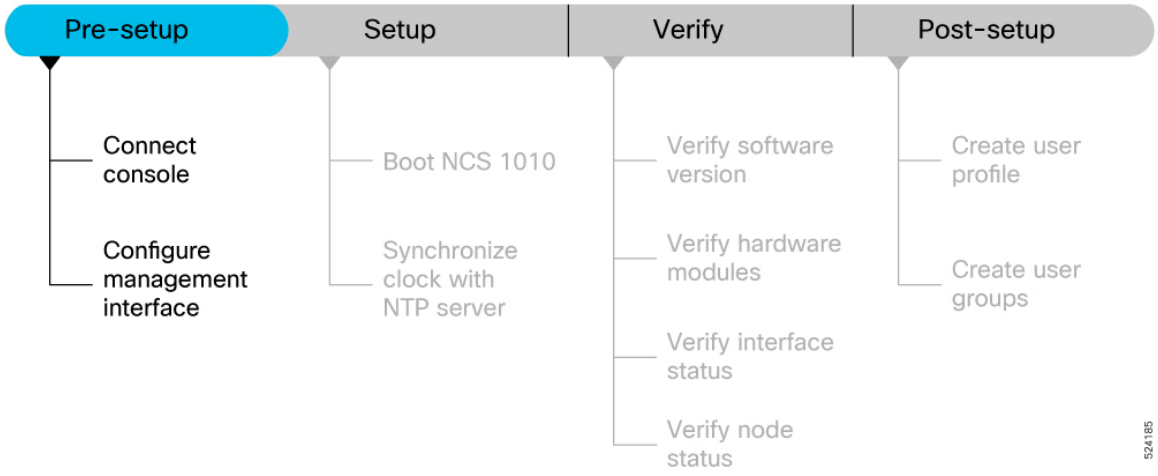
Setup Procedures

Pre-setup requirements for Cisco NCS 1010

Complete these prerequisite tasks to prepare the NCS 1010 for seamless setup.

Figure 1: Pre-setup Workflow for the Cisco NCS 1010



Connect the console port to a terminal

Connect your terminal or PC to the Cisco NCS 1010 console port and configure terminal settings for direct device access.

The console port allows you to log into the NCS 1010 without a network connection, using an emulation program such as HyperTerminal.

Procedure

Step 1 Connect the console (or rollover) cable to the console port on the NCS 1010.

Step 2 Use the correct adapter to connect the other end of the cable to your terminal or PC.

Step 3 Launch the terminal session.

Step 4 In the **COM1 Properties** window, select **Port Settings** tab, and enter these settings:

Setting	Value
Speed	9600
Data Bits	8
Parity	None
Stop bits	1
Flow Control	None

Step 5 Click **OK**.

You should see a blinking cursor in the HyperTerminal window indicating successful connection to the console port.

The terminal or PC is connected to the console port, and the terminal session is ready for initial access.

Configure the management interface

Use this procedure to configure the management interface.

The management interface can be used for system management and remote communication. To use the management interface for system management, you must configure an IP address and subnet mask. To use the management interface for remote communication, you must configure a static route. Use this procedure when NCS 1010 chassis is not booted using ZTP.

Before you begin

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

Procedure

Step 1 Enter configuration mode.

configure

Example:

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

Enters IOS XR configuration mode.

Step 2 Enter management interface configuration mode.

interface mgmtEth 0/RP0/CPU0/0

Example:

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

Enters interface configuration mode for the management interface.

Step 3 Configure the IPv4 address and subnet mask.

ipv4 address 192.0.2.254 255.255.255.0

Example:

```
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 192.0.2.254 255.255.255.0
```

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

Step 4 Enable the interface.

no shutdown

Example:

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

Places the management interface in an "up" state.

Step 5 Exit interface configuration mode.

exit

Example:

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

Exits the management interface configuration mode.

Step 6 Configure a static route.

ncs1010 static address-family ipv4 unicast 0.0.0.0/0 198.51.100.4

Example:

```
RP/0/RP0/CPU0:ios(config)#ncs1010 static address-family ipv4 unicast 0.0.0.0/0 198.51.100.4
```

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

Step 7 Save or exit the configuration session.

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

end-Prompts user to take one of these actions:

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

The configure the management interface task is complete.

What to do next

Connect the management interface to the Ethernet network. Establish a Configure SSH or Configure Telnet connection to the management interface using its IP address.

LLDP support on the management interface

The Link Layer Discovery Protocol (LLDP) support on management interface feature requires a system to form LLDP neighbor relationship over the system management interface, through which it advertises and learns LLDP neighbor information. This information about neighbors used to learn about the neighbors and in turn the topology of the devices for Operations, Administration, and Maintenance (OAM) purposes.

Advantages of LLDP

- Provides support on non-Cisco devices.
- Enables neighbor discovery between non-Cisco devices.

Limitation

- When you disable LLDP globally, the LLDP gets disabled on all the interfaces.

Note By default, LLDP is enabled for NCS 1010. But when you enable and disable LLDP in the global configuration mode, LLDP gets disabled on all the interfaces.

Workaround: You must enable LLDP globally or reload the NCS1010.

Cisco Discovery Protocol (CDP) vs LLDP

The CDP is a device discovery protocol that runs over Layer 2. Layer 2 is also known as the data link layer that runs on all Cisco devices, such as routers, bridges, access servers, and switches. This protocol allows the network management applications to automatically discover and learn about other Cisco devices that connect to the network.

The LLDP is also a device discovery protocol that runs over Layer 2. This protocol allows the network management applications to automatically discover and learn about other non-Cisco devices that connect to the network.

Interoperability between non-Cisco devices using LLDP

LLDP is also a neighbor discovery protocol that is used by network devices to advertise information about themselves to other devices on the network. This protocol runs over the data link layer, which allows two systems running different network layer protocols to learn about each other.

