-template default
!
interface MgmtEth0/RP0/CPU0/0
 ipv6 address dhcp6
 no shutdown
!
router static
 address-family ipv6 unicast
  2001:db8:1000::/36 2001:db8:2000:2::1
!
end
```

Invoke Zero Touch Provisioning manually through the CLI

Use this procedure to start, test, or reset Zero Touch Provisioning (ZTP) from the CLI to provision your router and review ZTP activity without rebooting.

Use manual ZTP commands to provision your router in stages and test ZTP configuration without rebooting the device.

Manual ZTP can be executed using CLI commands, allowing you to provision the router in stages–ideal for testing ZTP configuration without restarting. You can initiate ZTP on any interface (data ports or management port) without configuring or activating the interface first; even if the interface is down, the ZTP script brings it up and invokes DHCP client as needed.

Use the `ztp initiate`, `ztp terminate`, and `ztp clean` commands to control ZTP sessions over multiple interfaces.

- ztp initiate–Invokes a new ZTP DHCP session. Logs can be found in `/disk0:/ztp/ztp.log`.
- ztp terminate–Terminates any ZTP session in progress.

- ztp clean–Removes only the ZTP state files.

The log file `ztp.log` is saved in `/var/log/ztp.log` folder, and a copy of log file is available at `/disk0:/ztp/ztp.log` location using a soft link. However, executing **ztp clean** clears files saved on disk and not on `/var/logztp.log` folder where current ZTP logs are saved. In order to have a log from current ZTP run, you must manually clear the ZTP log file from `/var/log/ztp.log` folder.

Follow these steps to manually invoke Zero Touch Provisioning (ZTP) through the CLI:

Procedure

1. (Optional) Run the **ztp clean** command to removes all the ZTP logs and saved settings.

   Example

   ```
   RP/0/RP0/CPU0:ios#ztp clean
   Fri Apr 29 06:49:29.760 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.
   ```

2. Run the **ztp initiate** command to start a new ZTP DHCP session.

   Example

   ```
   RP/0/RP0/CPU0:ios#ztp initiate
   Fri Jun 17 11:44:08.791 UTC
   Initiating 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#
   ```

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

3. (Optional) Run the **ztp terminate** command to stop the active ZTP session.

   Example

   ```
   RP/0/RP0/CPU0:ios#ztp terminate
   Fri Apr 29 06:38:59.238 UTC
   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
   No ZTP process running
   ```

   This command terminates any ZTP session in progress.

> **Note**
>
> For IPv6 configuration, see Configure a DHCP server for iPXE boot on page 11.

ZTP is started, stopped, or reset based on the commands you issue, allowing for flexible provisioning and testing of your router without rebooting.

Invoke Zero Touch Provisioning through reload

Use this procedure to remove the running configuration, clean ZTP state files, and reload Cisco NCS 1001 so that ZTP starts automatically.

Trigger ZTP by reloading the system after replacing the running configuration and cleaning ZTP operation files.

The ZTP process can start automatically after a reload when the device has no configured username or retained running configuration.

Procedure

1. Run **configure** command to enter configuration mode.

   Example

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

2. Run **commit replace** command to removes the entire running configuration.

   Example

   ```
   Fri Apr 29 06:48:46.236 UTC
   RP/0/RP0/CPU0:ios(config)#commit replace
   Fri Apr 29 06:48:53.199 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)#
   RP/0/RP0/CPU0:ios(config)#end
   ```

3. Run the **ztp clean** command to removes all the ZTP logs and saved settings.

   Example

   ```
   RP/0/RP0/CPU0:ios#ztp clean
   Fri Apr 29 06:49:29.760 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.
```

4. Run the **reload** command to reload the node or hardware module.

   Example

```
RP/0/RP0/CPU0:ios#admin reload location 0/RP0 all
Fri Apr 29 06:50:29.760 UTC
Reload node ? [no,yes] yes
```

```
RP/0/RP0/CPU0:ios#admin hw-module location 0/RP0 reload
Fri Apr 29 06:52:29.760 UTC
Reload hardware module ? [no,yes] yes
```

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

   Example

```
RP/0/RP0/CPU0:Apr 29 06:55:33.242 UTC: pyztp2[377]:
%INFRA-ZTP-4-CONFIG_INITIATED : ZTP has initiated config load and commit
operations
RP/0/RP0/CPU0:Apr 29 06:55:39.263 UTC: ifmgr[381]: %PKT_INFRA-LINK-3-UPDOWN
: Interface GigabitEthernet0/0/0/0, changed state to Down
RP/0/RP0/CPU0:Apr 29 06:55:39.287 UTC: osa_driver[183]:
%PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :PROV-INPROGRESS :DECLARE
:GigabitEthernet0/0/0/0:
RP/0/RP0/CPU0:Apr 29 06:55:39.287 UTC: osa_driver[183]:
%PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :PROV-INPROGRESS :DECLARE
:Osc0/0/0/0:
RP/0/RP0/CPU0:Apr 29 06:55:39.287 UTC: ifmgr[381]: %PKT_INFRA-LINK-3-UPDOWN
: Interface GigabitEthernet0/0/0/0, changed state to Up
RP/0/RP0/CPU0:Apr 29 06:55:39.716 UTC: osa_driver[183]:
%PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :PROV-INPROGRESS :CLEAR :Osc0/0/0/0:

RP/0/RP0/CPU0:Apr 29 06:55:39.728 UTC: osa_driver[183]:
%PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :PROV-INPROGRESS :CLEAR
:GigabitEthernet0/0/0/0:
RP/0/RP0/CPU0:Apr 29 06:55:47.904 UTC: osa_driver[183]:
%PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :PROV-INPROGRESS :DECLARE
:Ots0/0/0/1:

User Access Verification

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

Cisco NCS 1001 reloads and initiates Zero Touch Provisioning automatically.

Zero Touch Provisioning logs

This section provides details about Zero Touch Provisioning log locations and representative log entries used to trace DHCP transactions and state changes.

ZTP logs operations on the flash file system in the `/disk0:/ztp/` directory. The logs include DHCP server transactions and ZTP state transitions.

Management interface log example

This example shows selected log entries from a simple configuration script downloaded through a management interface.

```
2023-05-12 14:43:18,406   5845 [Env          ] INF: MgmtDhcp4Fetcher fetcher
created.
2023-05-12 14:43:18,482   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:idle. Processing work: [privileged] start an engine. done = False
2023-05-12 14:43:18,484   5845 [Engine       ] INF: MgmtDhcp4Fetcher, current
 state:active: state changed to active
2023-05-12 14:43:18,508   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Monitor fetcher work for ZAdmin. done = False
2023-05-12 14:43:18,585   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: epoch work. done = False
2023-05-12 14:43:18,586   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: [privileged] getting engine status. done =
False
2023-05-12 14:43:18,587   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: Bringing up interfaces before next retry. done
 = False
2023-05-12 14:43:18,589   5845 [Port         ] DEB: <Port count=1>: bringing
interface(s) up "MgmtEth0/RP0/CPU0/0"
2023-05-12 14:43:18,624   5845 [Port         ] DEB: Saving 1 interfaces to
/disk0:/ztp/xr_config/interface_list
2023-05-12 14:43:18,626   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: Filtering up interfaces for MgmtDhcp4Fetcher.
 done = False
2023-05-12 14:43:18,627   5845 [Management   ] DEB: Determining operstate of
interface: brXRmgmt1
2023-05-12 14:43:18,666   5845 [Port         ] DEB: Filtered up interfaces:
[Name: MgmtEth0/RP0/CPU0/0 (Type: Management) (Up: False)]
2023-05-12 14:43:18,670   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: Setup interfaces for MgmtDhcp4Fetcher. done =
 False
2023-05-12 14:43:18,671   5845 [Env         ] INF: Env::getVlanIDs: vlan.mode:2
2023-05-12 14:43:18,672   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: Start Dhclient for MgmtDhcp4Fetcher. done =
False
2023-05-12 14:43:18,687   5845 [Port        ] DEB: <Dhclient count=1>: started
 dhclient using "ip netns exec xrnns /sbin/dhclient -4 -cf
/etc/dhcp/dhclient.conf.ztp -lf /var/lib/dhcp/dhclient.leases.ztp -sf
/etc/dhcp/dhclient-script.ztp2 brXRmgmt1"
2023-05-12 14:43:19,090   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Checking for username configuration. done =
False
2023-05-12 14:43:19,630   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Monitor fetcher work for ZAdmin. done = False
2023-05-12 14:43:19,692   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: [privileged] getting engine status. done =
False
2023-05-12 14:43:20,696   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Monitor fetcher work for ZAdmin. done = False
2023-05-12 14:43:20,797   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
```

```
 state:active. Processing work: [privileged] getting engine status. done =
False
2023-05-12 14:43:21,099   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Sending standby sync message. done = False
2023-05-12 14:43:21,212   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: [privileged] getting engine status. done = False
2023-05-12 14:43:21,700   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:active. Processing work: Monitor dhclient for MgmtDhcp4Fetcher on
interface ['brXRmgmt1']. done = False
2023-05-12 14:43:21,704   5845 [MgmtDhcp4Fetcher] DEB: Received DHCP4 response
2023-05-12 14:43:21,705   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
reason=BOUND
2023-05-12 14:43:21,706   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
interface=brXRmgmt1
2023-05-12 14:43:21,706   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_ip_address=10.58.227.177
2023-05-12 14:43:21,707   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_network_number=10.58.227.0
2023-05-12 14:43:21,708   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_subnet_mask=255.255.255.0
2023-05-12 14:43:21,708   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_broadcast_address=10.58.227.255
2023-05-12 14:43:21,709   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_routers=10.58.227.1
2023-05-12 14:43:21,710   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_dhcp_server_identifier=10.58.227.11
2023-05-12 14:43:21,711   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_domain_name=cisco.com
2023-05-12 14:43:21,711   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_domain_name_servers=198.51.100.3 198.51.100.1
2023-05-12 14:43:21,712   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_filename=http://10.58.227.11/rosco/mys-p1b-212/ztp.sh
2023-05-12 14:43:21,713   5845 [MgmtDhcp4Fetcher] INF: (dhclient env)
new_ip6_prefixlen=64
2023-05-12 14:43:21,714   5845 [Engine       ] INF: MgmtDhcp4Fetcher, current
 state:active, exit code:success
2023-05-12 14:43:21,815   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Monitor fetcher work for ZAdmin. done = False
2023-05-12 14:43:21,828   5845 [Port         ] DEB: Dhclient processes:
root      7033  0.0  0.0  31096  7312 ?        Ss   14:46   0:00 /sbin/dhclient
 -4 -cf /etc/dhcp/dhclient.conf.ztp -lf /var/lib/dhcp/dhclient.leases.ztp -sf
 /etc/dhcp/dhclient-script.ztp2 brXRmgmt1
root      7052  0.0  0.0  20316  1592 ?        S    14:46   0:00 /bin/sh -c
ps aux | grep dhclient
root      7054  0.0  0.0  16248   948 ?        Sl   14:46   0:00 grep dhclient
2023-05-12 14:43:21,830   5845 [Port         ] DEB: <Dhclient count=1>: dhclient
 4 is stopped: keepIpAddress=True
2023-05-12 14:43:21,831   5845 [Engine       ] INF: MgmtDhcp4Fetcher, current
 state:final, exit code:success: state changed to final
2023-05-12 14:43:21,832   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:final, exit code:success. Processing work: [privileged] getting engine
 status. done = False
2023-05-12 14:43:21,834   5845 [MgmtDhcp4Fetcher] DEB: dhcp: shutdown : Entry
2023-05-12 14:43:21,933   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:final, exit code:success. Processing work: [privileged] prepare engine
 shutdown. done = False
2023-05-12 14:43:22,035   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, current
 state:final, exit code:success. Processing work: [privileged] shutting down
MgmtDhcp4Fetcher engine. done = False
2023-05-12 14:43:22,036   5845 [Engine       ] INF: MgmtDhcp4Fetcher, current
 state:final, exit code:shutdown
2023-05-12 14:43:22,037   5845 [Engine       ] INF: MgmtDhcp4Fetcher, exit
```

```
code:shutdown: state changed to None
2023-05-12 14:43:22,038   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, exit
code:shutdown: breaking engine loop after shutdown
2023-05-12 14:43:22,039   5845 [Engine       ] DEB: ZAdmin, current
state:active. Processing work: Setup fetching. done = False
2023-05-12 14:43:22,039   5845 [Engine       ] DEB: MgmtDhcp4Fetcher, exit
code:shutdown: end of event loop
2023-05-12 14:43:22,052   5845 [Env          ] DEB: No authentication required
 for Mgmt Interface
2023-05-12 14:43:22,053   5845 [Env          ] DEB: No authentication required
 when initiated using CLI
2023-05-12 14:43:22,055   5845 [Xr           ] DEB: Writing to file
/tmp/ztp2-fzwmlfid/sysdb_cfg_cmd.tmp
2023-05-12 14:43:22,181   5845 [Xr           ] DEB: No inconsistency found in
 config
2023-05-12 14:43:23,460   5845 [Xr           ] DEB: Applying TPA default route
2023-05-12 14:43:23,523   5845 [Xr           ] DEB: No IPv4 Address assigned
to linux management interface
2023-05-12 14:43:23,524   5845 [Xr          ] DEB: Applying IPv4 configuration
2023-05-12 14:43:23,525   5845 [Xr           ] DEB: Validating IP Address:
10.58.227.177
2023-05-12 14:43:23,527   5845 [Xr          ] DEB: Applying IPv4 gateway route
 configuration
2023-05-12 14:43:23,527   5845 [Xr           ] DEB: Validating DHCP server
identifier IP Address: 10.58.227.11
2023-05-12 14:43:23,528   5845 [Xr           ] DEB: Validating Gateway IP
Address: 10.58.227.1
2023-05-12 14:43:23,530   5845 [Xr           ] DEB: Configuring domain name
with domain-name-server 198.51.100.3 198.51.100.1
2023-05-12 14:43:23,532   5845 [Configuration] DEB: Config file type is IOS
XR config. Replace False
2023-05-12 14:43:23,534   5845 [Configuration] DEB: Applying following config:

tpa
 vrf default
  address-family ipv4
   default-route mgmt
  address-family ipv6
   default-route mgmt

interface MgmtEth0/RP0/CPU0/0
no ipv4 address
ipv4 address 10.58.227.177 255.255.255.0
no shutdown

tpa
 vrf default
  address-family ipv4
   default-route mgmt
router static
 address-family ipv4 unicast
  0.0.0.0/0 10.58.227.1

domain name cisco.com

domain name-server 198.51.100.3

domain name-server 198.51.100.1
```

Collect tech support information for Zero Touch Provisioning

Use this procedure to collect debug data for Zero Touch Provisioning on Cisco NCS 1001 to facilitate troubleshooting with Cisco Technical Support.

Collect troubleshooting data that Cisco Technical Support can use to isolate and resolve Zero Touch Provisioning (ZTP).

When you have a problem in the ZTP process that you cannot resolve, the resource of last resort is your 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 problem isolation and resolution process, collect the necessary data before you contact your representative.

Follow these steps to collect tech support data for ZTP troubleshooting:

Procedure

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

Example

```
RP/0/RP0/CPU0:R1#show tech-support ztp
Thu Jun 15 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 15 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:R1#
```

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.

The ZTP tech support file is prepared and ready to be shared with Cisco Technical Support, enabling efficient diagnosis and resolution of your issue.

IPv6 support for Zero Touch Provisioning

This section explains how IPv6 addressing extends Zero Touch Provisioning support for Cisco NCS 1001 nodes by enabling onboarding through DHCPv6 client IDs and boot file URLs. This process enhances device management efficiency and security.

IPv6 support for Zero Touch Provisioning is a DHCP-based onboarding capability that

- supports Cisco NCS 1001 nodes that use IPv6 addressing,
- uses the DHCPv6 client ID to identify a node, and
- uses an IPv6 address-based boot file URL to locate the ZTP file.

Feature history
Table 1: Feature History

Feature Name	Release Information	Description
IPv6 Support for ZTP	Cisco IOS XR Release 7.10.1	From this release, the DHCP server supports Zero Touch Provisioning (ZTP) to bring up the NCS 1001 nodes with IPv6 addressing. The DHCP configuration file must be updated with the dhcp6.client id and IPv6 address-based bootfile URL. IPv6 addressing ensures efficient and secure management of the devices.