With LLDP, you can also access the information about a particular physical network connection. If you use a non-Cisco monitoring tool (through SNMP), LLDP helps you identify the Object Identifiers (OIDs) that the system supports. These OIDs are supported:

- 1.0.8802.1.1.2.1.4.1.1.4
- 1.0.8802.1.1.2.1.4.1.1.5
- 1.0.8802.1.1.2.1.4.1.1.6

- 1.0.8802.1.1.2.1.4.1.1.7
- 1.0.8802.1.1.2.1.4.1.1.8
- 1.0.8802.1.1.2.1.4.1.1.9
- 1.0.8802.1.1.2.1.4.1.1.10
- 1.0.8802.1.1.2.1.4.1.1.11
- 1.0.8802.1.1.2.1.4.1.1.12

Neighbor Discovery

System advertises the LLDP TLV (Type Length Value) details over the management network using which other devices in the management network can learn about this device.

Configuring LLDP

- LLDP full stack functionality is supported on all three management interfaces that are supported in NCS 1010.
- You can selectively enable or disable LLDP on any of the management interfaces on demand.
- You can selectively enable or disable LLDP transmit or receive functionality at the management interface level.
- Information gathered using LLDP can be stored in the device Management Information Database (MIB) and queried with the Simple Network Management protocol (SNMP).
- LLDP operational data is available in both CLI and netconf-yang interface.

Enabling LLDP Globally

When you enable LLDP globally, all interfaces that support LLDP are automatically enabled for both transmit and receive operations.

Note You can override this default operation at the interface to disable receive or transmit operations.

This table describes the global LLDP attributes that you can configure:

Table 1:

Attribute	Default	Range	Description
Holdtime	120	0-65535	Specifies the holdtime (in sec). Holdtime refers to the time or duration that an LLDP device maintains the neighbor information before discarding.
Reinit	2	2-5	Delay (in sec) for LLDP initialization on any interface
Timer	30	5-65534	Specifies the rate at which LLDP packets are sent (in sec)

Configure LLDP globally

Enable LLDP globally on all three management interfaces.

When you enable LLDP globally, all interfaces that support LLDP are automatically enabled for both transmit and receive operations.

Note You can override this default operation at the interface to disable receive or transmit operations.

Procedure

Step 1 Enter global configuration mode.

configure terminal

Example:

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

Step 2 Enable LLDP on the management interfaces.

lldp management enable

Example:

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

Step 3 Set the LLDP holdtime.

lldp holdtime 30

Example:

```
RP/0/RP0/CPU0:ios(config)#lldp holdtime 30
```

Step 4 Set the LLDP reinitialization delay.

lldp reinit 2

Example:

```
RP/0/RP0/CPU0:ios(config)#lldp reinit 2
```

Step 5 Commit the configuration.

commit

Example:

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

The global LLDP configuration enables LLDP on all the three management interfaces.

Verify LLDP configuration and data

Verify the LLDP configuration and operational data.

Procedure

Step 1 Verify the LLDP running configuration.

show running-config lldp

Example:

```
RP/0/RP0/CPU0:ios#show running-config lldp
Tue Dec 10 10:36:11.567 UTC
lldp
timer 30
reinit 2
holdtime 120
management enable
!
```

Step 2 Verify the LLDP interface data.

show lldp interface

Example:

```
RP/0/RP0/CPU0:ios#show lldp interface
Mon Nov 11 14:33:58.982 IST


MgmtEth0/RP0/CPU0/0:
        Tx: enabled
        Rx: enabled
        Tx state: IDLE
        Rx state: WAIT FOR FRAME


MgmtEth0/RP0/CPU0/2:
        Tx: enabled
        Rx: enabled
        Tx state: IDLE
        Rx state: WAIT FOR FRAME


GigabitEthernet0/0/0/0:
        Tx: enabled
        Rx: enabled
        Tx state: IDLE
        Rx state: WAIT FOR FRAME
```

Step 3 Verify the LLDP neighbor data.

show lldp neighbors

Example:

```
RP/0/RP0/CPU0ios:M-131#show lldp neighbors
Mon Dec 9 14:57:55.915 IST
Capability codes:
(R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
(W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

Device ID Local Intf Hold-time Capability Port ID
P1C_DT_01.cisco.com GigabitEthernet0/0/0/0 120 R GigabitEthernet0/0/0/0
NCS1004-HH-10 MgmtEth0/RP0/CPU0/2 60 R MgmtEth0/RP0/CPU0/2
```

```
Total entries displayed: 2
```

where [DISABLED] shows that the LLDP is disabled on the interface MgmtEth0/RP0/CPU0/0.

Note
If the RCOM interface is enabled, the output of **show lldp neighbors** command would include the entries for both LLDP neighbours and remote connect neighbours.

The LLDP running configuration, interface data, and neighbor data are verified.

Configure LLDP on a management interface

Enable LLDP at the management interface level.

Procedure

Step 1 Enter the management interface configuration mode.

interface mgmtEth 0/RP0/CPU0/X

Example:

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

Step 2 Enable LLDP on the management interface.

lldp enable

Example:

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

Step 3 Commit the configuration.

commit

Example:

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

LLDP is configured at the management interface level.

Disable LLDP transmit and receive operations

Disable LLDP transmit operations, receive operations, or both at a specified management interface.

Procedure

Step 1 Enter the management interface configuration mode.

interface mgmtEth 0/RP0/CPU0/X

Example:

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

Step 2 Disable LLDP transmit operations at the specified management interface.

lldp transmit disable

Example:

```
RP/0/RP0/CPU0:ios(config-if)#lldp transmit disable
```

Step 3 Disable LLDP receive operations at the specified management interface.

lldp receive disable

Example:

```
RP/0/RP0/CPU0:ios(config-if)#lldp receive disable
```

Step 4 Commit the configuration.

commit

Example:

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

LLDP transmit or receive operations are disabled at the specified management interface.