From Release 7.10.1, you can bring up the NCS 1001 nodes that are enabled with IPv6, using the DHCP configuration file updated with dhcp6.client id and IPv6 address based bootfile URL. See How fresh boot with DHCP works for Zero Touch Provisioning on page 15.

DHCP relay agents

This section explains how DHCP relay agents forward DHCP messages across network segments and support remote Cisco NCS 1001 provisioning over the OSC channel.

A DHCP relay agent is a network device that

- forwards DHCP packets between clients and servers that are not on the same subnet,
- enables DHCP support in networks where direct communication between clients and servers is not possible, and
- uses profiles and helper addresses to direct DHCP messages appropriately.

The NCS 1001 device, connected to a Data Communications Network (DCN), can act as a DHCP relay agent through the OSC channel established between the client NCS 1001 device and the relay agent. Zero Touch Provisioning (ZTP) uses the DHCP relay agent to automate device onboarding and configuration for remote NCS 1001 devices in environments where direct management connectivity is not feasible. This method is essential for deploying NCS 1001 devices in distributed or constrained environments. When a client resides on a network segment without a DHCP server, a relay agent on that segment ensures DHCP packets reach servers across other segments.

Table 2: Feature History

Feature Name	Release	Description
ZTP of remote NCS 1001 devices through DHCP frame relay over OSC	Cisco IOS XR Release 25.4.1	This feature enables you to configure the NCS 1001 device, connected to the DCN, to function as a DHCP Frame Relay over the OSC channel. This supports zero touch provisioning for remote NCS 1001 devices. With this capability, you can provision NCS 1000 devices remotely, removing the need for onsite personnel or manual configuration. This saves time and effort, especially in locations that are difficult to access. It is particularly beneficial in RON architecture where all links are point to point.

Key features and advantages

- DHCP relay agents extend DHCP communication across subnet boundaries over the OSC channel.
- Enables remote configuration and centralized management of NCS 1001 devices not directly attached to the management network.
- Accelerates deployment and reduces manual effort in hard-to-reach sites or Routed Optical Network (RON) environments.

Limitations for DHCP relay

- Multicast addresses are not supported. The helper-address command in DHCP relay profile submode supports only valid unicast IP addresses as the helper address.
- Configuring the helper-address command directly (not using profile) under an interface (such as a BVI interface) is not supported.
- Only interface-id and remote-id DHCP option codes are added by the relay agent while forwarding packets to a DHCP server.
- Configuring DHCP option codes is not supported in DHCP relay profile submode.

How DHCP relay works

This section describes how a DCN-connected Cisco NCS 1001 relays DHCP requests over the OSC channel to provision remote Cisco NCS 1001 devices.

DHCP relays enable remote Cisco NCS 1001 devices to be configured automatically by forwarding DHCP requests over the OSC channel, using a DCN-connected NCS 1001 as the intermediary

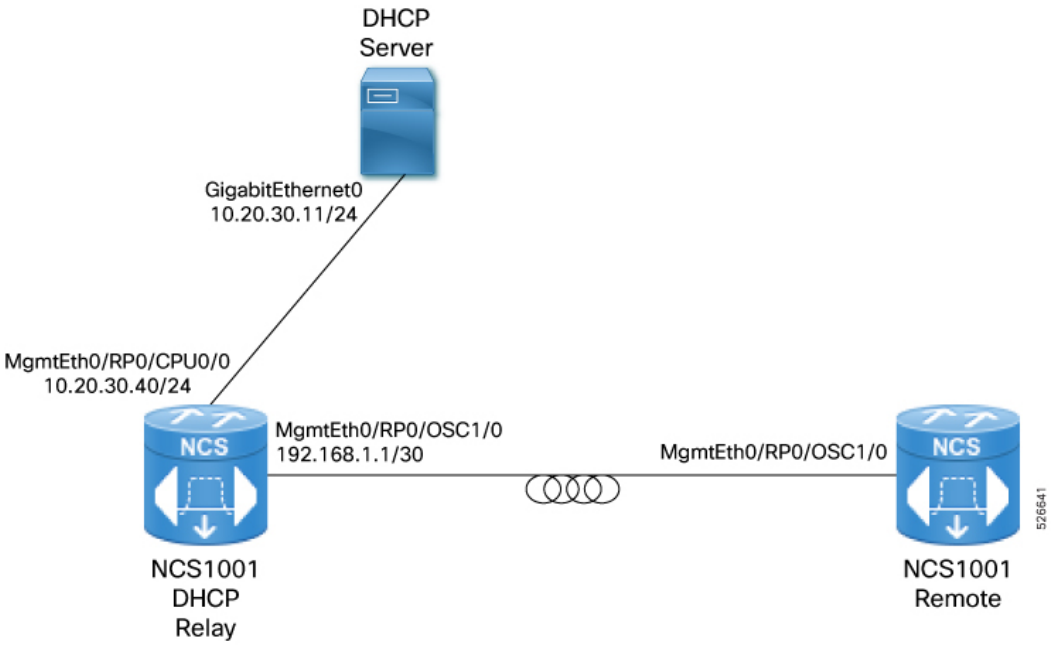
Summary

The key components involved in the process are:

- **Remote Cisco NCS 1001 device:** Starts and sends a DHCP request for ZTP.
- **Cisco NCS 1001 DHCP relay:** Connects to the DCN and relays DHCP packets between the remote device and the DHCP server over the OSC channel.
- **DHCP server:** Provides IP addressing and configuration to the remote device.

Workflow

Figure 2: DHCP relay workflow for remote Cisco NCS 1001 configuration



The process involves these stages:

1. The remote Cisco NCS 1001 starts and sends a DHCP request over the OSC channel.
2. The DCN-connected Cisco NCS 1001, configured as a DHCP relay, receives the request and forwards it to the DHCP server.
3. The DHCP server responds with the required configuration, and the relay passes the response back to the remote device.
4. The remote Cisco NCS 1001 receives its configuration and completes Zero Touch Provisioning.

Result

Remote NCS1001 devices are automatically configured and onboarded without manual intervention, streamlining deployment in distributed network environments.

Configure a DHCP relay agent

Use this procedure to configure a DCN-connected Cisco NCS 1001 as a DHCP relay agent for remote Zero Touch Provisioning (ZTP) over the OSC channel.

Configure a Cisco NCS 1001 device to relay DHCP requests from remote devices over the OSC channel, enabling remote devices to complete automated provisioning.

Before you begin

- Ensure that you have administrative access to the Cisco NCS 1001 device connected to the DCN.
- Confirm that the device supports DHCP relay functionality.
- Verify the required IP addresses for the DHCP server and relay interface.

Procedure

1. Enter configuration mode.

Example

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

2. Enable DHCP IPv4 services.

   Example

   ```
   RP/0/RP0/CPU0:ios(config)# dhcp ipv4
   ```

3. Configure a relay profile for the DHCP IPv4 component.

   Example

   ```
   RP/0/RP0/CPU0:ios(config-dhcpv4)# profile p1 relay
   ```

4. Configure the DHCP IPv4 relay agent to relay DHCP packets to the DHCP server.

   Example

   ```
   RP/0/RP0/CPU0:ios(config-dhcpv4-relay-profile)# helper-address vrf default
   10.20.30.11 giaddr 192.168.1.1
   ```

5. Configure the relay profile on the interface.

   Example

   ```
   RP/0/RP0/CPU0:ios(config-dhcpv4-relay-profile)# interface MgmtEth0/RP0/OSC1/0
    relay profile p1
   ```

6. Commit the configuration.

   Example

   ```
   RP/0/RP0/CPU0:ios(config-dhcpv4)# commit
   ```

The Cisco NCS 1001 device relays DHCP requests from remote devices over the OSC channel, allowing those devices to automatically complete Zero Touch Provisioning.

Golden ISO

This section explains Golden ISO support for Cisco NCS 1001, including image contents, prerequisites, limitations, supported filename formats, and deployment value.

A Golden ISO is a customized bootable ISO image that

- packages the mini ISO, required Software Maintenance Updates (SMUs), and Cisco IOS XR configuration,
- installs the required image and packages in a single boot cycle, and
- reduces the manual steps needed to install, activate, and configure Cisco NCS 1001.

Before Golden ISO, installing a new image required booting the system with mini ISO, installing and activating SMUs or optional packages, and applying Cisco IOS XR configuration. Golden ISO combines these actions into one image-based workflow.

Benefits of Golden ISO

Golden ISO provides these benefits:

- Saves installation effort and time.
- Makes the system ready in a single command and a single boot.

Prerequisites for Golden ISO

Meet these prerequisites before you build a Golden ISO:

- The **mount**, **rm**, **cp**, **umount**, **zcat**, **chroot**, and **mkisofs** tools must be available, and the user must have permission to run them.
- Python 2.7 must be available.
- You can run the `gisobuild.py` script on a Linux machine or on Cisco NCS 1001. The destination must use an EXT file system. FAT32 and NTFS file systems are not supported.
- If `gisobuild.py` runs on Cisco NCS 1001, set `PYTHONPATH=/pkg/bin`.
- The system must have 3 GB to 4 GB of free disk space.
- The mini ISO and `-r RPMREPO` argument are mandatory.
- The system kernel version must be greater than 3.16 or greater than the kernel version of the Cisco ISO.
- The user must have permission for the security RPM (`k9sec-rpm`) in the RPM repository. Otherwise, the security RPM is ignored during Golden ISO creation.

Limitations for Golden ISO

Golden ISO has this limitation:

- Install operation over IPv6 is not supported.

Golden ISO filename formats

Use these filename formats when you create a Golden ISO file:

- `platform-name-golden-x.iso-version.label`: Does not contain the security package RPM, such as `k9sec`. **Example:** `ncs1001-golden-x-7.1.1.14I-V1.iso`.
- `platform-name-goldenk9-x.iso-version.label`: Contains the security package RPM, such as `k9sec`. **Example:** `ncs1001-goldenk9-x-7.1.1.14I-V1.iso`.

Boot Cisco NCS 1001 with Golden ISO

Use this procedure to create a Golden ISO image, prepare the USB package, and verify Cisco NCS 1001 after it boots from the new image.

Boot the Cisco NCS 1001 using a Golden ISO image, ensuring the system loads the required image, packages, and configuration for operational integrity.

Follow these steps to boot the Cisco NCS 1001 with a Golden ISO image:

Procedure

1. Create the Golden ISO image by using the `gisobuild.py` script available at `/pkg/bin`.
2. Create the USB ZIP package.

   Example

   ```
   ./create_usb_zip ncs1001 ncs1001xxxxx-giso.iso
   ```

3. Extract the ZIP package contents and copy them to the USB drive used for boot.
4. Insert the USB drive in the USB port under test.
5. Reboot the system and install the new image.
6. After the system reboots, verify the configuration, release, and active packages.

   Example

   ```
   RP/0/RP0/CPU0:ios# show running-config
   RP/0/RP0/CPU0:ios# show version
   RP/0/RP0/CPU0:ios# show install active
   ```

The Cisco NCS 1001 boots successfully using the Golden ISO image, with the correct configuration, release, and active packages displayed.

Verify boot operation

Use this procedure to verify that the running Cisco IOS XR software version matches the boot image version after the Cisco NCS 1001 starts.

Ensure Cisco NCS 1001 is running the expected software version following the boot process.

Follow these steps to verify the boot operation of Cisco NCS 1001:

Procedure

1. After the boot operation, reload Cisco NCS 1001.
2. Run the **show version** command to display the running software version.

   Example

   ```
   Tue Jan 14 12:31:05.745 CET
   Cisco IOS XR Software, Version 7.1.1
   Copyright (c) 2013-2019 by Cisco Systems, Inc.
   Build Information:
   Built By : nkhai
   Built On : Tue Jan 7 16:31:55 PST 2020
   Built Host : iox-lnx-071
   Workspace : /auto/iox-lnx-071-san1/prod/7.1.1/ncs1001/ws
   Version : 7.1.1
   Location : /opt/cisco/XR/packages/
   ```

```
Label : 7.1.1
cisco NCS-1001 () processor
```

Compare the displayed version with the boot image version. The versions must match.

The displayed Cisco IOS XR software version matches the boot image version.

Console sessions on Cisco NCS 1001

This section provides details about Cisco NCS 1001 console session types, console swap commands, keyboard shortcuts, and first-login notes for administrators.

Cisco NCS 1001 has XR, System Admin, and host console sessions. After you enter the root username and password, you log in to the XR console. You can switch from XR to System Admin and host consoles by using CLI commands or the keyboard shortcut.

Table 3: Feature history

Feature Name	Release	Description
Console Swap for NCS 1001	Cisco IOS XR Release 7.8.1	Console swap feature provides a quicker and simpler way to toggle between the following console sessions using a keyboard shortcut: • from XR to Admin console • from Admin to Host console • from Host to XR console

NCS 1001 has three types of consoles to interact with it. After entering the root username and password, you are logged in to the XR console. From the XR console, you can switch to the System Admin console and then to the host console using CLI commands.

This table lists the console session available on NCS 1001.

Table 4: Console session types

Console type	Functions
XR Console	Run the regular CLI commands to configure and manage the node.
System Admin Console	Perform all system administration and hardware management setups.
Host Console	Run the Linux commands such as pwd, scp, cp, traceroute, ls -la and so on.

From Release 7.8.1, you can enter **Ctrl+O** to swap between the console session, host session, and XR session. This keyboard shortcut enables you to toggle between consoles without using several CLI commands. However, you can still use the CLI commands to toggle between the console sessions.

This table lists the CLI to move between consoles.

Table 5: Source and target console swap for NCS 1001

Changing console session		Command prompt	Keyboard shortcut	Notes
From	**To**			

XR console	System Admin console	**XR console**
		`RP/0/RP0/CPU0:ios# admin`
		System Admin console
		`sysadmin-vm:0_RP0#`
			CTRL+O
			Example
			`RP/0/RP0/CPU0# Disconnecting from 'default-sdr--1' console. Continue(Y/N)? (Enter -> Y and then <enter> to confirm) Connecting to 'default-sdr--1' console Connecting to 'sysadmin' console ▒▒▒▒▒▒▒▒▒▒ System Admin Username: <enter username> Password: <enter password> sysadmin-vm:0_RP0#`
				When entering the System Admin console for the first time, you must provide the username and password to continue with the System Admin console.
				To return to the XR console from the System Admin console using CLI, enter **exit**.

System admin console	Host console	**System Admin console**
		`sysadmin-vm:0_RP0#run`
		`sysadmin-vm:0_RP0#ssh 10.0.2.16`
		Host console
		`[host:~]$`
			CTRL+O
			Example
			`sysadmin-vm:0_RP0# Disconnecting from 'sysadmin' console. Continue(Y/N)? (Enter -> Y and then <enter> to confirm) Connecting to 'host' console host login:<username> Password:<password> [host:/]$`
				When entering the host console for the first time, you must provide the username and password to continue with the host console.
				To access the host console, use the internal host IP address 10.0.2.16 using SSH.

Changing console session		Command prompt	Keyboard shortcut	Notes
From	To			

Host console	XR console	**Host console**
		`[host:~]$exit`
		`logout`
		`Connection to 10.0.2.16 closed.`
		`[sysadmin-vm:0_RP0:~]$ [sysadmin-vm:0_RP0:~]$exit`
		`exit`
		`sysadmin-vm:0_RP0# sysadmin-vm:0_RP0#exit`
		`Thu Oct 13 10:08:17.162 UTC+00:00`
		`RP/0/RP0/CPU0#`
		XR console
		RP/0/RP0/CPU0:ios#
			CTRL+O
			Example
			`[host:/]$`
			`Disconnecting from host console. Continue(Y/N)? (Enter -> Y and then <enter> to confirm)`
			`Connecting to 'default-sdr--1' console`
			`RP/0/RP0/CPU0#`
				When toggling from host console to XR console using CLI, you must enter **exit** to switch to System Admin console. Then in Admin console, enter **exit** twice to switch to XR console.

Configure the management interface

Use this procedure to assign the management Ethernet interface address, bring the interface up, and configure the default static route.

Configure the management interface for system management and remote communication.

Before you begin

- Obtain IP addresses and a subnet mask for the management port from your network administrator or system planner.
- Ensure the management port is connected to the management network.