LLDP traffic and debug commands

Use these commands for debugging issues in the LLDP functionality.

Table 2: LLDP troubleshooting commands

Command	Description
show lldp traffic	Displays statistics for LLDP traffic.
debug lldp all	Enables all LLDP debugging information.
debug lldp errors	Enables debugging information for LLDP errors.
debug lldp events	Enables debugging information for LLDP events.
debug lldp packets	Enables debugging information for LLDP packets.
debug lldp tlvs	Enables debugging information for LLDP TLVs.
debug lldp trace	Enables LLDP trace debugging information.
debug lldp verbose	Enables verbose LLDP debugging information.

Configure Telnet

Use this procedure to configure Telnet.

This procedure allows you to establish a telnet session to the management interface using its IP address. Use this procedure when NCS 1010 chassis is not booted using ZTP.

Before you begin

Ensure that two xr-telnet-* rpms are installed. .

Procedure

Step 1 Enter configuration mode.

configure

Example:

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

Enters the configuration mode.

Step 2 Specify the number of allowable Telnet servers.

telnet ipv4 server max-servers 10

Example:

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

Specifies the number of allowable telnet servers (up to 100). By default, telnet servers are not allowed. You must configure this command to enable the use of telnet servers.

Step 3 Save or exit the configuration session.

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

end-Prompts user to take one of these actions:

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

The configure Telnet task is complete.

Configure SSH

Use this procedure to configure SSH.

This procedure allows you to establish an SSH session to the management interface using its IP address. Use this procedure when NCS 1010 chassis is not booted using ZTP.

Before you begin

- Generate the crypto key for SSH using the **crypto key generate dsa** command.

Procedure

Step 1 Enter configuration mode.

configure

Example:

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

Enters the configuration mode.

Step 2 Enable SSH server version 2.

ssh server v2

Example:

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

Enables the SSH server to accept only SSHv2 client connections.

Step 3 Save or exit the configuration session.

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

end-Prompts the user to take one of these actions:

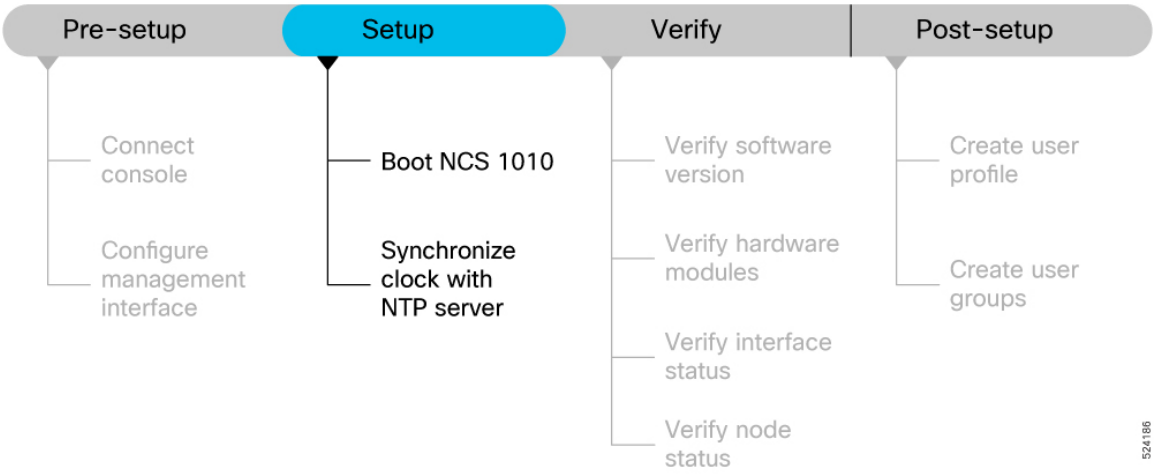
- **Yes**-Saves configuration changes and exits the configuration session.
- **No**-Exits the configuration session without committing the configuration changes.
- **Cancel**-Remains in the configuration session without committing the configuration changes.

The configure SSH task is complete.

Cisco NCS 1010 setup workflow

Complete these tasks to bring up your NCS 1010 for further configuration.

Figure 2: Setup workflow for Cisco NCS 1010



Boot Cisco NCS 1010

Use this procedure to boot Cisco NCS 1010.

Use the console port to connect to NCS 1010. By default, the console port connects to the XR mode. If necessary, you can establish subsequent connections through the management port, after it is configured.

Procedure

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

Step 2 Start the terminal emulation program on your workstation.

The console settings are 9600 bps, 8 data bits, 1 stop bit and no parity.

Step 3 Power on NCS 1010.

To power on the shelves, install the AC or DC power supplies and cables. As NCS 1010 boots up, you can view the boot process details at the console of the terminal emulation program.

Step 4 Press **Enter**.

The boot process is complete when the system prompts you to enter the root-system username. If the prompt does not appear, wait for a while to give NCS 1010 more time to complete the initial boot procedure; then press **Enter**.

Note
If the boot process fails, it may be because the preinstalled image on the NCS 1010 is corrupt. In this case, you can boot NCS 1010 using an external bootable USB drive.

The boot Cisco NCS 1010 task is complete.

Boot Cisco NCS 1010 using a USB drive

Use this supertask to prepare a bootable USB drive and boot Cisco NCS 1010 from that drive.

Before you begin

- You need a USB drive with a storage capacity of at least 4 GB.
- The USB drive should have a single partition.
- NCS 1010 software image can be downloaded from Software Download page on Cisco.com.
- Copy the compressed boot file from the software download page at Cisco.com to your local machine. The filename for the compressed boot file is in the format *ncs1010-usb-boot-<release_number>.zip*.