To communicate with devices on other networks, such as remote management stations or TFTP servers, configure a default (static) route for Cisco NCS 1001.

The management plane supports an MTU range of 64 to 1514 bytes.

Follow these steps to configure the management interface:

Procedure

1. Use the **configure interface mgmtEth** *rack/slot/instance/port* to enter the management interface configuration mode.

   Example

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

2. Use the **ipv4 address** *ipv4-address subnet-mask* command to assign an IP address and subnet mask to the interface.

Example

For IPv4, use this command.

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

For IPv6, use this command.

ipv6 address *ipv6-address subnet-mask*

Example

```
RP/0/RP0/CPU0:ios(config-if)# ipv6 address 2001:420:4491:2000::229:118/64
```

3. Use the **no shutdown** command to bring up the interface.

Example

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

4. Use the **exit** command to exit management interface configuration mode.

Example

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

5. Use the **router static address-family ipv4 unicast** *0.0.0.0/0default-gateway* to configure a default static route.

Example

Specifies the IP address of the default-gateway to configure a static route; this is to be used for communications with devices on other networks.

For IPv4, use this command

Example

```
RP/0/RP0/CPU0:ios(config)# router static address-family ipv4 unicast 0.0.0.0/0
 198.51.100.2
```

For IPv6, use this command.

Example

```
RP/0/RP0/CPU0:ios(config)# router static address-family ipv6 unicast ::/0
2001:420:4491:2000::228:2
```

The default gateway IP address enables communication with devices on other networks.

Example

```
RP/0/RP0/CPU0:ios(config)# router static address-family ipv6 unicast ::/0
2001:DB8:4491:2000::228:2
```

6. Use the **commit** or **end** command.

- **Commit** – saves the configuration changes and remains within the configuration session.

- **End** – prompts you to save (**Yes**), discard (**No**), or stay in the session (**Cancel**).

The management Ethernet interface is configured with an IP address, subnet mask, and default route.

What to do next

For details about remote access, see Configure Telnet and Configure SSH.

Configure Telnet

Use this procedure to configure Telnet access for remote management on Cisco NCS 1001 and set the maximum number of allowed Telnet sessions.

Enable Telnet server access for remote system management and specify the session limit to control concurrent access.

With a terminal emulation program, establish a telnet session to the management interface port using its IP address.

Follow these steps to configure Telnet server access and session limits:

Procedure

1. Enter the configuration mode.

   Example

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

2. Use the **telnet {ipv4 | ipv6} server max-servers** *limit* command to specify the number of allowed Telnet server sessions.

   Up to 100 Telnet servers are allowed. By default, no Telnet servers are allowed. You must configure this command to enable the use of Telnet servers.

   Example

   ```
   RP/0/RP0/CPU0:ios(config)# telnet ipv4 server max-servers 10
   ```

3. Use the **commit** or **end** command.
   - **Commit** – saves the configuration changes and remains within the configuration session.
   - **End** – prompts you to save (**Yes**), discard (**No**), or stay in the session (**Cancel**).

Telnet server access is enabled according to the session limit you set.

What to do next

For details about secure remote access, see Configure SSH.

Configure SSH

Use this procedure to configure and enable SSH Version 2 access on Cisco NCS 1001, and verify secure remote administration capability.

Enable secure remote access to the Cisco NCS 1001 management interface using SSH Version 2.

Use a terminal emulation program to establish an SSH connection to the management interface IP address for remote administration.

Before you begin

- Install the `ncs1001-k9sec` package on Cisco NCS 1001. For details about package installation, see Install packages on page 82.
- Generate the crypto key for SSH by using the **crypto key generate dsa** command.

Follow these steps to configure and enable SSH Version 2 access:

Procedure

1. Enter the configuration mode.

   Example

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

2. Use the **ssh server v2** command to enable the SSH server to accept only SSHv2 client connections

   Example

   ```
   RP/0/RP0/CPU0:ios(config)# ssh server v2
   ```

3. Use the **commit** or **end** command.
   - **Commit** – saves the configuration changes and remains within the configuration session.
   - **End** – prompts you to save (**Yes**), discard (**No**), or stay in the session (**Cancel**).

4. Use the **show ssh session details** command to verify that the SSH sessions are established.

   Example

   ```
   RP/0/RP0/CPU0:ios# show ssh session details
   ```

SSH Version 2 server access is enabled, and you can verify established SSH sessions for remote administration.

What to do next

For details about time synchronization, see Synchronize the system clock with an NTP server.

Synchronize the system clock with an NTP server

Use this procedure to configure a Network Time Protocol server so XR and System Admin clocks remain synchronized with accurate network time.

Synchronize the XR clock with an NTP server so the System Admin clock can synchronize with the XR clock.

Before you begin

Configure and connect to the management port.

XR and System Admin have independent system clocks. To prevent these clocks from deviating from true time, synchronize them with the clock of an NTP server.

Procedure

1. Enter XR configuration mode.

   Example

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

2. Configure the NTP server.

   Example

   ```
   RP/0/RP0/CPU0:ios# ntp server 198.51.100.4
   ```

The XR clock synchronizes with the specified NTP server, and the System Admin clock synchronizes with the XR clock.

2 Cisco NCS 1001 Preliminary Checks

Topics:

Guides users through essential Cisco NCS 1001 preliminary checks, including verifying hardware, software, firmware, management interfaces, alarms, and environmental parameters to ensure the system is ready for configuration.

Preliminary checks verify the default Cisco NCS 1001 setup after console login and identify issues before further system configuration.

Preliminary checks cover these system areas:

- Hardware, software, firmware, and inventory information: Confirms that the required system information is available.
- Management, alarm, and environmental status: Verifies that the reported values are as expected.
- Setup issues: Identifies conditions that require corrective action before further configuration.

Run the preliminary checks immediately after you log in to the console. If a check detects a setup issue, take corrective action before you continue the system configuration.

> **Note**
>
> The procedure examples do not show output from the latest software release. Output that explicitly identifies a release changes with the current release.

Perform preliminary checks

This section explains how preliminary checks verify the default Cisco NCS 1001 setup after login and help prevent configuration issues.

A preliminary check is a post-login verification task that

- confirms that hardware, software, firmware, and inventory information is available,
- verifies that management, alarm, and environmental status values are as expected, and
- helps identify setup issues that require corrective action before further configuration.

Run preliminary checks immediately after you log in to the console. If any setup issue is detected, take corrective action before proceeding with further system configuration.

> **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.

Verify status of hardware components

Use this procedure to verify that Cisco NCS 1001 hardware components are installed, visible, and operating in the expected state.

Confirm that the Cisco NCS 1001 hardware and software components are operational and properly installed.

Use Cisco IOS XR EXEC mode and System Admin EXEC mode to verify that the system displays the expected components and that the states are operational.

Before you begin

Ensure that all required hardware components are installed on Cisco NCS 1001. For installation details, see *Cisco Network Convergence System 1001 Hardware Installation Guide*.

Follow these steps to verify the status of hardware components:

Procedure

1. Enter the **show platform** command to display the Cisco IOS XR platform state.

   Example

```
RP/0/RP0/CPU0:ios# show platform
Sun Mar  5 02:33:53.075 CET
Node               Type                         State             Config state
--------------------------------------------------------------------------------

0/0                NCS1001-K9                   OPERATIONAL       NSHUT
0/3                NCS1K-EDFA                   OPERATIONAL       NSHUT
0/RP0/CPU0         NCS1K-CNTLR2(Active)         IOS XR RUN        NSHUT
0/FT0              NCS1K1-FAN                   OPERATIONAL       NSHUT
```

```
0/FT1              NCS1K1-FAN                  OPERATIONAL       NSHUT
0/FT2              NCS1K1-FAN                  OPERATIONAL       NSHUT
0/FT3              NCS1K1-FAN                  OPERATIONAL       NSHUT
```

The output must show Cisco IOS XR in the **IOS XR RUN** state and installed components in the **OPERATIONAL** state.

a) If the Cisco IOS XR is not operational, no output is shown in the result. In this case, verify the state of service domain router (SDR) on the node using the **show sdr** command in Cisco IOS XR mode.

This example shows sample output from the **show sdr** command in Cisco IOS XR mode.

```
RP/0/RP0/CPU0:ios# show sdr
Sun Mar  5 02:37:09.174 CET
Type                 NodeName       NodeState      RedState
PartnerName
-------------------------------------------------------------------------------

NCS1001-K9           0/0            OPERATIONAL                   N/A

NCS1K-EDFA           0/3            OPERATIONAL                   N/A

RP                   0/RP0/CPU0     IOS XR RUN     ACTIVE         NONE

NCS1K-CNTLR2         0/RP0          OPERATIONAL                   N/A

NCS1K1-FAN           0/FT0          OPERATIONAL                   N/A

NCS1K1-FAN           0/FT1          OPERATIONAL                   N/A

NCS1K1-FAN           0/FT2          OPERATIONAL                   N/A

NCS1K1-FAN           0/FT3          OPERATIONAL                   N/A
```

2. Enter the **admin** command to enter the System Admin EXEC mode.

Example

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

3. Enter the **show platform** to information and status for each node in the system.

Example

```
sysadmin-vm:0_RP0# show platform
Sun Mar  5  01:38:22.282 UTC
Location  Card Type                HW State      SW State      Config State
-------------------------------------------------------------------------------
0/0       NCS1001-K9               OPERATIONAL   N/A           NSHUT
0/3       NCS1K-EDFA               OPERATIONAL   N/A           NSHUT
0/RP0     NCS1K-CNTLR2             OPERATIONAL   OPERATIONAL   NSHUT
0/FT0     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
0/FT1     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
0/FT2     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
0/FT3     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
```

Verify that all components of the NCS 1001 are displayed in the result. The software state and the hardware state must be in the OPERATIONAL state. The various hardware and software states are:

Hardware states:

- OPERATIONAL–Node is operating normally and is fully functional.
- POWERED_ON–Power is on and the node is booting up.
- FAILED–Node is powered on but has experienced some internal failure.
- PRESENT–Node is in the shutdown state.
- OFFLINE–User has changed the node state to OFFLINE. The node is accessible for diagnostics.

Software states:

- OPERATIONAL–Software is operating normally and is fully functional.
- SW_INACTIVE–Software is not completely operational.
- FAILED–Software is operational but the card has experienced some internal failure.
- N/A–Valid option for modules where software is not running.

4. Enter the **show platform detail** command to display the hardware and software states, and other details of the node.

Example

```
sysadmin-vm:0_RP0# show platform detail
Sun Mar  5  01:39:45.411 UTC

Platform Information for 0/0
 PID :               NCS1001-K9
 Description :       "Network Convergence System 1001 line system 3 slots"
 VID/SN :            V00
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.1
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Event OK"

Platform Information for 0/3
 PID :               NCS1K-EDFA
 Description :       "Network Convergence System 1000 amplifier module"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.1
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Operational"

Platform Information for 0/RP0
 PID :               NCS1K-CNTLR2
 Description :       "Network Convergence System 1000 Controller 2"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     OPERATIONAL
 Configuration :     "NSHUT RST"
 HW Version :        0.1
 Last Event :        HW_EVENT_OK
```

```
 Last Event Reason : UNKNOWN

Platform Information for 0/FT0
 PID :               NCS1K1-FAN
 Description :       "Network Convergence System 1001 Fan"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.0
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Operational"

Platform Information for 0/FT1
 PID :               NCS1K1-FAN
 Description :       "Network Convergence System 1001 Fan"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.0
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Operational"
```

5. Enter the **show inventory** command to display the details of the physical entities of the NCS 1001 when you execute this command in the Cisco IOS XR EXEC mode.

Example

```
RP/0/RP0/CPU0:ios# show inventory
Sun Mar  5 02:42:04.865 CET
NAME: "0/0", DESCR: "Network Convergence System 1001 line system 3 slots"
PID: NCS1001-K9        , VID: V00, SN: CAT2018B033

NAME: "0/3", DESCR: "Network Convergence System 1000 amplifier module"
PID: NCS1K-EDFA        , VID: V01, SN: IIF2044002L

NAME: "0/3-PORT-0", DESCR: "Cisco SFP Pluggable Optics Module"
PID: ONS-SC-Z3-1510    , VID: V02 , SN: FNS200801EK

NAME: "0/RP0", DESCR: "Network Convergence System 1000 Controller 2"
PID: NCS1K-CNTLR2      , VID: V01, SN: CAT2051B0R5

NAME: "0/RP0-SFP-PORT", DESCR: "Unqualified SFP Pluggable Optics Module"
PID: UNQUALIFIED-SFP   , VID: N/A, SN: N/A

NAME: "Rack 0", DESCR: "Network Convergence System 1001 line system 3 slots"
PID: NCS1001-K9        , VID: V00, SN: CAT2018B033

NAME: "0/FT0", DESCR: "Network Convergence System 1001 Fan"
PID: NCS1K1-FAN        , VID: V01, SN: N/A

NAME: "0/FT1", DESCR: "Network Convergence System 1001 Fan"
PID: NCS1K1-FAN        , VID: V01, SN: N/A

NAME: "0/FT2", DESCR: "Network Convergence System 1001 Fan"
PID: NCS1K1-FAN        , VID: V01, SN: N/A

NAME: "0/FT3", DESCR: "Network Convergence System 1001 Fan"
PID: NCS1K1-FAN        , VID: V01, SN: N/A
```

```
NAME: "0/PM0", DESCR: "Network Convergence System 1000 2KW AC PSU 2"
PID: NCS1K-2KW-AC2     , VID: V01, SN: POG2049JT21

NAME: "0/PM1", DESCR: "Network Convergence System 1000 2KW AC PSU 2"
PID: NCS1K-2KW-AC2     , VID: V01, SN: POG2049JT01
```

All installed Cisco NCS 1001 hardware components are visible and display expected operational states.

Verify node status

Use this procedure to verify the operational status of all Cisco NCS 1001 nodes from XR EXEC and System Admin EXEC modes.

Ensure all nodes display expected hardware and software states.

You can verify node status by using the **show platform** command independently from Cisco IOS XR EXEC mode and System Admin EXEC mode.

Follow these steps to verify node status:

Procedure

1. Enter the **show platform** command to display Cisco IOS XR platform status.

   Example

   ```
   RP/0/RP0/CPU0:ios# show platform
   Sun Mar  5 02:53:27.755 CET
   Node               Type                          State             Config state
   ----------------------------------------------------------------------------------

   0/0                NCS1001-K9                    OPERATIONAL       NSHUT
   0/3                NCS1K-EDFA                    OPERATIONAL       NSHUT
   0/RP0/CPU0         NCS1K-CNTLR2(Active)          IOS XR RUN        NSHUT
   0/FT0              NCS1K1-FAN                    OPERATIONAL       NSHUT
   0/FT1              NCS1K1-FAN                    OPERATIONAL       NSHUT
   0/FT2              NCS1K1-FAN                    OPERATIONAL       NSHUT
   0/FT3              NCS1K1-FAN                    OPERATIONAL       NSHUT
   ```

   If Cisco IOS XR is not operational and no output appears, use **show sdr** in System Admin EXEC mode to verify the SDR state on the node.

2. Use the **admin** command to enter System Admin EXEC mode.

   Example

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

3. Enter the **show platform** command to display information and status for each node.

Example

```
sysadmin-vm:0_RP0# show platform
Sun Mar  5  01:56:15.749 UTC
Location  Card Type                HW State      SW State      Config State
--------------------------------------------------------------------------------
0/0       NCS1001-K9               OPERATIONAL   N/A           NSHUT
0/3       NCS1K-EDFA               OPERATIONAL   N/A           NSHUT
0/RP0     NCS1K-CNTLR2             OPERATIONAL   OPERATIONAL   NSHUT
0/FT0     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
0/FT1     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
0/FT2     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
0/FT3     NCS1K1-FAN               OPERATIONAL   N/A           NSHUT
```

Verify that all the modules of NCS 1001 are displayed in the result. The software state and the hardware state must be in the OPERATIONAL state. The various hardware and software states are:

Hardware states:

- OPERATIONAL–Node is operating normally and is fully functional.
- POWERED_ON–Power is on and the node is booting up.
- FAILED–Node is powered on but has experienced some internal failure.
- PRESENT–Node is in the shutdown state.
- OFFLINE–User has changed the node state to OFFLINE. The node is accessible for diagnostics.

Software states:

- OPERATIONAL–Software is operating normally and is fully functional.
- DIAG_MODE–User has changed the card state to OFFLINE for diagnosis.
- SW_INACTIVE–Software is not completely operational.
- FAILED–Software is operational but the card has experienced some internal failure.
- N/A–Valid option for modules where software is not running.

4. Enter the **show platform detail** command to display the detailed platform state information for each node.

Example

```
sysadmin-vm:0_RP0# show platform detail
Sun Mar  5  01:57:40.918 UTC

Platform Information for 0/0
 PID :               NCS1001-K9
 Description :       "Network Convergence System 1001 line system 3 slots"
 VID/SN :            V00
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.1
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Event OK"

Platform Information for 0/3
 PID :               NCS1K-EDFA
 Description :       "Network Convergence System 1000 amplifier module"
```

```
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.1
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Operational"

Platform Information for 0/RP0
 PID :               NCS1K-CNTLR2
 Description :       "Network Convergence System 1000 Controller 2"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     OPERATIONAL
 Configuration :     "NSHUT RST"
 HW Version :        0.1
 Last Event :        HW_EVENT_OK
 Last Event Reason : UNKNOWN

Platform Information for 0/FT0
 PID :               NCS1K1-FAN
 Description :       "Network Convergence System 1001 Fan"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.0
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Operational"

Platform Information for 0/FT1
 PID :               NCS1K1-FAN
 Description :       "Network Convergence System 1001 Fan"
 VID/SN :            V01
 HW Oper State :     OPERATIONAL
 SW Oper State :     N/A
 Configuration :     "NSHUT RST"
 HW Version :        0.0
 Last Event :        HW_EVENT_OK
 Last Event Reason : "HW Operational"
```

All expected nodes appear in the output with acceptable operational hardware and software states.

Verify software version

Use this procedure to verify the Cisco IOS XR software version running on NCS 1001 to determine if a system upgrade is required.

Ensure NCS 1001 operates with the latest Cisco IOS XR software by verifying the installed version.

NCS 1001 ships with Cisco IOS XR software pre-installed. To ensure access to the latest features, you should verify whether the installed version is current and perform a system upgrade if necessary.

Follow these steps to verify the Cisco IOS XR software version on NCS 1001:

Procedure

Enter the **show version** command to display the software version and system uptime.

Example

```
RP/0/RP0/CPU0:ios#show version
Mon Feb 28 15:52:01.424 UTC
Cisco IOS XR Software, Version 7.3.2
Copyright (c) 2013-2021 by Cisco Systems, Inc.

Build Information:
Built By     : deenayak
Built On     : Mon Jul 28 01:19:52 PST 2020
Built Host   : iox-lnx-071
Workspace    : /auto/srcarchive15/prod/7.3.2/ncs1001/ws
Version      : 7.3.2
Location     : /opt/cisco/XR/packages/
Label        : 7.3.2
cisco NCS-1001 () processor
```

The output displays the Cisco IOS XR software version, build information, package location, label, and system uptime.

What to do next

Review the output to determine whether a system upgrade is required. For details about system upgrades, see Install packages on page 82.

Verify firmware version

Use this procedure to verify installed firmware on NCS 1001 components to confirm compatibility with the current Cisco IOS XR image.

Ensure that the firmware installed on NCS 1001 hardware components is compatible with Cisco IOS XR, preventing malfunctions and system issues.

Firmware incompatibility can cause NCS 1001 devices to malfunction. Regularly verify firmware versions on all hardware components to maintain system stability and ensure compatibility with Cisco IOS XR.

Follow these steps to verify the firmware version on NCS 1001 hardware components:

Procedure

Enter the **show hw-module fpd** comand to display firmware information for hardware components.

Example

```
RP/0/RP0/CPU0:ios#show hw-module fpd
Thu Jan 16 15:28:28.146 CEST
          FPD Versions

        ================
Location      Card type            HWver     FPD device        ATR     Status
```

```
      Running   Programd
-------------------------------------------------------------------------------------------
0/0           NCS1001-K9            0.1       Control_BKP        B      CURRENT
                1.10
0/0           NCS1001-K9            0.1       Control_FPGA              CURRENT
       1.10     1.10
0/1           NCS1K-EDFA            0.0       FW_EDFAv1                 CURRENT
       1.60     1.60
0/2           NCS1K-PSM             0.0       FW_PSMv1                  CURRENT
       1.51     1.51
0/3           NCS1K-EDFA            0.0       FW_EDFAv1                 CURRENT
       1.60     1.60
0/RP0         NCS1K-CNTLR2          0.1       BIOS_Backup        BS     CURRENT
       15.10    15.10
0/RP0         NCS1K-CNTLR2          0.1       BIOS_Primary       S      CURRENT
       15.10    15.10
0/RP0         NCS1K-CNTLR2          0.1       Daisy_Duke_BKP     BS     CURRENT
                0.20
0/RP0         NCS1K-CNTLR2          0.1       Daisy_Duke_FPGA    S      CURRENT
       0.20     0.20
```

Displays the firmware information of various hardware components of the NCS 1001 in the Cisco IOS XR EXEC mode.

In the above output, some of the significant fields are:

- FPD Device: name of the hardware component such as FPD, CFP, and so on.
- ATR: attribute of the hardware component. Some of the attributes are:
  - B: backup Image
  - S: secure Image
  - P: protected Image
- Status: Upgrade status of the firmware. The different states are:
  - CURRENT: The firmware version is the latest version.
  - READY: The firmware of the FPD is ready for an upgrade.
  - NOT READY: The firmware of the FPD is not ready for an upgrade.
  - NEED UPGD: A newer firmware version is available in the installed image. It is recommended that an upgrade be performed.
  - RLOAD REQ: The upgrade has been completed, and the ISO image requires a reload.
  - UPGD DONE: The firmware upgrade is successful.
  - UPGD FAIL: The firmware upgrade has failed.
  - BACK IMG: The firmware is corrupted. Reinstall the firmware.
  - UPGD SKIP: The upgrade has been skipped because the installed firmware version is higher than the one available in the image.
- Running: Current version of the firmware running on the FPD.

The output shows whether each firmware component is current or requires upgrade action.

What to do next

If firmware must be upgraded, use the **upgrade hw-module location all fpd all** command in Cisco IOS XR EXEC mode.

If the status column shows **RLOAD REQ** after upgrade, reload the required location. If the FPGA location is `0/RP0`, use **admin hw-module location 0/RP0 reload**. If the FPGA location is `0/0`, use **admin hw-module location all reload**.

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

Verify the management interface status

Use this procedure to verify the NCS 1001 management Ethernet interface state, counters, and configuration before continuing system setup.

Ensure the management interface is up and properly configured before you continue system setup.

You can use interface status commands in Cisco IOS XR EXEC mode to verify the management interface state.

Follow these steps to verify the management interface status:

Procedure

Enter the **show interfaces mgmtEth** *instance* command to display the management interface configuration and counters.

Example

```
RP/0/RP0/CPU0:ios# show interfaces MgmtEth 0/RP0/CPU0/0
Sun Mar  5 03:21:33.272 CET
MgmtEth0/RP0/CPU0/0 is up, line protocol is up
  Interface state transitions: 1
  Hardware is Management Ethernet, address is 6c9c.ed50.2aa2 (bia 6c9c.ed50.2aa2
)
  Internet address is 10.58.229.131/22
  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 2d12h
  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 16000 bits/sec, 22 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     4959018 packets input, 462164262 bytes, 0 total input drops
     0 drops for unrecognized upper-level protocol
     Received 3531513 broadcast packets, 1419827 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     20720 packets output, 1284846 bytes, 0 total output drops
     Output 0 broadcast packets, 0 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions
```

In the above result, the management interface is administratively down.

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

Example

```
RP/0/RP0/CPU0:ios# show interfaces summary
Sun Mar  5 03:22:45.830 CET
Interface Type          Total    UP       Down     Admin Down
--------------          -----    --       ----     ----------
ALL TYPES               2        2        0        0
--------------
IFT_ETHERNET            1        1        0        0
IFT_NULL                1        1        0        0
```

Example

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

```
RP/0/RP0/CPU0:ios# show interfaces brief
Sun Mar  5 03:23:55.330 CET

               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
```

The management interface status output shows the interface state, line protocol, address, MTU, bandwidth, and traffic counters.

What to do next

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

- Check the Ethernet cable connection.
- Verify the IP configuration of the management interface. For details on configuring the management interface, see the *Bring-up NCS 1001* chapter.
- Verify whether the management interface is in the no shut state using the **show running-config interface mgmtEth** command.

  This 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
  Sun Mar  5 03:25:26.191 CET
  interface MgmtEth0/RP0/CPU0/0
   ipv4 address 10.58.229.131 255.255.252.0
  !
  ```

  In the above output, the management interface is in the no shut state.

Verify alarms

Use this procedure to verify active, historical, brief, or detailed alarm information and confirm whether NCS 1001 reports any current conditions.

Review alarm information to identify active or historical issues.

The **show alarms** command displays alarm information in brief or detailed form.

Follow these steps to verify the alarms:

Procedure

Enter the **show alarms brief card | rack | system location** *location* **active | history detail card | rack | system location** *location* **active | clients | history | stats** commands to displays alarms in brief or detail.

Example

```
RP/0/RP0/CPU0:ios# show alarms brief card location 0/RP0/CPU0 active
Sun Mar  5 03:27:57.137 CET

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



------------------------------------------------------------------------------
0/3            Critical     Controller      03/02/2017 14:51:45 CET    Ots0/3/
0/2 - Output OTS Power Reading Below The Fail-Low Threshold

0/3            Minor        Controller      03/04/2017 06:32:27 CET    Optics0
/3/0/4 - Optics Low Receive Power
```

The output displays alarm location, severity, group, set time, and description, helping you confirm whether NCS 1001 reports any current conditions.

What to do next

For more information about alarms and steps to clear them, see the *Alarm Troubleshooting* chapter of the *Cisco NCS 1001 Troubleshooting Guide*.

Verify environmental parameters

Use this procedure to verify NCS 1001 fan, temperature, power, voltage, and current environmental values from the System Admin EXEC mode.

Confirm that environmental parameters are within expected values.

The **show environment** command displays environmental parameters for NCS 1001.

Follow these steps to verify environmental parameters:

Procedure

1. Use the **admin** command to enter System Admin EXEC mode.

   Example

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

2. Enter the **show environment all fan power voltages current temperatures location** *location* commands to displays the environmental parameters of the NCS 1001.

   Example

   This example shows sample output from the **show environment** command with the **fan** keyword.

   ```
   sysadmin-vm:0_RP0# show environment fan
   Sun Mar  5  02:33:51.700 UTC
   ====================================
                           Fan speed (rpm)
   Location     FRU Type           FAN_0
   ------------------------------------
   0/FT0        NCS1K1-FAN         11640
   0/FT1        NCS1K1-FAN         11640
   0/FT2        NCS1K1-FAN         11400
   0/FT3        NCS1K1-FAN         11640
   0/PM0        NCS1K-2KW-AC2       9696
   0/PM1        NCS1K-2KW-AC2       9760
   ```

   This example shows sample output from the **show environment** command with the **temperatures** keyword.

   ```
   sysadmin-vm:0_RP0# show environment temperatures location 0/RP0
   Sun Mar  5  02:34:55.985 UTC
   ================================================================================

   Location  TEMPERATURE                Value   Crit Major Minor Minor Major
   Crit

             Sensor                    (deg C)    (Lo)  (Lo)   (Lo)   (Hi)   (Hi)
   (Hi)

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

   0/RP0
             Thermistor 1                 40    -10     0      0     55     55
     85

             Thermistor 2                 41    -10     0      0     55     55
     85

             Hot Spot Temperature         40    -10     0      0     55     55
     85
   ```

   This example shows sample output from the **show environment** command with the **power** keyword.

   ```
   sysadmin-vm:0_RP0# show environment power
   Sun Mar  5  02:36:17.380 UTC
   ================================================================================
   ```

```
CHASSIS LEVEL POWER INFO: 0
================================================================================


   Total output power capacity (N + 1)             :    2000W +    2000W
   Total output power required                      :     269W
   Total power input                                :     211W
   Total power output                               :      67W

Power Group 0:
================================================================================


   Power       Supply     ------Input----   ------Output---      Status
   Module      Type        Volts     Amps    Volts     Amps
================================================================================


   0/PM0       2kW-AC      235.0      0.4     12.0      1.1    OK

Total of Power Group 0:         94W/  0.4A        13W/  1.1A

Power Group 1:
================================================================================


   Power       Supply     ------Input----   ------Output---      Status
   Module      Type        Volts     Amps    Volts     Amps
================================================================================


   0/PM1       2kW-AC      234.5      0.5     12.0      4.5    OK

Total of Power Group 1:        117W/  0.5A        54W/  4.5A

================================================================================


   Location     Card Type            Power       Power       Status
                                     Allocated   Used
                                     Watts       Watts
================================================================================


   0/0          NCS1001-K9             30           -       ON
   0/1                -                68           -       RESERVED
   0/2                -                68           -       RESERVED
   0/3          NCS1K-EDFA             68           -       ON
   0/RP0        NCS1K-CNTLR2           35           -       ON
   0/FT0        NCS1K1-FAN              0           -       ON
   0/FT1        NCS1K1-FAN              0           -       ON
   0/FT2        NCS1K1-FAN              0           -       ON
   0/FT3        NCS1K1-FAN              0           -       ON
```

This example shows sample output from the **show environment** command with the **voltages** keyword.

```
sysadmin-vm:0_RP0# show environment voltages location 0/RP0
Sun Mar  5  02:37:24.468 UTC
================================================================================

Location  VOLTAGE                    Value   Crit Minor Minor  Crit
          Sensor                     (mV)    (Lo) (Lo)  (Hi)   (Hi)
```

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

0/RP0
          VP1P0_CPU                     1002    900   950  1050  1100
          CPU_CORE_VCC                   713    400   450  1350  1400
          CPU_CORE_VNN                   952    400   450  1350  1400
          VP1P1                         1077    990  1050  1160  1210
          VP1P2                         1206   1080  1140  1260  1320
          VP1P35_DDR                    1353   1220  1280  1420  1490
          VP1P35                        1346   1220  1280  1420  1490
          VP1P5                         1503   1350  1430  1580  1650
          VP1P8_CPU                     1801   1620  1710  1890  1980
          VP3P3_STBY                    3323   2970  3140  3470  3630
          VP3P3                         3346   2970  3140  3470  3630
          VP5P0                         5029   4500  4750  5250  5500
          VP12P0                       12047  10800 11400 12600 13200
          VREF                          1224   1190  1200  1240  1250
          12V Input Voltage            11208   8000 10000 14000 16000
```

The output displays the requested environmental values and thresholds for the selected component or location.

What to do next

Environmental parameter anomalies are logged in syslog. If a displayed environmental parameter is not as expected, use the **show logging** command to review problem details.

Verify inventory

Use this procedure to verify the hardware inventory of the NCS 1001 using the show inventory command..

Ensure accurate hardware information is available for NCS 1001 devices.

The **show inventory** command displays hardware inventory details, including product IDs, version IDs, and serial numbers.