The bootable USB drive is used to reimage NCS 1010 for system upgrade or to boot the NCS 1010 in case of boot failure. A bootable USB drive is created by copying a compressed boot file into a USB drive. The USB drive becomes bootable after the contents of the compressed file are extracted.

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

Use this task to boot the NCS 1010 using the USB drive.

Procedure

Step 1 Prepare a USB boot drive.

See Prepare a USB boot drive.

Step 2 Start Cisco NCS 1010 from a USB drive.

See Start Cisco NCS 1010 from a USB drive.

Cisco NCS 1010 boots from the USB image and reboots after installation.

Prepare a USB boot drive

Use this procedure to format the USB drive, copy the compressed boot file, verify the file, and extract the contents at the root of the drive.

Before you begin

- You need a USB drive with a storage capacity of at least 4 GB.
- The USB drive should have a single partition.
- NCS 1010 software image can be downloaded from Software Download page on Cisco.com.
- Copy the compressed boot file from the software download page at Cisco.com to your local machine. The filename for the compressed boot file is in the format *ncs1010-usb-boot-<release_number>.zip*.

The prepared USB drive contains the extracted boot files that make the drive bootable.

Procedure

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

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

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

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

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

The USB drive is ready to boot Cisco NCS 1010.

Start Cisco NCS 1010 from a USB drive

Use this procedure to insert the prepared USB drive, select the BIOS boot option, and remove the drive after the image loads.

Before you begin

Prepare the USB boot drive before you start Cisco NCS 1010 from the drive. See Prepare a USB boot drive.

Use the console or BIOS boot option when you need to boot Cisco NCS 1010 from the prepared USB drive.

Procedure

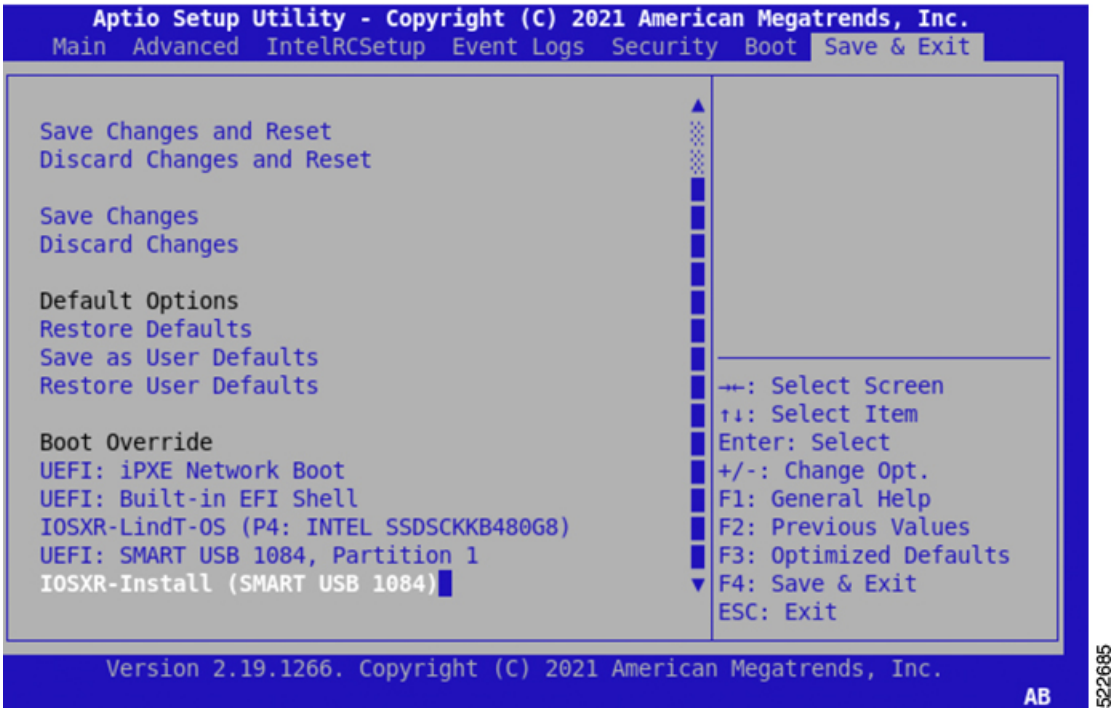
Step 1 Insert the USB drive in one of the USB ports of NCS 1010 line card/controller card.

Step 2 Reboot NCS 1010 using power cycle or console.

Note
Use the **reload bootmedia usb noprompt** command to boot the NCS 1010 from the USB. If you are using the **reload bootmedia usb noprompt** command, then you can skip the remaining steps.

Step 3 Press **Esc** to enter BIOS.

Step 4 Select the **Save & Exit** tab of BIOS.



Step 5 Choose **IOS -XR Install**.

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

The system detects USB and boots the image from USB.

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

Step 6 Remove the USB drive after the `Rebooting the system after installation` message is displayed. The NCS 1010 reboots automatically.

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

Cisco NCS 1010 starts from the USB drive and reboots after installation.

Boot Cisco NCS 1010 using iPXE

Use iPXE boot to reimage Cisco NCS 1010 through a network boot workflow.

Use iPXE boot when the router fails to boot or when no valid bootable partition is available. iPXE enables network boot for an offline router. The iPXE bootloader downloads and installs the ISO image located on an HTTP, FTP, or TFTP server and reimages the router.

iPXE boot can be invoked through the CLI terminal or through the BIOS interface.

Before you begin

- Ensure that the DHCP server is set and running. For details, see Configure a DHCP server for iPXE boot.
- Ensure that the management port of the NCS 1010 chassis is in *UP* state.

Procedure

Step 1 Invoke iPXE boot through the CLI terminal.

For details, see Boot Cisco NCS 1010 using iPXE from the CLI.

Step 2 Invoke iPXE boot through the BIOS interface.