Follow these steps to verify the inventory for NCS 1001:

Procedure

1. Enter the **show inventory** command to display inventory details in Cisco IOS XR EXEC mode.

   Example

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

Sun Mar  5  02:42:57.359 UTC

 Name: Rack 0               Descr: Network Convergence System 1001 line
system 3 slots
 PID: NCS1001-K9            VID: V00                  SN: CAT2018B033

 Name: 0/0                  Descr: Network Convergence System 1001 line
system 3 slots
 PID: NCS1001-K9            VID: V00                  SN: CAT2018B033

 Name: 0/3                 Descr: Network Convergence System 1000 amplifier
```

```
 module
 PID: NCS1K-EDFA             VID: V01                   SN: IIF2044002L

 Name: 0/RP0-SFP-PORT        Descr: Unqualified SFP Pluggable Optics Module
 PID: UNQUALIFIED-SFP        VID:                       SN:

 Name: 0/RP0                Descr: Network Convergence System 1000 Controller
 2
 PID: NCS1K-CNTLR2           VID: V01                   SN: CAT2051B0R5

 Name: 0/FT0                 Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN             VID: V01                   SN: N/A

 Name: 0/FT1                 Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN             VID: V01                   SN: N/A

 Name: 0/FT2                 Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN             VID: V01                   SN: N/A

 Name: 0/FT3                 Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN             VID: V01                   SN: N/A

 Name: 0/PM0                 Descr: Network Convergence System 1000 2KW AC
PSU 2
 PID: NCS1K-2KW-AC2          VID: V01                   SN: POG2049JT21

 Name: 0/PM1                 Descr: Network Convergence System 1000 2KW AC
PSU 2
 PID: NCS1K-2KW-AC2          VID: V01                   SN: POG2049JT01
```

```
NAME: "0/RP0-USB0", DESCR: "NCS 1000 32 chs Even Mux/Demux Patch Panel -
150GHz - C-band"
PID: NCS1K-MD-32E-CE   , VID: V00 , SN: ACW2804YJ23

NAME: "0/RP0-USB1", DESCR: "NCS 1000 32 chs Odd Mux/Demux Patch Panel - 150GHz
 - C-band"
PID: NCS1K-MD-32O-CE   , VID: V00 , SN: ACW2804YH08
```

```
NAME: "0/RP0-USB0", DESCR: "NCS 1000 4 Channel add drop multiplexer Module"
PID: 15216-FLD-4-55.7=   , VID: V00 , SN: OPL19380046
```

2. Use the **admin** command to enter the System Admin EXEC mode.

   Example

```
RP/0/RP0/CPU0:router# admin
```

3. Enter the **show inventory** command to display inventory information for all physical entities.

   Example

```
sysadmin-vm:0_RP0# show inventory
Sun Mar  5  02:44:30.350 UTC

 Name: Rack 0                Descr: Network Convergence System 1001 line
system 3 slots
 PID: NCS1001-K9             VID: V00                   SN: CAT2018B033
```

```
 Name: 0/0                    Descr: Network Convergence System 1001 line
system 3 slots
 PID: NCS1001-K9              VID: V00                    SN: CAT2018B033

 Name: 0/3                   Descr: Network Convergence System 1000 amplifier
 module
 PID: NCS1K-EDFA              VID: V01                    SN: IIF2044002L

 Name: 0/RP0-SFP-PORT         Descr: Unqualified SFP Pluggable Optics Module
 PID: UNQUALIFIED-SFP         VID:                        SN:

 Name: 0/RP0                 Descr: Network Convergence System 1000 Controller
 2
 PID: NCS1K-CNTLR2            VID: V01                    SN: CAT2051B0R5

 Name: 0/FT0                  Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN              VID: V01                    SN: N/A

 Name: 0/FT1                  Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN              VID: V01                    SN: N/A

 Name: 0/FT2                  Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN              VID: V01                    SN: N/A

 Name: 0/FT3                  Descr: Network Convergence System 1001 Fan
 PID: NCS1K1-FAN              VID: V01                    SN: N/A

 Name: 0/PM0                  Descr: Network Convergence System 1000 2KW AC
PSU 2
 PID: NCS1K-2KW-AC2           VID: V01                    SN: POG2049JT21

 Name: 0/PM1                  Descr: Network Convergence System 1000 2KW AC
PSU 2
 PID: NCS1K-2KW-AC2           VID: V01                    SN: POG2049JT01
```

In the above output, the significant fields are:

- **PID**: Physical model name of the chassis or node.
- **VID**: Physical hardware revision of the chassis or node.
- **SN**: Physical serial number for the chassis or node.

The inventory output displays physical entity information for the NCS 1001 chassis, modules, fans, power modules, and connected passive modules.

Display EEPROM data from a passive module

Use this procedure to verify insertion loss and manufacturing details stored in EEPROM for passive modules connected to NCS 1001.

Retrieve and validate important insertion loss values and manufacturing information from passive modules using the built-in USB port on NCS 1001.

Follow these steps to verify EEPROM data from a passive module:

Procedure

1. Enter the **show hw-module usb-port 0 passive-eeprom il** command to retrieve insertion loss details from the passive device.

   Example
   This is a sample output from NCS1K-MD-32O-CE (OOD)

```
INSERTION LOSS VALUES:
[  1]: from: Com-O-Rx to: Ch-00-Tx IL value:   6.04dB
[  2]: from: Com-O-Rx to: Ch-01-Tx IL value:   5.79dB
[  3]: from: Com-O-Rx to: Ch-02-Tx IL value:   5.81dB
[  4]: from: Com-O-Rx to: Ch-03-Tx IL value:   6.01dB
.....................................................
.....................................................
[ 65]: from: Ch-31-Rx to: Com-O-Tx IL value:   5.46dB
[ 66]: from: Probe-Rx to: Com-O-Tx IL value:  10.61dB
[ 67]: from: Com-O-Rx to: Mon-O-Dmx IL value:  20.30dB
[ 68]: from: Odd-Even-Rx to: Odd-Tx    IL value:   3.51dB
[ 69]: from: Odd-Even-Rx to: Even-Tx   IL value:   3.53dB
[ 70]: from: Com-O-Tx to: Mon-O-Mux IL value:  19.67dB
```

   Example
   This is a sample output from 15216-FLD-4-58.9=

```
INSERTION LOSS VALUES:
[  1]: from: Ch-01-Rx to: Com-Tx   IL value:   2.15dB
[  2]: from: Ch-02-Rx to: Com-Tx   IL value:   1.36dB
[  3]: from: Ch-03-Rx to: Com-Tx   IL value:   1.66dB
.....................................................
[ 10]: from: Exp-Rx   to: Com-Tx   IL value:   1.31dB
[ 11]: from: Com-Rx   to: Mon-Rx   IL value:  20.44dB
[ 12]: from: Com-Tx   to: Mon-Tx   IL value:  19.84dB
```

2. Enter the **show hw-module usb-port 0 passive-eeprom manuf** command to retrieve manufacture related details from the passive device.

   Example
   This is a sample output from NCS1K-MD-32O-CE (OOD)

```
EEPROM DATA:
000:  56 19 04 FF 0D 01 48 00 FA 40 0A 1A 41 0C 01 CB   V.....H..@..A...
010:  8F 4E 43 53 31 4B 2D 4D 44 2D 33 32 4F 2D 43 45   .NCS1K-MD-32O-CE
020:  89 56 30 30 20 DA B3 4E 43 53 20 31 30 30 30 20   .V00 ..NCS 1000
030:  33 32 63 68 73 20 4F 64 64 20 4D 75 78 2F 44 65   32chs Odd Mux/De
.....................................................................
.....................................................................
350:  08 02 14 01 02 13 00 03 35 E7 08 02 12 01 01 13   ........5.......
360:  01 13 43 FF FF FF FF FF FF FF FF FF FF FF FF FF   ..C.............
370:  FF FF FF                                          ...
```

   Example
   This is a sample output from 15216-FLD-4-58.9

```
000:  04 FF 40 48 09 C0 46 03 20 00 84 1C 02 41 00 00   ..@H..F. ....A..
010:  42 41 30 C1 8B 4F 50 4C 32 35 30 39 30 30 30 31   BA0..OPL25090001
020:  D0 8A 46 4C 44 2D 34 2D 35 38 2E 39 C6 8A 57 4F   ..FLD-4-58.9..WO
```

```
030:  4F 4D 41 34 33 57 41 41 CB 91 31 35 32 31 36 2D   OMA43WAA..15216-
................................................................................
................................................................................
0B0:  06 00 01 05 01 01 1F E0 08 01 05 00 01 07 00 14   ................
0C0:  2C E0 08 01 05 01 01 07 01 13 54 FF FF FF FF FF   ,.........T.....
0D0:  FF FF FF FF FF FF FF FF FF FF FF                  ...........
```

3. Enter the **show hw-module usb-port 0 passive-eeprom all** command to retrieve both insertion loss and manufacturing details from the passive device.

Example
This is a sample output from NCS1K-MD-32O-CE (OOD)

```
000:  04 FF 40 48 09 C0 46 03 20 00 84 1C 02 41 00 00   ..@H..F. ....A..
010:  42 41 30 C1 8B 4F 50 4C 32 35 30 39 30 30 30 31   BA0..OPL25090001
020:  D0 8A 46 4C 44 2D 34 2D 35 38 2E 39 C6 8A 57 4F   ..FLD-4-58.9..WO
030:  4F 4D 41 34 33 57 41 41 CB 91 31 35 32 31 36 2D   OMA43WAA..15216-
................................................................................
................................................................................
0B0:  06 00 01 05 01 01 1F E0 08 01 05 00 01 07 00 14   ................
0C0:  2C E0 08 01 05 01 01 07 01 13 54 FF FF FF FF FF   ,.........T.....
0D0:  FF FF FF FF FF FF FF FF FF FF FF                  ...........

EEPROM DATA:
000:  56 19 04 FF 0D 01 48 00 FA 40 0A 1A 41 0C 01 CB   V.....H..@..A...
010:  8F 4E 43 53 31 4B 2D 4D 44 2D 33 32 4F 2D 43 45   .NCS1K-MD-32O-CE
020:  89 56 30 30 20 DA B3 4E 43 53 20 31 30 30 30 20   .V00 ..NCS 1000
030:  33 32 63 68 73 20 4F 64 64 20 4D 75 78 2F 44 65   32chs Odd Mux/De
................................................................................
................................................................
350:  08 02 14 01 02 13 00 03 35 E7 08 02 12 01 01 13   ........5.......
360:  01 13 43 FF FF FF FF FF FF FF FF FF FF FF FF FF   ..C.............
370:  FF FF FF
```

Example
This is a sample output from 15216-FLD-4-58.9

```
INSERTION LOSS VALUES:
[  1]: from: Ch-01-Rx to: Com-Tx   IL value:   2.15dB
[  2]: from: Ch-02-Rx to: Com-Tx   IL value:   1.36dB
[  3]: from: Ch-03-Rx to: Com-Tx   IL value:   1.66dB
................................................
[ 10]: from: Exp-Rx   to: Com-Tx   IL value:   1.31dB
[ 11]: from: Com-Rx   to: Mon-Rx   IL value:  20.44dB
[ 12]: from: Com-Tx   to: Mon-Tx   IL value:  19.84dB

EEPROM DATA:
000:  04 FF 40 48 09 C0 46 03 20 00 84 1C 02 41 00 00   ..@H..F. ....A..
010:  42 41 30 C1 8B 4F 50 4C 32 35 30 39 30 30 30 31   BA0..OPL25090001
020:  D0 8A 46 4C 44 2D 34 2D 35 38 2E 39 C6 8A 57 4F   ..FLD-4-58.9..WO
030:  4F 4D 41 34 33 57 41 41 CB 91 31 35 32 31 36 2D   OMA43WAA..15216-
................................................................................
................................................................................
0B0:  06 00 01 05 01 01 1F E0 08 01 05 00 01 07 00 14   ................
0C0:  2C E0 08 01 05 01 01 07 01 13 54 FF FF FF FF FF   ,.........T.....
0D0:  FF FF FF FF FF FF FF FF FF FF FF                  ...........
```

You have retrieved insertion loss and manufacturing details from the passive module connected to NCS 1001.

3 Cisco NCS 1001 System Upgrades and Feature Packages

Topics:

Provides comprehensive procedures for upgrading Cisco NCS 1001 system software, managing feature packages, and performing firmware updates to ensure device compatibility and optimal operational performance.

Cisco NCS 1001 uses install commands to add and activate system software, feature packages, and software maintenance updates from a network server.

The installation process supports these file types:

- ISO image (`.iso`): Provides a system software image for an upgrade.
- Feature package (`.rpm`): Adds an optional software feature.
- Software Maintenance Upgrade (SMU) file (`.smu`): Applies a software maintenance update.

The install commands access the required files on a network server and activate them on Cisco NCS 1001. If an installed package or SMU causes an issue, you can uninstall it.

Perform system upgrade and install feature packages

This section explains how the Cisco NCS 1001 uses install commands to upgrade system software and install feature packages, ISO images, and software maintenance updates.

A system upgrade and feature package installation is a software management process that

- uses install commands to add and activate ISO images, feature packages, and software maintenance upgrade files,
- accesses the required installation files from a network server, and
- allows uninstallation of packages or SMUs if issues arise after installation.

The system upgrade and package installation processes are executed using install commands on the NCS 1001. The processes involve adding and activating the iso images (`.iso`), feature packages (`.rpm`), and software maintenance upgrade files (`.smu`) on the NCS 1001. These files are accessed from a network server and then activated on the NCS 1001. If the installed package or SMU causes any issue, it can be uninstalled.

Software compatibility matrix

This table lists firmware and software compatibility details for Cisco NCS 1001 releases, including field-programmable device versions supported by each release.

This table provides software compatibility for all firmware versions. This matrix shows the supported firmware (FPD and other programmable devices) versions by release for NCS 1001.

Table 6: Software compatibility matrix

FPD	R6.2.1	R6.2.2	R6.3.1	R6.3.2	R6.5.1	R6.5.2	R7.0.0	R7.0.1	R7.1.1	R7.1.2	R7.2.1	R7.3.1 and R7.3.2	R7.5.1, R7.5.2, and R7.7.1	R7.8.1, R7.9.1, and 7.10.1	R25.4.1
FW_PSMv1	1.38	1.38	1.43	1.45	1.45	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51 1.43 only for 7.10.1	1.43
FW_PSMv2	0.09	0.09	0.12	0.14	0.14	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16
FW_PSMv3	–	–	–	–	–	–	–	–	–	–	–	–	–	1.64 only for 7.10.1	1.64
Control_ BKP	1.07	1.09	1.09	1.09	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10
Control_ FPGA	1.07	1.09	1.09	1.09	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10

FPD	R6.2.1	R6.2.2	R6.3.1	R6.3.2	R6.5.1	R6.5.2	R7.0.0	R7.0.1	R7.1.1	R7.1.2	R7.2.1	R7.3.1 and R7.3.2	R7.5.1, R7.5.2, and R7.7.1	R7.8.1, R7.9.1, and 7.10.1	R2.4.1
FW_EDFAv1	1.39	1.39	1.43	1.43	1.43	1.51	1.54	1.55	1.56	1.56	1.56	1.60	1.60 for R7.5.1 and 1.61 for R7.5.2 and R7.7.1	1.61	1.61
FW_EDFAv2	0.12	0.12	0.28	0.31	0.37	0.37	0.39	0.40	0.40	0.40	0.40	0.43	0.43	0.45	0.45
FW_OTDR_p	NA	NA	NA	NA	5.02	5.02	6.02	6.02	6.02	6.02	6.02	6.03	6.03	6.03	NA
FW_OTDR_s	NA	NA	NA	NA	1.47	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51	NA
PO-PriMCU (AC)	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.0	4.0	4.0	4.0	4.0
PO-PriMCU (DC)	NA	1.10	1.10	1.10	1.10	1.10	1.10	1.14	1.14	1.14	1.14	1.14	1.14	1.14	1.14
BIOS_ Backup	13.50	13.60	13.80	14.20	14.20	14.20	14.20	14.50	14.60	14.60	15.10	15.10	15.10	15.10	15.10
BIOS_ Primary	13.50	13.60	13.80	14.20	14.20	14.20	14.20	14.50	14.60	14.60	15.10	15.10	15.10	15.10 for 7.8.1 and 7.9.1; 15.30 for 7.10.1	15.40
Daisy_ Duke_ BKP	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20
Daisy_ Duke_ FPGA	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20

System upgrades

This section explains how a Cisco NCS 1001 system upgrade installs a new Cisco IOS XR operating system image while updating XR and System Admin environments.

A system upgrade is a process that

- installs a new version of the Cisco IOS XR operating system image on the NCS 1001,
- updates both the XR and System Admin environments during the operation, and
- ensures access to the latest features and improvements on the device.

Upgrading the system is the process of installing a new version of the Cisco IOS XR operating system on the NCS 1001. The NCS 1001 comes pre-installed with the Cisco IOS XR image. However, you can install the new version in order to keep features up to date. The system upgrade operation is performed from the XR mode. However, during system upgrade, the operating systems that run both on the XR and the System Admin get upgraded.

System upgrade is done by installing a base package-Cisco IOS XR Core Bundle plus Manageability Package. The file name for this bundle is `ncs1001-mini-x-7.10.1.iso`. Install this ISO image using **install** commands. For more information about the install process, see Package installation and removal workflows on page 78.

Software upgrade matrix

This table lists supported NCS 1001 software upgrade paths and identifies whether a bridge software maintenance update (SMU) is required for a destination release.

Use this matrix to identify supported upgrade paths for NCS 1001.

This table lists the upgrade paths supported for Cisco NCS 1001. Software downgrades are not supported on the NCS 1001.

Upgrade path		
Source release	**Destination release**	**Bridge SMU**
• R6.3.1 • R6.5.1 • R7.0.1 • R7.1.1 • R7.2.1 • R7.3.1 • R7.8.1 • R7.9.1	R7.10.1	NA
R7.10.1	R25.4.1	• CSCwq24232 • CSCwq24218

Upgrade firmware

Upgrade the NCS 1001 firmware in the required sequence and verify field-programmable device compatibility.

Complete an NCS 1001 firmware upgrade following the specified steps to ensure compatibility across field-programmable devices (FPDs).

This task guides you through upgrading the NCS 1001 firmware, including the recommended sequence for upgrading BIOS, Daisy Duke FPDs, control devices, PSM, and EDFA.

Note

If you are upgrading FPDs from Release 7.8.1 to later releases, refer to upgrade field-programmable devices for serialized upgrade.

Note

The output of this procedure is related to Release 7.7.1. The FPDs to be installed for later releases are different. Refer to software compatibility matrix for FPD versions.

Note

The BIOS and Daisy Duke FPD upgrade from R6.3.x and R6.5.x to R7.1.1, R7.1.2, R7.3.1, R7.3.2, R7.5.1, R7.5.2, R7.7.1, R7.8.1, R7.9.1, or R7.10.1 must be executed with proper steps. We recommend that you read through the entire procedure before performing any upgrade steps.

Follow these steps to upgrade the firmware for NCS 1001:

Procedure

Perform the procedures in this sequence.

a. upgrade BIOS and Daisy Duke FPDs

b. upgrade Control FPGA and Control BKP FPDs

c. upgrade PSM

d. upgrade EDFA

The firmware upgrade sequence completes, with all NCS 1001 devices and FPDs updated and compatible with your target release.

Upgrade BIOS and Daisy Duke FPDs

Use this procedure to upgrade the BIOS and Daisy Duke field-programmable devices, verify upgrade status, and ensure prerequisites are met before proceeding to additional firmware updates.

Ensure BIOS and Daisy Duke field-programmable devices are updated to the required firmware versions before proceeding with other device firmware tasks.

> **Note**
>
> This upgrade process applies to devices running software up to Release 7.7.1.

From Release 7.0.1 and later, the upgrade procedure also updates the secure boot keys.

> **Note**
>
> Ensure that secure boot keys are updated prior to other FPD upgrades. If BIOS and Daisy Duke FPGA FPDs are upgraded without updating the secure boot keys, it can lead to RMA of the device.

> **Note**
>
> - BIOS downgrade is not supported once BIOS FPD is upgraded to 15.10.
> - The 15.10 BIOS FPD version does not have issues for software images prior to Release 7.2.1. If the user needs to downgrade the software image prior to Release 7.2.1, the BIOS FPDs always show the status as "NEED UPGRADE".

Before you begin

- For software upgrades from any release to R7.0.1 and later, disable auto-fpd upgrade using the **fpd auto-upgrade disable** command.
- Install SMUs for R6.3.2 and R6.5.2 software to upgrade the FPDs with secure boot keys and then upgrade the software to R7.0.1 and later.

Procedure

1. Enter the **show hw-module fpd** command from IOS XR console.

Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                                FPD
Versions

=================
Location   Card type             HWver FPD device       ATR Status   Running
 Programd
--------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   NEED UPGD
   1.09
0/0        NCS1001-K9            0.1   Control_FPGA         NEED UPGD  1.09
   1.09
0/1        NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31
   0.31
```

```
0/2         NCS1K-PSM             0.0   FW_PSMv2              NEED UPGD  0.12
   0.12
0/3         NCS1K-EDFA            0.0   FW_EDFAv2             NEED UPGD  0.31
   0.31
0/RP0       NCS1K-CNTLR2          0.1   BIOS_Backup       BS  NEED UPGD
  13.80
0/RP0       NCS1K-CNTLR2          0.1   BIOS_Primary       S  NEED UPGD 13.80
  13.80
0/RP0       NCS1K-CNTLR2          0.1   Daisy_Duke_BKP    BS  NEED UPGD
   0.17
0/RP0       NCS1K-CNTLR2          0.1   Daisy_Duke_FPGA    S  NEED UPGD  0.17
   0.17
```

2. Enter the **upgrade hw-module location 0/RP0 fpd all** command from IOS XR console to upgrade the BIOS and Daisy Duke FPDs.
3. Enter the **show hw-module fpd** command again from IOS XR console to view the status of BIOS and Daisy Duke FPDs.

Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                          FPD
Versions

=================
Location   Card type             HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/0         NCS1001-K9            0.1   Control_BKP      B   NEED UPGD
   1.09
0/0         NCS1001-K9            0.1   Control_FPGA         NEED UPGD  1.09
   1.09
0/1         NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31
   0.31
0/2         NCS1K-PSM             0.0   FW_PSMv2             NEED UPGD  0.12
   0.12
0/3         NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31
   0.31
0/RP0       NCS1K-CNTLR2          0.1   BIOS_Backup      BS  RLOAD REQ
  15.10
0/RP0       NCS1K-CNTLR2          0.1   BIOS_Primary      S  RLOAD REQ 13.80
  15.10
0/RP0       NCS1K-CNTLR2          0.1   Daisy_Duke_BKP   BS  CURRENT
   0.20
0/RP0       NCS1K-CNTLR2          0.1   Daisy_Duke_FPGA   S  RLOAD REQ  0.17
   0.20
```

4. Enter the **admin** command to change to admin console.
5. Enter the **hw-module location 0/RP0 reload** command to reload NCS 1001.

   The system reboots within few seconds.
6. Enter the **show hw-module fpd** command from the admin console to view the status of BIOS and Daisy Duke FPDs.

Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
```

```
                                                                                FPD
Versions

=================
Location   Card type              HWver FPD device       ATR Status   Running
 Programd
--------------------------------------------------------------------------------
0/0        NCS1001-K9             0.1   Control_BKP      B   NEED UPGD
   1.09
0/0        NCS1001-K9             0.1   Control_FPGA         NEED UPGD  1.09
   1.09
0/1        NCS1K-EDFA             0.0   FW_EDFAv2            NEED UPGD  0.31
   0.31
0/2        NCS1K-PSM              0.0   FW_PSMv2             NEED UPGD  0.12
   0.12
0/3        NCS1K-EDFA             0.0   FW_EDFAv2            NEED UPGD  0.31
   0.31
0/RP0      NCS1K-CNTLR2           0.1   BIOS_Backup      BS  CURRENT
  15.10
0/RP0      NCS1K-CNTLR2           0.1   BIOS_Primary      S  CURRENT   15.10
  15.10
0/RP0      NCS1K-CNTLR2           0.1   Daisy_Duke_BKP   BS  CURRENT
   0.20
0/RP0      NCS1K-CNTLR2           0.1   Daisy_Duke_FPGA   S  CURRENT    0.20
   0.20
```

BIOS and Daisy Duke field-programmable devices report the expected status and are successfully upgraded.

What to do next

upgrade Control FPGA and Control BKP FPDs

Upgrade Control FPGA and Control BKP FPDs

Use this procedure to upgrade the Control FPGA and Control BKP field-programmable devices and verify their status before proceeding with firmware upgrades.

Ensure that Control FPGA and Control BKP field-programmable devices are upgraded and report the current state before upgrading other firmware components.

Upgrade the Control FPGA and Control BKP FPDs before upgrading the FPDs of PSM and EDFA optical modules.

Note

This procedure is valid only until Release 7.7.1.

Note

Traffic loss might occur if the following steps are not followed.

Follow these steps to upgrade the Control FPGA and Control BKP FPDs:

Procedure

1. Enter the **show hw-module fpd** command from admin console.

   Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                                FPD
Versions

==================
Location   Card type             HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   NEED UPGD
   1.09
0/0        NCS1001-K9            0.1   Control_FPGA         NEED UPGD  1.09
   1.09
```

2. Enter the **upgrade hw module location 0/0 fpd Control_FPGA** command from admin console to upgrade the Control FPGA FPD.

   Note

   Wait for Control FPGA FPD status to reach the RLOAD REQ state.

3. Enter the **show hw-module fpd** command from IOS XR console to view the status of Control FPGA FPD.

   Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                              FPD
```

```
Versions

=================
Location   Card type            HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/0        NCS1001-K9           0.1   Control_BKP      B   NEED UPGD
   1.09
0/0        NCS1001-K9           0.1   Control_FPGA         RLOAD REQ  1.09
   1.09
```

4. Enter the **upgrade hw-module location 0/0 fpd Control_BKP** command from admin console to upgrade the Control BKP FPD.

   Note

   Wait for Control BKP FPD status to reach the CURRENT state.

5. Enter the **show hw-module fpd** command from IOS XR console to view the status of Control BKP FPD.

   Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                          FPD
Versions

=================
Location   Card type            HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/0        NCS1001-K9           0.1   Control_BKP      B   CURRENT
   1.10
0/0        NCS1001-K9           0.1   Control_FPGA         RLOAD REQ  1.09
   1.10
```

6. Enter the **hw-module location 0/RP0 reload** command from admin console to reload NCS 1001.

   The system reboots within few seconds.

7. Enter the **show hw-module fpd** command from the admin console to view the status of Control FPGA and Control BKP FPDs.

   Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd

                                                                          FPD
Versions

================
```

```
Location   Card type              HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/0        NCS1001-K9             0.1   Control_BKP      B   CURRENT
   1.10
0/0        NCS1001-K9             0.1   Control_FPGA         CURRENT  1.10
   1.10
```

Control FPGA and Control BKP field-programmable devices report the expected current state, indicating that firmware upgrade prerequisites are met.

What to do next

upgrade PSM

Upgrade PSM

Use this procedure to upgrade the PSM field-programmable device, including preparation, upgrade completion, and verification of firmware status.

Upgrade the PSM field-programmable device and verify its firmware status as part of network maintenance.

Note

This procedure is valid only until Release 7.7.1.

Note

When you upgrade the FW_PSMv1 FPD from 1.38 to 1.43 or higher version, traffic is affected for around 120 seconds.

Note

When you upgrade the FW_PSMv1 FPD from 1.43 to higher version, traffic is affected.

Follow these steps to upgrade the PSM field-programmable device:

Procedure

1. Prepare the PSM upgrade.

   For details about prepare PSM upgrade, see Prepare PSM upgrade.

2. Complete the PSM upgrade.

For details about complete PSM upgrade, see Complete PSM upgrade.

The PSM field-programmable device upgrade is complete, and the current firmware state is verified..

What to do next

upgrade EDFA

Prepare the PSM for upgrade

Use this procedure to verify the PSM status, clear any working-path lockouts, and prepare the section-protection state before upgrading the PSM firmware.

Ensure the PSM is ready so that traffic is on the working path prior to a firmware upgrade.

Follow these steps to prepare the PSM for firmware upgrade:

Procedure

---

1. Enter the **show hw-module fpd** command from IOS XR console.

   Example

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                                FPD
Versions

=================
Location   Card type             HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/2        NCS1K-PSM             0.0   FW_PSMv1         NEED UPGD       a.bc
   x.yz
```

2. Enter the **show controllers ots 0/2/0/* summary** command to check the port status.

   If the output shows **Ots0_2_0_2 Protected Active**, switch the PSM state to **Ots0_2_0_1 Working Active** and **Ots0_2_0_2 Protected Standby**.

3. Enter the **config** command.
4. Enter the **no hw-module location 0/RP0/CPU0 slot 2 psm lockout-from WORKING** to move the PSM state to **Ots0_2_0_1 Working Active** without any lockout in place.
5. Enter the **commit** command to save and apply the configuration changes.

The PSM working path is active and ready for firmware upgrade.

What to do next

upgrade EDFA

Complete the PSM upgrade

Use this procedure to configure traffic paths, upgrade the PSM field-programmable device firmware, and verify that the upgrade is successful.

Upgrade the PSM firmware and validate that the field-programmable device (FPD) reaches the current state.

Follow these steps to complete the PSM upgrade:

Procedure

1. Enter the **hw-module slot 2 manual-switch-to WORKING** command from IOS XR console to switch the traffic manually from the protect to the working path.

   PSM switch causes traffic loss for less than 50 ms.

2. Enter the **show controllers ots 0/2/0/* summary** command to check the output.

   If the switch to the Working port is successful, **Ots0_2_0_1 Working** port is reported as **Active**. If the switch to the Working port is successful, check whether the far-end node is also active on the same port.

3. Enter the **show controllers ots 0/2/0/* summary** command on the far-end node to verify that **Ots0_2_0_1 Working** port is reported as **Active**.

   > **Note**
   >
   > If the status of PSM Working port is still Standby, it indicates that the previous switch command was not successful.

   Failure to switch to the Working path indicates that the Working path is alarmed, or a **lockout-from working** is intentionally set in the configuration. In both the cases, further troubleshooting of the overall Working path (near-end node to far-end node) is required to resolve outstanding problems.

   When both the near-end and far-end nodes are **Active** on the Working port, proceed with the following upgrade steps.

4. Enter the **upgrade hw module location 0/2 fpd FW_PSMv1** command from IOS XR console to upgrade the PSMv1 FPD.

   If the FPD type is FW_PSMv2, the above command changes to **upgrade hw module location 0/1 fpd FW_PSMv2**.

   If the configuration includes path protection topology, FW_ PSMv1 upgrade is traffic-affecting and can be done using the force option as follows:

   Enter the **upgrade hw module location 0/2 fpd FW_PSMv1 force** command from IOS XR console to upgrade the PSMv1 FPD.

5. Enter the **show hw-module fpd** command from the admin console to view the status of PSM FPD.

   Example

   ```
   RP/0/RP0/CPU0:ios# show hw-module fpd
   ```

> **Note**
>
> Wait for PSM FPD status to reach the CURRENT state.

The PSM field-programmable device firmware is successfully upgraded, and its status is confirmed as CURRENT.

What to do next

upgrade EDFA

Upgrade EDFA

Use this procedure to configure and upgrade the Erbium-Doped Fiber Amplifier (EDFA) firmware, then verify controller status after the upgrade.

Upgrade the firmware for Erbium-Doped Fiber Amplifier modules to ensure optimal performance and reliability..

> **Note**
>
> This procedure is valid only until Release 7.7.1.

To upgrade each EDFA module present in NCS 1001, you must identify the correct location, based on the specific FPD device.

Follow these steps to upgrade the EDFA firmware and verify controller status:

Procedure

1. Enter the **upgrade hw module location 0/1 fpd FW_EDFAv1 force** command from IOS XR console to upgrade the EDFA FPD.

   If the FPD type is FW_EDFAv2, the above command changes to **upgrade hw module location 0/1 fpd FW_EDFAv2 force**.

2. Enter the **show hw-module fpd** command from IOS XR console to view the status of EDFA FPD.

   > **Note**
   >
   > Wait for EDFA FPD status to reach the CURRENT state.

   Example

   ```
   RP/0/RP0/CPU0:ios# show hw-module fpd
   ```

```
                                                                    FPD Versions


=================
Location   Card type              HWver FPD device       ATR Status   Running
 Programd
-------------------------------------------------------------------------------
0/0        NCS1001-K9             0.1   Control_BKP      B   CURRENT
   1.10
0/0        NCS1001-K9             0.1   Control_FPGA         CURRENT    1.10
   1.10
0/1        NCS1K-EDFA             0.0   FW_EDFAv2            CURRENT    0.43
   0.43
0/2        NCS1K-PSM              0.0   FW_PSMv2             CURRENT    0.16
   0.16
0/3        NCS1K-EDFA             0.0   FW_EDFAv1            CURRENT    1.60
   1.60
0/3        NCS1K-EDFA             0.0   FW_EDFAv1            CURRENT    1.60
   1.60
```

Note

When you upgrade the FW_EDFAv1 FPD from 1.38 to 1.43 or higher version, traffic is affected for around 120 seconds due to restart of line amplifier. FW_EDFAv2 FPD is not affected by this issue.

3. (Only for FW_EDFAv1 FPD) Enter the **admin** command to change to admin console.
4. (Only for FW_EDFAv1 FPD) Enter the **hw-module location 0/slot reload** command to perform hardware reset of the EDFA module.
5. (Only for FW_EDFAv1 FPD) After the reload, check the state of EDFA modules using the **show controllers ots 0/slot/0/0** and **show controllers ots 0/slot/0/0** commands.

   Example