For details, see Boot Cisco NCS 1010 using iPXE from BIOS.

The iPXE boot process downloads the ISO image and reimages the Cisco NCS 1010 chassis.

Configure a DHCP server for iPXE boot

Configure a DHCP server to provide Cisco NCS 1010 iPXE boot information.

A DHCP server must be configured for IPv4, IPv6, or both communication protocols before Cisco NCS 1010 can use iPXE boot.

For DHCPv6, send a routing advertisement (RA) message to all nodes in the network to indicate the method used to obtain the IPv6 address.

Procedure

Step 1 If you use DHCPv6, configure Router Advertisement Daemon to allow the client to send the DHCP request.

Example:

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

Step 2 Create the dhcpd.conf file, dhcpv6.conf file, or both files in the /etc/ directory.

The configuration file stores network information, such as the script path, ISO install file location, provisioning configuration file location, serial number, and chassis MAC address.

Step 3 Add a host entry that uses the chassis MAC address.

Example:

```
host ncs1010
{
hardware ethernet ab:cd:ef:01:23:45;
fixed-address <ip address>;
filename "http://<httpserver-address>/<path-to-image>/ncs1010-mini-x.iso";
}
```

Ensure that the DHCP host configuration is successful after the DHCP server is running.

Step 4 If you identify the chassis by serial number, add a host entry that uses the chassis serial number.

Example:

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

Example:

```
host 10.89.205.202 {
       hardware ethernet 40:55:39:56:0c:e8;
        option dhcp-client-identifier "<FCB2437B066>";
          if exists user-class and option user-class = "iPXE" {
          filename "http://10.89.205.127/box1/ncs1010-x64.iso";
       } else {
           filename "http://10.89.205.127/box1/StartupConfig.cfg";
       }
       fixed-address 10.89.205.202;
    }
```

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

The DHCP server provides the iPXE boot image or provisioning configuration file information to Cisco NCS 1010.

Boot Cisco NCS 1010 using iPXE from the CLI

Invoke iPXE boot from the CLI terminal to reimage the chassis.

Use this method to start the iPXE boot process from the CLI terminal.

Before you begin

- Ensure that the DHCP server is set and running.
- Ensure that the management port of the NCS 1010 chassis is in *UP* state.

Procedure

Step 1 Run the command to invoke the iPXE boot process and reimage the chassis.

reload bootmedia network location all

Example:

```
RP/0/RP0/CPU0:ios# reload bootmedia network location all
Wed Jul  6 15:11:33.791 UTC
Reload hardware module ? [confirm]
```

Step 2 Review the iPXE boot output.

Example:

```
Preparing system for backup. This may take a few minutes especially for large configurations.
        Status report: node0_RP0_CPU0: BACKUP INPROGRESS
RP/0/RP0/CPU0:P1D_DT#   Status report: node0_RP0_CPU0: BACKUP HAS COMPLETED SUCCESSFULLY
[Done]
[FAILED] Failed unmounting /mnt/fuse/parser_server.
[  OK  ] Unmounted /mnt/fuse/ftp.
[  OK  ] Unmounted /mnt/fuse/nvgen_server.
[  OK  ] Unmounted /boot/efi.
[  OK  ] Unmounted /selinux.
.
.
Output Snipped
.
.
..         *** Sirius ***
System Initializing..
..

ERROR: Class:0; Subclass:10000; Operation: 1004

Shelf Assembly Reset
Shelf Assembly Reset for P1


..         *** Sirius ***
System Initializing..
..

ERROR: Class:0; Subclass:10000; Operation: 1004
.
.
Output Snipped
.
.

NCS1010, Initializing Devices

Booting from Primary Flash
Aldrin: Programmed MI 10
.
.
Output Snipped
.
.
Version 2.19.1266. Copyright (C) 2022 American Megatrends, Inc.
BIOS Date: 05/20/2022 10:47:39 Ver: 0ACHI0410
Press <DEL> or <ESC> to enter setup.
TAM Chipguard Validate Observed DB Error: 0x48

WARNING!!! TAM: Empty Chip DB


Software Boot OK, Validated

iPXE initialising devices...ok
```

```
iPXE 1.0.0+ (c2215) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP TFTP VLAN EFI ISO9660 ISO9660_grub Menu
Trying net0-2051,net0-2052 and net0-2053...
net0-2051: 68:9e:0b:b8:71:1e using NII on NII-PCI06:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
  [Link status: Unknown (http://ipxe.org/1a086194)]
Configuring (net0-2051 68:9e:0b:b8:71:1e)................... Error 0x040ee186 (http://ipxe.org/040ee186)
net0-2052: 68:9e:0b:b8:71:1f using NII on NII-PCI06:00.0 (open)
  [Link:up, TX:0 TXE:0 RX:18 RXE:14]
  [RXE: 8 x "Operation not supported (http://ipxe.org/3c086083)"]
  [RXE: 3 x "Error 0x440e6083 (http://ipxe.org/440e6083)"]
  [RXE: 3 x "The socket is not connected (http://ipxe.org/380f6093)"]
Configuring (net0-2052 68:9e:0b:b8:71:1f)................... ok
net0: fe80::6a9e:bff:feb8:711e/64
net1: fe80::6a9e:bff:feb8:7121/64 (inaccessible)
net2: fe80::6a9e:bff:feb8:7122/64 (inaccessible)
net3: fe80::6a9e:bff:feb8:7123/64 (inaccessible)
net0-2051: fe80::6a9e:bff:feb8:711e/64
net0-2051: 2001:420:5446:2014::281:0/119 gw fe80::676:b0ff:fed8:c100 (no address)
net0-2051: 2002:420:54ff:93:6a9e:bff:feb8:711e/64 gw fe80::fa4f:57ff:fe72:a640
net0-2052: 10.4.33.44/255.255.0.0 gw 10.4.33.1
net0-2052: fe80::6a9e:bff:feb8:711e/64
net0-2053: fe80::6a9e:bff:feb8:711e/64
Filename: http://10.4.33.51/P1D_DT_05/ncs1010-x64.iso
http://10.4.33.51/P1D_DT_05/ncs1010-x64.iso... ok
.
.
Output Snipped
.
.
User Access Verification

Username: cisco
Password:
```

The iPXE boot process downloads the ISO image and displays the user access verification prompt.

Cisco NCS 1010 boots through iPXE from the CLI terminal and starts the reimage workflow.

Boot Cisco NCS 1010 using iPXE from BIOS

Invoke iPXE boot from the BIOS interface to reimage the chassis.

Use this method to start the iPXE boot process from the BIOS interface.

Before you begin

- Ensure that the DHCP server is set and running.
- Ensure that the management port of the NCS 1010 chassis is in *UP* state.

Procedure

Step 1 Reboot NCS 1010 using power cycle or console.

Step 2 Press **Esc** to enter BIOS.

Step 3 Select the **Save & Exit** tab of BIOS.

Step 4 Choose **UEFI: iPXE Network Boot**.

Example:

```
Preparing system for backup. This may take a few minutes especially for large configurations.
        Status report: node0_RP0_CPU0: BACKUP INPROGRESS
RP/0/RP0/CPU0:P1D_DT#   Status report: node0_RP0_CPU0: BACKUP HAS COMPLETED SUCCESSFULLY
[Done]
[FAILED] Failed unmounting /mnt/fuse/parser_server.
[  OK  ] Unmounted /mnt/fuse/ftp.
[  OK  ] Unmounted /mnt/fuse/nvgen_server.
[  OK  ] Unmounted /boot/efi.
[  OK  ] Unmounted /selinux.
.
.
Output Snipped
.
.
..         *** Sirius ***
System Initializing..
..

ERROR: Class:0; Subclass:10000; Operation: 1004

Shelf Assembly Reset
Shelf Assembly Reset for P1


..         *** Sirius ***
System Initializing..
..

ERROR: Class:0; Subclass:10000; Operation: 1004
.
.
Output Snipped
.
.

NCS1010, Initializing Devices

Booting from Primary Flash
Aldrin: Programmed MI 10
.
.
Output Snipped
.
.
Version 2.19.1266. Copyright (C) 2022 American Megatrends, Inc.
BIOS Date: 05/20/2022 10:47:39 Ver: 0ACHI0410
Press <DEL> or <ESC> to enter setup.
TAM Chipguard Validate Observed DB Error: 0x48

WARNING!!! TAM: Empty Chip DB


Software Boot OK, Validated

iPXE initialising devices...ok


iPXE 1.0.0+ (c2215) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP TFTP VLAN EFI ISO9660 ISO9660_grub Menu
```

```
Trying net0-2051,net0-2052 and net0-2053...
net0-2051: 68:9e:0b:b8:71:1e using NII on NII-PCI06:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
  [Link status: Unknown (http://ipxe.org/1a086194)]
Configuring (net0-2051 68:9e:0b:b8:71:1e).................. Error 0x040ee186 (http://ipxe.org/040ee186)
net0-2052: 68:9e:0b:b8:71:1f using NII on NII-PCI06:00.0 (open)
  [Link:up, TX:0 TXE:0 RX:18 RXE:14]
  [RXE: 8 x "Operation not supported (http://ipxe.org/3c086083)"]
  [RXE: 3 x "Error 0x440e6083 (http://ipxe.org/440e6083)"]
  [RXE: 3 x "The socket is not connected (http://ipxe.org/380f6093)"]
Configuring (net0-2052 68:9e:0b:b8:71:1f).................. ok
net0: fe80::6a9e:bff:feb8:711e/64
net1: fe80::6a9e:bff:feb8:7121/64 (inaccessible)
net2: fe80::6a9e:bff:feb8:7122/64 (inaccessible)
net3: fe80::6a9e:bff:feb8:7123/64 (inaccessible)
net0-2051: fe80::6a9e:bff:feb8:711e/64
net0-2051: 2001:420:5446:2014::281:0/119 gw fe80::676:b0ff:fed8:c100 (no address)
net0-2051: 2002:420:54ff:93:6a9e:bff:feb8:711e/64 gw fe80::fa4f:57ff:fe72:a640
net0-2052: 10.4.33.44/255.255.0.0 gw 10.4.33.1
net0-2052: fe80::6a9e:bff:feb8:711e/64
net0-2053: fe80::6a9e:bff:feb8:711e/64
Filename: http://10.4.33.51/P1D_DT_05/ncs1010-x64.iso
http://10.4.33.51/P1D_DT_05/ncs1010-x64.iso... ok
.
.
Output Snipped
.
.
User Access Verification

Username: cisco
Password:
```

The iPXE boot process downloads the ISO image and displays the user access verification prompt.

Cisco NCS 1010 boots through iPXE from the BIOS interface and starts the reimage workflow.

Install a new image without Golden ISO

Install a new image without using the Golden ISO feature.

Before the introduction of Golden ISO, you had to perform this sequence to install a new image.

Before you begin

- Ensure that the mini ISO is available.
- Ensure that all relevant SMUs, optional packages, and IOS XR configuration are available.

Procedure

Step 1 Boot the system with mini ISO.

You can use iPXE or USB boot.

Step 2 Install, add, and activate all relevant SMUs and optional packages on NCS 1010.
NCS 1010 reloads when any SMU reloads.

Step 3 Apply IOS XR configuration.

The new image is installed without using Golden ISO.

Build a Golden ISO boot image for Cisco NCS 1010

Build a customized Golden ISO image that includes the mini ISO, required SMUs, and IOS XR configuration.

Golden ISO is a feature that enables you to build a customized ISO using mini ISO, required SMUs, and IOS XR configuration.