```
RP/0/RP0/CPU0:ios# show controllers ots 0/1/0/0
Fri Jan 22 10:14:29.305 CET
 Controller State: Down
 Transport Admin State: In Service
 Port Type: Com
 Laser State: Off
 Optics Status::
```

6. (Only for FW_EDFAv1 FPD) If the **Laser State** field is reported as **Off**, restart the laser using the **controller ots 0/slot/0/*port* osri on** and **controller ots 0/slot/0/*port*osri off** commands.

The Erbium-Doped Fiber Amplifier firmware is successfully upgraded. The controller status is verified, and laser state is restored as needed.

Upgrade FPDs

Use this procedure to upgrade Cisco NCS 1001 field-programmable devices and verify their status after reload when required for completion.

Ensure that all Cisco NCS 1001 field-programmable devices (FPDs) are upgraded to the current version and properly reloaded to complete the upgrade.

Table 7: Feature History

Feature Name	Release	Description
FPD Upgrade Enhancement	Cisco IOS XR Release 7.8.1	FPD upgrade is made easy with a new command **upgrade hw-module location all fpd all**. It performs the end-to-end upgrade of all FPD modules with a single execution. As a result, the need to make progressive upgrades is eliminated.

From Release 7.8.1, the upgrade of the NCS 1001 FPD devices is serialized and requires no manual intervention. The command works only when the FPD devices are in `NEED UPGD` status.

You can also use the `upgrade hw-module location all fpd all force` command to force the upgrade of the FPD devices. This command upgrades all the components forcefully even if the FPDs are in the current version.

Note

The `upgrade hw-module location all fpd all force` command on FW_PSMv1 from 1.43 to higher version is traffic-affecting.

The FPD devices upgrade in this sequence:

1. `Control_BKP`

   `Control_FPGA`

   Note

   After upgrade of `Control_BKP` and `Control_FPGA` FPDs, the NCS 1001 controller cards reload automatically.

2. FPDs of NCS 1001 cards in slots 1, 2, and 3
3. `BIOS_Backup`

   `BIOS_Primary`

   `Daisy_Duke_BKP`

   `Daisy_Duke_FPGA`

Follow these steps to upgrade the field-programmable devices (FPDs) on NCS 1001:

Procedure

1. Use the **show hw-module fpd** command to check the status of the FPD.

   Example

   This output shows the details of the FPD devices.

```
RP/0/RP0/CPU0:ios#show hw-module fpd
                                     FPD Versions
Location     Card type    HWver   FPD device            ATR    Status
 Running     Programd
0/0          NCS1001-K9   0.1     Control_BKP           B      NEED UPGD
 1.09
0/0          NCS1001-K9   0.1     Control_FPGA                 NEED UPGD
 1.09
0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD
 1.43        1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    NEED UPGD
 6.02        6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    NEED UPGD
 1.47        1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD
 1.43        1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     NEED UPGD
 13.80
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      NEED UPGD
 13.80       13.80
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     NEED UPGD
 0.20
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      NEED UPGD
 0.20        0.20
```

2. Enter the **upgrade hw-module location all fpd all** command to upgrade all the FPD devices.

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

3. Ener the **show hw-module fpd** command to check the status of the FPD devices.

   This output shows the status of `Control_BKP` and `Control_FPGA` as `Current` and `RLOAD REQ`.

   > **Note**
   >
   > The `Control_BKP` and `Control_FPGA` FPD devices reload automatically.

   Example

```
RP/0/RP0/CPU0:ios#show hw-module fpd
                                     FPD Versions
Location     Card type    HWver   FPD device            ATR    Status
 Running     Programd
0/0          NCS1001-K9   0.1     Control_BKP           B      Current
 1.10
```

```
0/0          NCS1001-K9   0.1     Control_FPGA                 RLOAD REQ
 1.10
0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD
 1.43        1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    NEED UPGD
 6.02        6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    NEED UPGD
 1.47        1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD
 1.43        1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     NEED UPGD
 13.80
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      NEED UPGD
 13.80       13.80
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     NEED UPGD
 0.20
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      NEED UPGD
 0.20        0.20
```

4. After the status of the `Control_BKP` and `Control_FPGA` FPD devices becomes `Current`, repeat the previous step to check the status of other FPDs.

   This output shows that all the optical cards are upgraded with the status as `Current`.

   Example

   This output shows that all the optical cards are upgraded with the status as **`Current`**.

```
 FPD Versions
Location     Card type    HWver   FPD device           ATR    Status
 Running     Programd
0/0          NCS1001-K9   0.1     Control_BKP          B      Current
 1.10
0/0          NCS1001-K9   0.1     Control_FPGA                Current
 1.10
0/1          NCS1K-EDFA   0.0     FW_EDFAv1                   Current
 1.43        1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                   Current
 6.02        6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                   Current
 1.47        1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                   Current
 1.43        1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup          BS     NEED UPGD
 13.80
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary         S      NEED UPGD
 13.80       13.80
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP       BS     NEED UPGD
 0.20
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA      S      NEED UPGD
 0.20        0.20
```

5. Enter the **`show hw-module fpd`** command to see if all the FPD devices are upgraded.

Example

This output shows the status of the BIOS and Daisy_Duke FPDs as `RLOAD REQ`.

```
 RP/0/RP0/CPU0:ios#show hw-module fpd
                                      FPD Versions
Location     Card type    HWver   FPD device            ATR    Status
 Running     Programd
0/0          NCS1001-K9   0.1     Control_BKP           B      Current
 1.10
0/0          NCS1001-K9   0.1     Control_FPGA                 Current
 1.10
0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    Current
 1.43        1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    Current
 6.02        6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    Current
 1.47        1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    Current
 1.43        1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     RLOAD REQ
 15.10
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      RLOAD REQ
 15.10       15.10
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     RLOAD REQ
             0.20
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      RLOAD REQ
 0.20        0.20
```

6. Enter the **hw-module location 0/RP0 reload** command to reload the BIOS and Daisy_Duke FPDs.

```
RP/0/RP0/CPU0:ios#hw-module location 0/RP0 reload
```

7. Enter the **show hw-module fpd** command to check the status of the upgraded FPD devices.

Example

This output shows the details of the upgraded FPD devices with their status highlighted as `Current`.

```
 RP/0/RP0/CPU0:ios#show hw-module fpd
                                       FPD Versions
Location     Card type    HWver   FPD device            ATR    Status
 Running     Programd
0/0          NCS1001-K9   0.1     Control_BKP           B      Current
 1.10
0/0          NCS1001-K9   0.1     Control_FPGA                 Current
 1.10
0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    Current
 1.43        1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    Current
 6.02        6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    Current
 1.47        1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    Current
 1.43        1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     Current
 15.10
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      Current
 15.10       15.10
```

```
0/RP0          NCS1K-CNTLR2 0.1      Daisy_Duke_BKP         BS      Current
               0.20
0/RP0          NCS1K-CNTLR2 0.1      Daisy_Duke_FPGA        S       Current
 0.20          0.20
```

All field-programmable devices on Cisco NCS 1001 are upgraded to the current version. Device reloads complete the process, and FPDs report "Current" status.

Feature packages

This section explains how Cisco IOS XR packages and software maintenance updates add specific features to Cisco NCS 1001 and how package filenames identify installable content.

A feature package is a software bundle that

- adds specific functionality to NCS 1001 devices,
- uses package filenames to identify installable content, and
- allows selection and installation of features based on operational requirements.

Packages and software maintenance updates (SMUs) can be installed on NCS 1001 devices. Installing a package adds specific features included within that package. Cisco IOS XR software is divided into various packages, allowing you to select which features to run on NCS 1001. Each package contains components that perform specific functions for the device.

The naming convention of the package is `<platform>-<pkg>-<pkg version>-<release version>.<architecture>.rpm`.

Table 8: Standard packages

Feature Set	Filename	Description
Composite package		
Cisco IOS XR Core Bundle + Manageability Package	ncs1001-iosxr-px-k9-7.10.1.tar	Contains required core packages, including OS, Admin, Base, Forwarding, SNMP Agent, FPD, and Alarm Correlation and Netconf-yang, Telemetry, Extensible Markup Language (XML) Parser, HTTP server packages.
Individually-installable optional packages		
Cisco IOS XR Security Package	ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm	Support for Encryption, Decryption, IP Security (IPSec), Secure Socket Layer (SSL), and Public-key infrastructure (PKI).

Package installation and removal workflows

This section explains how package installation and removal procedures support Cisco NCS 1001 feature packages and software maintenance updates during system management.

Package installation and removal workflows are related package-management procedures that

- install a feature package or software maintenance update on a Cisco NCS 1001, and
- remove the package if it causes an issue.

Package management procedures

Use these procedures to manage packages:

- To install a package, see Install packages.
- To remove a package, see Uninstall packages.

Each linked topic contains a flowchart for its procedure. If an installed software maintenance update or feature package causes an issue, use the removal procedure.

Software repositories for Cisco IOS XR package installation

This section explains how software repositories enable storage and access of Cisco IOS XR packages, upgrades, and updates for installation tasks.

A software repository is a storage location that

- holds Cisco IOS XR packages, upgrades, and update files required for installation,
- enables access for both local and remote installations via supported protocols, and
- allows routers and remote servers to fetch files through FTP, HTTP, HTTPS, or local file paths.

To install packages (RPM), code upgrades, and updates in XR, you need to copy the required RPMs to a reachable repository to download and install. You can host the repository locally on the router or on a remote server that can be accessed via FTP, HTTP, or HTTPS.

When you access the repository remotely, you must provide a repository URL from where the install files are fetched. The URL contains:

- IP address of the server
- Port number of the server

The format of the repository URL is one of the following:

- FTP: ftp://[;]/
- HTTP: http://[;]/
- HTTPS: https://[;]/
- Local: file:///

  The path to the repository must be under /harddisk:/ location, for example, the URL for HTTP server is http://172.16.0.0:3333/.

> **Note**
>
> Username and password are not supported for HTTP and FTP repositories.

Create and configure local repository

Use this procedure to create a local repository on Cisco NCS 1001 when a remote repository is not the preferred method for hosting RPM files.

Create a local repository on the router hard disk and point the install repository configuration to that location.

The router can serve as a repository to host the RPMs. You must be a root-lr user with access to the router shell. Remote repository is the recommended method to access the RPMs. However, if remote repository is not your preferred option, then you can use the router as a repository to host the RPMs.

Using a local repository removes the need to setup an external server for software installation. In this method, the image files are copied directly to the router, and used to create a repository locally. However, on the downside, the files for future updates must be copied to each router individually.

Follow these steps to create and configure a local repository:

Procedure

1. Create a directory locally on the router's /harddisk, for example, "new_repo".

```
[node0_RP0_CPU0:/harddisk:]$mkdir /harddisk\:/new_repo
```

2. Copy the required RPMs and ISO files (using the copy or scp command) from the server to the local directory on the router.
3. Access the shell of the router using run the command and untar the RPMs.

Example

```
Router#run
[node:~]$cd <directory-with-rpms>
[node:~]$tar -xvzf <rpm-name>.tgz
```

4. Exit from the shell.
5. Configure the local repository.

Example

```
RP/0/RP0/CPU0:ios# install repository new_repo url file:/harddisk:/new_repo
RP/0/RP0/CPU0:ios#install commit
RP/0/RP0/CPU0:ios# show running-config install repository new_repo
Mon Mar  6 16:46:45.891 IST install
 repository new_repo   url file:/harddisk:/new_repo  !
!
```

> **Note**
>
> Only the top-level packages are displayed. The contents of the repository are displayed only when the configured repository is valid, and the RPMs are present in the repository. It displays only the packages that are available in the repository and not part of the active system.

The router accesses locally stored RPM and ISO files from the configured repository for software installation and upgrades.

Create and configure external repository

Use this procedure to configure an external repository on a reachable server so multiple Cisco NCS 1001 routers can access common RPM files.

Create repository metadata on an external server and configure Cisco NCS 1001 to access that repository.

To create an external repository, use a server that can be reached over HTTP, HTTPS, or FTP. The following instructions are applicable to Linux distribution systems. Using an external repository provides a central common repository to be used across devices. This eliminates the need to copy files for future updates to each router individually. It also serves as a single source when new RPMs (bug fixes, packages, and updates) are made available. This is the recommended method to setup a repository.

Ensure that you have completed the following tasks:

- Set up your HTTP, HTTPS, or FTP server. Ensure that the server is reachable.
- Install the createrepo utility on the Linux distribution system (if not installed already).

Follow these steps to create and configure an external repository:

Procedure

1. Create a new directory on the server and copy all the RPMs to a directory. This directory hosts the repository and must be accessible to the HTTP, HTTPS, or FTP server that the router will use to access the repository. For example, /var/www/html, is the directory where the repository will be created.

   If the RPM files are archived (.tar format) or compressed (.tgz or .gz format), extract the files. The files hierarchically arrange in sub directories under the main directory that is used as a repository.

2. Convert the directory to a repository by running 'createrepo </path/to/repo-dir/>' . This creates a directory named repodata with the metadata of all the RPMs.

```
[node]$createrepo --database /var/www/html/
Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
[node]$cd /var/www/html/
[node]$ls
Repodata
```

   If you add new packages to the repository, change, or remove packages from the repository, you must run createrepo command again to update the metadata. This ensures that the package manager chooses the correct packages.

3. Configure the external repository.

```
RP/0/RP0/CPU0:ios# install repository new_repo url file:/harddisk:/new_repo
RP/0/RP0/CPU0:ios#install commit
RP/0/RP0/CPU0:ios# show running-config install repository new_repo
Mon Mar  6 16:46:45.891 IST install
repository new_repo   url file:/harddisk:/new_repo  ! !
```

Verify connectivity to the server and check the contents of the repository.

The external repository is available for package installation operations from Cisco NCS 1001 routers.

Install packages

Use this procedure to install NCS 1001 software packages, ISO images, and software maintenance updates.

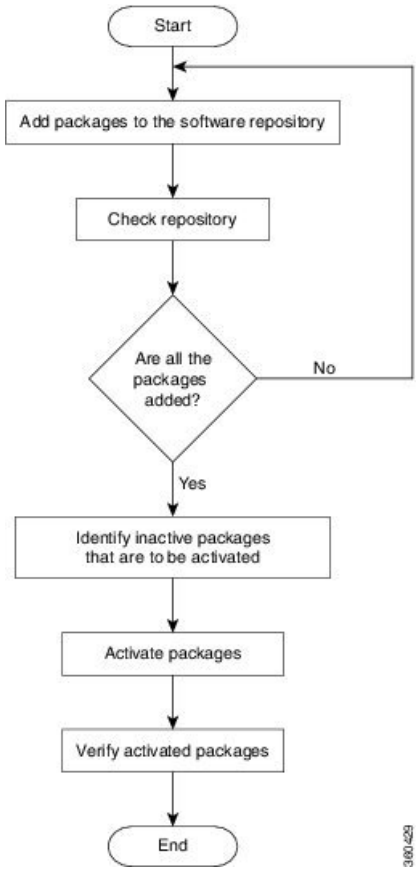
Upgrade the Cisco NCS 1001 system or install patches by adding software packages, ISO images, or software maintenance updates (SMUs). Bundles such as .tar files may contain multiple packages for streamlined updates.

Complete this task to upgrade the system or install a patch. The system upgrade is done using an ISO image file, while the patch installation is done using packages and SMUs. This task is also used to install `.tar` files. The `.tar` file contains multiple packages and SMUs that are merged into a single file. A single `.tar` file can contain up to 64 individual files. The packaging format defines one RPM per component, without dependency on the card type.

> **Note**
>
> To install a System Admin package or a XR package, execute the **install** commands in System Admin EXEC mode or XR EXEC mode respectively. All **install** commands are applicable in both these modes.

The workflow for installing a package is shown in this flowchart.

Figure 3: Installing Packages Workflow



Note

Disable auto-fpd upgrade before start of software upgrade.

```
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
```

Before you begin

- Configure and connect to the management port. You can access the installable file through the management port. For details about configuring the management port, see Configure the management interface on page 34.
- You need to create either local or external repository to copy the tar file from FTP or TFTP server. For more information, See Creating Repository to Access Files for Upgrading IOS XR Software.
- Copy the package to be installed either on the NCS 1001 hard disk or on a network server to which the NCS 1001 has access.
- When ncs1001-k9sec package is not installed, use only FTP or TFTP to copy files or during the **install add** operation.

Procedure

1. Download the install package files.

For details about download install package files, see Download install package files.

2. Add and verify the install packages.

   For details about add and verify install packages, see Add and verify install packages.

3. Activate and commit the install packages.

   For details about activate and commit install packages, see Activate and commit install packages.

The selected packages are downloaded, added to the repository, activated, and committed.

What to do next

- After performing a system upgrade, upgrade FPD by using the **upgrade hw-module location all fpd all** command from the Cisco IOS XR mode. The progress of FPD upgrade process can be monitored using the **show hw-module fpd** command.
- Reload NCS 1001 if any FPD status is in RLOAD REQ state. If CTRL FPGA is in RLOAD REQ state, use the **hw-module location 0/0 reload** command. If Daisy Duke or BIOS is in RLOAD REQ state, use the **hw-module location 0/RP0 reload** command.
- Verify the installation using the **install verify packages** command.
- Uninstall the packages or SMUs if their installation causes any issues on the NCS 1001. See uninstall packages.

> **Note**
>
> ISO images cannot be uninstalled. However, you can perform a system downgrade by installing an older ISO version.

Download install package files

Use this procedure to download the Cisco NCS 1001 software images, copy the install package tar file to your router, and prepare for installation.

Obtain the install package files and ensure they are available on your router or repository for software upgrades.

Use this task when you need to upgrade Cisco NCS 1001 software and must place package files where the install operation can access them.

Procedure

---

1. Log into the Cisco software download site with your Cisco user ID and password.
2. Click **Browse All** to see the product categories.
3. Select the software either by searching across all product categories or by entering the complete product name or a partial string in the search box.

   Example

   For example, NCS1001 software download page is available at Cisco software download site

4. Download the Cisco IOS tar file either to FTP or TFTP server.
5. Copy the tar file from the FTP or TFTP server either to router's harddisk or external repository.

Example

```
RP/0/RP0/CPU0:ios#copy tftp:ncs1001-iosxr-px-k9-7.10.1.tar harddisk
RP/0/RP0/CPU0:ios#copy ftp:ncs1001-iosxr-px-k9-7.10.1.tar harddisk
```

Note

This operation may take time, depending on the size of the files being added. For details about configuring the repository, see Creating Repository to Access Files for Upgrading IOS XR Software.

The install package files are stored on the router hard disk or in the configured repository, ready for use in the upgrade process.

Add and verify install packages

Use this procedure to add package files to the install repository and verify repository, request, log, and inactive-package status before activation.

Add packages to the install repository and verify the add operation.

Follow these steps to add and verify install packages:

Procedure

---

1. Execute **install add** command to extract and install the packages from the tar file.
   - **install add source** *<tftp transfer protocol>/package_path filename1filename2...*
   - **install add source** *<ftp or sftp transfer protocol>//user@server:/package_path filename1filename2...*

   Example

   ```
   RP/0/RP0/CPU0:ios#install add source harddisk:
   ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
   ```

   or

   ```
   RP/0/RP0/CPU0:ios#install add source tftp://10.58.230.32/mystique/iso/7.10.1/
    ncs1001-mini-x-7.10.1.iso ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
   ```

   ```
   Thu Jul 25 14:26:43.661 CEST
   Jul 25 14:26:47 Install operation 13 started by root:
    install add source tftp://10.58.230.32/mystique/iso/7.10.1
   ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm ncs1001-mini-x-7.10.1.iso
   Jul 25 14:26:49 Install operation will continue in the background
   ```

> **Note**
>
> - A space must be provided between the `package_path` and `filename`.
> - Install operation over IPv6 is not supported.

2. Execute **show install repository** to display packages that are added to the repository.

   Example

```
RP/0/RP0/CPU0:router#show install repository
```

3. (Optional) Execute the **show install request** command to display the operation ID of the add operation and its status. The operation ID can be used later to execute the activate command.

   Example

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

```
User root, Op Id 13
install add
ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
ncs1001-mini-x-7.10.1.iso

The install add operation 13 is 30% complete

........
........
Jul 25 14:46:36 Install operation 13 finished successfully
```

4. Execute **show install log***operation ID* to display packages that are added to the repository.

   Example

```
RP/0/RP0/CPU0:ios#show install log 13
```

```
Thu Jul 25 14:48:53.270 CEST
Jul 25 14:26:47 Install operation 13 started by root:
 install add source tftp://10.58.230.32/mystique/iso/7.10.1
ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm ncs1001-mini-x-7.10.1.iso
Jul 25 14:26:48 Action 1: install add action started
Jul 25 14:26:49 Install operation will continue in the background
Jul 25 14:46:34 Packages added:
Jul 25 14:46:34     ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 14:46:34     ncs1001-mini-x-7.10.1.iso
Jul 25 14:46:34 Action 1: install add action completed successfully
Jul 25 14:46:36 Install operation 13 finished successfully
Jul 25 14:46:36 Ending operation 13
```

   Packages are displayed only after the `install add` operation is complete.

5. Execute **show install inactive** to display inactive packages that are present in the repository. Only inactive packages can be activated.

   Example

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

```
Thu Jul 25 14:51:45.852 CEST
2 inactive package(s) found:
ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
ncs1001-mini-x-7.10.1.iso
```

The package files are now present as inactive packages and are ready for activation.

Activate and commit install packages

Use this procedure to prepare, activate, verify, and commit Cisco NCS 1001 packages after they are present in the install repository.

Activate new software packages on NCS 1001 and commit the active software state to ensure system updates and stability.

Follow these steps to activate and commit install packages:

Procedure

1. Run one of these commands for the pre-activation checks and load individual components of the installable files onto the router setup.
   - install prepare
   - install prepare id

```
RP/0/RP0/CPU0:ios#install prepare ncs1001-mini-x-7.10.1.iso
ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
or
RP/0/RP0/CPU0:ios#install prepare id 5
Fri Jul 26 11:36:41.163 CEST
Jul 26 11:36:45 Install operation 5 started by root:
install prepare pkg ncs1001-mini-x-7.10.1.iso pkg
ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 26 11:36:46 Package list:
Jul 26 11:36:46 ncs1001-mini-x-7.10.1.iso
Jul 26 11:36:46 ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 26 11:36:53 Install operation will continue in the background
Could not start this install operation. Install operation 5 is still in
progress
RP/0/RP0/CPU0:152#Jul 26 11:48:57 Install operation 5 finished successfully
```

   The prepare process takes place. This operation is performed in asynchronous mode. The **install prepare** command runs in the background, and the EXEC prompt is returned.**Install prepare** is the first step of the activation process. If you use the operation ID, all packages that were added in the specified operation are prepared together. For example, if 5 packages are added in operation 5, by executing the **install prepare id 5** command, all 5 packages are prepared together. You do not have to prepare the packages individually.

2. Run one of these commands for the package activation. **install activate** *package_name*
   - Install activate `package_name`
   - Install activate id `operation_id`

Example

```
RP/0/RP0/CPU0:ios#install activate ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
ncs1001-mini-x-7.10.1.iso
Thu Jul 25 14:53:27.564 CEST

Jul 25 14:53:30 Install operation 14 started by root:
  install activate pkg ncs1001-k9sec-1.0.0.0-r7101 ncs1001-mini-x-7.10.1
Jul 25 14:53:30 Package list:
Jul 25 14:53:30     ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 14:53:31     ncs1001-mini-x-7.10.1.iso

This install operation will reload the system, continue?
 [yes/no]:[yes]   yes

Install operation will continue in the background
```

```
RP/0/RP0/CPU0:ios#show install log 14
Thu Jul 25 15:11:49.780 CEST
Jul 25 14:53:30 Install operation 14 started by root:
  install activate pkg ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
ncs1001-mini-x-7.10.1.iso
Jul 25 14:53:30 Package list:
Jul 25 14:53:30     ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 14:53:31     ncs1001-mini-x-7.10.1.iso
Jul 25 14:53:37 Action 1: install prepare action started
Jul 25 14:55:17 The prepared software is set to be activated with reload
upgrade
Jul 25 14:55:18 This install operation will reload the system, continue?
 [yes/no]:[yes] yes
Jul 25 14:55:18 Install operation will continue in the background
Jul 25 14:55:18 Start preparing new VM for reload upgrade
Jul 25 15:06:17 All the above nodes completed System Upgrade prepare.
Jul 25 15:06:18 Action 1: install prepare action completed successfully
Jul 25 15:06:19 Action 2: install activate action started
Jul 25 15:06:19 The software will be activated with reload upgrade
Jul 25 15:06:22 Following nodes are available for System Upgrade activate:
Jul 25 15:06:22  0/RP0
Jul 25 15:11:09 Action 2: install activate action completed successfully
Jul 25 15:11:10 Action 2: install activate action completed successfully
Jul 25 15:11:20 Install operation 14 finished successfully
Jul 25 15:11:21 Ending operation 14
```

The package configurations are made active on the NCS 1001. As a result, new features and software fixes take effect. This operation is performed in asynchronous mode. The **install activate** command runs in the background, and the EXEC prompt is returned.

Note

After an RPM of a higher version is activated, and if it is required to activate an RPM of a lower version, use the force option. For example:

Using the traditional method, add the RPM with lower version to the repository and then force the activation:

```
install add source repository ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

```
install activate ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm force
```

or

Using the install update command:

```
install update source repository ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

If you use the operation ID, all packages that were added in the specified operation are activated together. For example, if 5 packages are added in operation 2, all the 5 packages are activated together. You do not have to activate the packages individually.

3. Run the **show install active** to display active packages. You can verify from the result that the same image and package versions are active on all RPs and LCs

Example

```
RP/0/RP0/CPU0:ios#show install active
Thu Jul 25 17:04:47.600 CEST
Node 0/RP0/CPU0 [RP]
  Boot Partition: xr_lv48
  Active Packages: 2
        ncs1001-xr-7.10.1 version=7.10.1 [Boot image]
        ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

Review the command output for operation status, package names, and completion messages.

Displays packages that are active.

4. Run the **install commit** to commit the Host, XR, and System Admin active software.

Example

```
RP/0/RP0/CPU0:ios#install commit system
Thu Jul 25 17:05:27.364 CEST
Jul 25 17:05:30 Install operation 15 started by root:
  install commit system
Jul 25 17:05:31 Install operation will continue in the background

Jul 25 17:05:55 Install operation 15 finished successfully
```

```
install commit system
```

> **Note**
>
> - If you perform a manual or automatic system reload without completing the transaction with the install commit command during system upgrade, the action will revert the system to the point before the install transaction commenced, including any configuration changes. Only the log is preserved for debugging. This action clears all configuration rollback points available. You will not be able to rollback to, or view, any commits made until the install rollback event. Any new commits made after the install rollback event will start from commit ID '1000000001'.
> - On Multi-SDR mode, you can use the **install commit sdr** to commit just the sdr from where the CLI is being triggered.

5. Run the **show install commit** command to display the committed packages.

   Example

```
RP/0/RP0/CPU0:ios#show install commit
Thu Jul 25 17:07:54.255 CEST
Node 0/RP0/CPU0 [RP]
  Boot Partition: xr_lv48
  Committed Packages: 2
        ncs1001-xr-7.10.1 version=7.10.1[Boot image]
        ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

   Review the command output for operation status, package names, and completion messages.

The selected package versions are active and fully committed on the NCS 1001.

What to do next

- After performing a system upgrade, upgrade FPD by using the **upgrade hw-module location all fpd all** command from the Cisco IOS XR mode. The progress of FPD upgrade process can be monitored using the **show hw-module fpd** command.
- Reload NCS 1001 if any FPD status is in RLOAD REQ state. If CTRL FPGA is in RLOAD REQ state, use the **hw-module location 0/0 reload** command. If Daisy Duke or BIOS is in RLOAD REQ state, use the **hw-module location 0/RP0 reload** command.
- Verify the installation using the **install verify packages** command.
- Uninstall the packages or SMUs if their installation causes any issues on the NCS 1001. See uninstall packages.

> **Note**
>
> ISO images cannot be uninstalled. However, you can perform a system downgrade by installing an older ISO version.

Uninstall packages

Use this procedure to uninstall NCS 1001 packages or software maintenance updates when an installed package causes an issue.

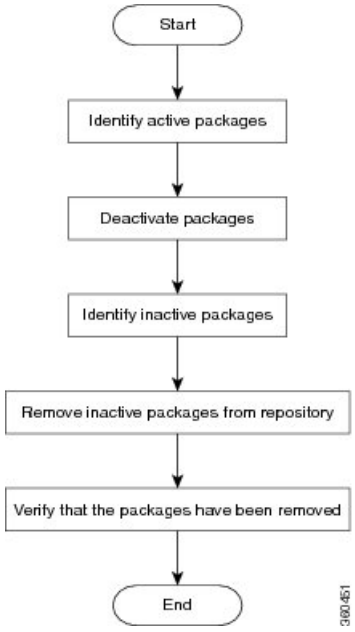
Deactivate inactive software packages from the running system and remove package files from the repository.

Complete this task to uninstall a package. All the NCS 1001 functionalities that are part of the uninstalled package are deactivated. Packages that are added in the XR mode cannot be uninstalled from the System Admin mode, and vice versa.

> **Note**
>
> Installed ISO images cannot be uninstalled. Also, kernel SMUs that install third party SMU on host, XR mode and System Admin mode, cannot be uninstalled. However, subsequent installation of ISO image or kernel SMU overwrites the existing installation.

The workflow for uninstalling a package is shown in this flowchart.

Figure 4: Uninstalling Packages Workflow



Procedure

1. Run the **show install active** command to displays active packages.

Example

```
RP/0/RP0/CPU0:ios#show install active
Thu Jul 25 16:23:36.579 CEST
Node 0/RP0/CPU0 [RP]
  Boot Partition: xr_lv48
  Active Packages: 2
        ncs1001-xr-7.10.1 version=7.10.1 [Boot image]
        ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

Only active packages can be deactivated.

2. Run the **show install repository** command.

Example

```
show install repository
```

```
Thu Jul 25 16:52:03.432 CEST
2 package(s) in XR repository:
    ncs1001-mini-x-7.10.1.iso
    ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

3. Run one of these commands:
   - **install deactivate** *package_name*
   - **install deactivate id** *operation_id*

Example

```
RP/0/RP0/CPU0:ios#install deactivate ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

or

```
RP/0/RP0/CPU0:ios#install deactivate id 48
```

All features and software patches associated with the package are deactivated. You can specify multiple package names and deactivate them simultaneously.

If you use the operation ID, all packages that were added in the specified operation are deactivated together. You do not have to deactivate the packages individually.

```
Thu Jul 25 16:23:52.789 CEST
Jul 25 16:23:56 Install operation 48 started by root:
  install deactivate pkg ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 16:23:56 Package list:
Jul 25 16:23:56     ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 16:24:11 Install operation will continue in the background
Jul 25 16:26:38 Install operation 48 finished successfully
```

4. Run the **show install inactive** command.

Example

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

```
Thu Jul 25 16:27:54.005 CEST
1 inactive package(s) found:
    ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

The deactivated packages are now listed as inactive packages. Only inactive packages can be removed from the repository.

5. Run the **install remove***package_name* command to remove the inactive packages from the repository.

Example

```
RP/0/RP0/CPU0:ios#install remove ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
```

```
Thu Jul 25 16:30:11.870 CEST
Jul 25 16:30:14 Install operation 49 started by root:
  install remove ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 16:30:14 Package list:
Jul 25 16:30:15     ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
Jul 25 16:30:16 Install operation will continue in the background
Jul 25 16:30:21 Install operation 49 finished successfully
```

The inactive packages are removed from the repository.

Use the **install remove** command with the **id** *operation-id* keyword and argument to remove all packages that were added for the specified operation ID.

6. Run the **show install repository** command to display the packages available in the repository.

Example

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

```
Thu Jul 25 16:52:03.432 CEST
.. package(s) in XR repository:
    ncs1001-mini-x-7.10.1.iso
```

The package that are removed are no longer displayed in the result.

The specified packages are deactivated, removed from the repository, and no longer listed in the install repository.

What to do next

Install required packages. See install packages.