Golden ISO saves installation effort and time. It makes the system available in a single command and boot.

The `gisobuild.py` script is available at `/pkg/bin/gisobuild.py`.

Note Install operation over IPv6 is not supported.

Before you begin

- For details about the image installation sequence used before Golden ISO was introduced, see Install a new image without Golden ISO.

  Copy the `/pkg/bin/gisobuild.py` script from NCS 1010 to the Linux environment.
- Ensure that the mini ISO, required SMUs, and IOS XR configuration file are available.

Procedure

Step 1 Build the Golden ISO image.

gisobuild.py -i *mini-iso* **-r** *rpm-directory* **-c** *xr-config* **-l** *label*

- *rpm-directory* - Directory where SMUs (xr, calvados, and host) are copied.
- *xr-config* - IOS XR configuration to be applied to the system after booting.
- *label* - Label of the Golden ISO.

Example:

```
gisobuild.py -i./ncs1010-mini-x.iso -r ./rpm-directory -c ./xr-config -l label
```

Step 2 Review the Golden ISO build output.

Example:

```
python gisobuild.py -i ./ncs1010-mini-x-7.0.1.04I.iso -r. -c startup_new.cfg -l v2
System requirements check [PASS]
Golden ISO build process starting...

Platform: ncs1010 Version: 7.0.1.04I

XR-Config file (/bh/bosshogg_images/r701/701_04I_DT_IMAGE/giso/startup_new.cfg) will be encapsulated
 in Golden ISO.
```

```
Scanning repository [/bh/bosshogg_images/r701/701_04I_DT_IMAGE/giso]...

Building RPM Database...
Total 1 RPM(s) present in the repository path provided in CLI

XR x86_64 rpm(s) used for building Golden ISO:

(+) ncs1010-k9sec-192.0.2.1-r70104I.x86_64.rpm

...RPM compatibility check [PASS]

Building Golden ISO...
Summary .....

XR rpms:
ncs1010-k9sec-192.0.2.1-r70104I.x86_64.rpm

XR Config file:
router.cfg

...Golden ISO creation SUCCESS.

Golden ISO Image Location:
/bh/bosshogg_images/r701/701_04I_DT_IMAGE/giso/ncs1010-goldenk9-x-7.0.1.04I-v2.iso

Detail logs: /bh/bosshogg_images/r701/701_04I_DT_IMAGE/giso/Giso_build.log-2019-03-20:15:47:19.516203
```

The command output shows that Golden ISO creation succeeded and displays the Golden ISO image location.

Step 3 Verify the Golden ISO file format.

Use these Golden ISO filename formats:

- *platform-name*-golden-x.iso-*version.label* does not contain the security (*k9sec*.rpm) rpm.
- *platform-name*-goldenk9-x.iso-*version.label* contains the security (*k9sec*.rpm) rpm.

Example:

Example 1: `ncs1010-golden-x-7.0.1.014I-V1.iso`

Example 2: `ncs1010-goldenk9-x-7.0.1.014I-V1.iso`

The Golden ISO boot image is built and its filename format is verified.

Network Time Protocol

A Network Time Protocol implementation is a time synchronization function that

- uses UDP and Coordinated Universal Time to synchronize device clocks,
- forms configured associations with NTP servers to exchange timing messages, and
- supports accurate event timing for network management, security, planning, and debugging.

Details

Table 3: Feature History

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

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

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

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

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

Synchronize the clock with an NTP server

Use this procedure to synchronize the clock with an NTP server.

There is an independent system clock for IOS XR. To ensure that this clock does not deviate from true time, it must be synchronized with the clock of an NTP server.

Before you begin

Configure Management Interface

Procedure

Step 1 Enter configuration mode.

configure

Example:

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

Enters the configuration mode.

Step 2 Enter NTP configuration mode.

ntp

Example:

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

Enters NTP configuration mode.

Step 3 Configure an NTP server.

server 198.51.100.1 version 4 prefer iburst

server 2001:DB8::1 version 4 prefer iburst

Example:

IPv4:

```
RP/0/RP0/CPU0:ios(config-ntp)#server 198.51.100.1 version 4 prefer iburst
```

IPv6:

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

Synchronizes the console clock with the specified NTP server.

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

Step 4 Save or exit the configuration session.

end

commit

- **end**
- **commit**

Example:

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

or

```
RP/0/RP0/CPU0:ncs1010(config-ntp)#commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

```
Uncommitted changes found, commit them before
  exiting(yes/no/cancel)?
[cancel]:
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns to EXEC mode.
- Entering **no** exits the configuration session and returns to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the system in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Step 5 Verify the running NTP configuration.

show running-config ntp

Example:

```
RP/0/RP0/CPU0:ios#show running-config ntp

Sun Nov  5 15:14:24.969 UTC

ntp

 server 192.0.2.51 burst iburst

!
```

Displays the running configuration.

The synchronize the clock with an NTP server task is complete.

Verify NTP synchronization status

Use this procedure to verify NTP synchronization status.

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

Procedure

Step 1 Verify NTP associations.

show ntp associations

Example:

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

address ref clock st when poll reach delay offset disp
*~192.0.2.1 198.51.100.1 2 81 128 377 1.84 7.802 2.129
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

selected, + candidate, - outlayer, x falseticker, ~ configured

Displays the status of NTP associations.

Step 2 Verify detailed NTP association information.

show ntp associations detail

Example:

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

192.0.2.1 configured, our_master, stratum 2
ref ID 198.51.100.1, time E8F22BB9.79D4A841 (14:56:57.475 UTC Sun Nov 5 2023)
our mode client, peer mode server, our poll intvl 128, peer poll intvl 128
root delay 0.6866 msec, root disp 1.04, reach 377, sync dist 6.2590
delay 1.84 msec, offset 7.802 msec, dispersion 2.129
precision 2**23, version 4
org time E8F22F92.B647E8FC (15:13:22.712 UTC Sun Nov 5 2023)
rcv time E8F22F92.B88F303C (15:13:22.720 UTC Sun Nov 5 2023)
xmt time E8F22F92.B88F303C (15:13:22.720 UTC Sun Nov 5 2023)
filtdelay = 1.844 1.772 1.983 1.954 1.945 2.000 1.902 1.778
filtoffset = 7.857 7.802 8.065 8.063 8.332 8.397 8.664 8.684
filterror = 0.000 0.060 1.995 2.055 4.050 4.110 6.060 6.120
```

Step 3 Verify detailed NTP association information for a location.

show ntp associations detail location 0/RP0/CPU0

Example:

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

192.0.2.1 configured, our_master, stratum 2
ref ID 198.51.100.1, time E8F233C0.5606A159 (15:31:12.336 UTC Sun Nov 5 2023)
our mode client, peer mode server, our poll intvl 128, peer poll intvl 128
root delay 0.7019 msec, root disp 0.47, reach 377, sync dist 5.6762
delay 2.01 msec, offset 7.226 msec, dispersion 3.856
precision 2**23, version 4
org time E8F23563.DE5D42D5 (15:38:11.868 UTC Sun Nov 5 2023)
rcv time E8F23563.E07C296D (15:38:11.876 UTC Sun Nov 5 2023)
xmt time E8F23563.E07C296D (15:38:11.876 UTC Sun Nov 5 2023)
filtdelay = 2.006 1.865 1.936 1.762 1.932 1.875 1.881 2.011
filtoffset = 7.210 7.305 7.372 7.226 7.298 7.258 7.251 7.224
filterror = 0.000 2.025 2.085 4.035 4.095 6.060 6.120 8.070
```

Step 4 Verify NTP status.

show ntp status

Example:

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

Clock is synchronized, stratum 3, reference is 192.0.2.1
nominal freq is 1000000000.0000 Hz, actual freq is 44881851.3383 Hz, precision is 2**24
reference time is E8F22D7A.AB020D97 (15:04:26.668 UTC Sun Nov 5 2023)
clock offset is 9.690 msec, root delay is 2.553 msec
root dispersion is 24.15 msec, peer dispersion is 2.13 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.0000212807 s/s
system poll interval is 128, last update was 610 sec ago
authenticate is disabled, panic handling is disabled,
hostname resolution retry interval is 1440 minutes.
```

Verifies that the clock is synchronized with the NTP server.

The verify NTP synchronization status task is complete.

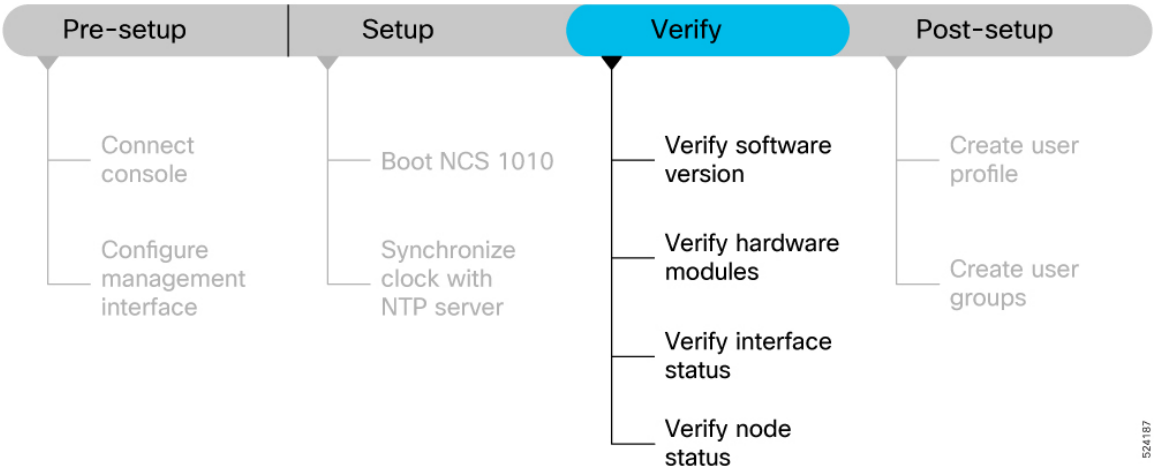
NTP troubleshooting reference for Cisco NCS 1010

For details about NTP issue resolution, see Troubleshoot Network Time Protocol Issues.

Cisco NCS 1010 software and hardware verification

After logging into the console, perform preliminary checks to verify the default setup.

Figure 3: Verification Workflow for the Cisco NCS 1010 Setup



Complete the procedures in Cisco NCS 1010 setup workflow before you proceed with the verification tasks.

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 the software version

Use this procedure to verify the software version.

View the software version installed on the NCS 1010.

Procedure

Verify the latest version of the Cisco IOS XR software installed on the NCS 1010.

show version

Example:

```
RP/0/RP0/CPU0:ios#show version
Sat Mar 25 11:38:23.614 IST
Cisco IOS XR Software, Version 24.3.1
Copyright (c) 2013-2023 by Cisco Systems, Inc.
Build Information:
Built By : ingunawa
Built On : Tue Mar 07 02:22:55 UTC 2023
Build Host : iox-ucs-063
Workspace : /auto/iox-ucs-063-san2/prod/203.0.113.1I.SIT_IMAGE/ncs1010/ws
Version : 24.3.1
Label : 24.3.1
cisco NCS1010 (C3758 @ 2.20GHz)
cisco NCS1010-SA (C3758 @ 2.20GHz) processor with 32GB of memory
OLT-C-R-SITE-1 uptime is 2 weeks, 12 hours, 59 minutes
NCS 1010 - Chassis
```

Note
You must upgrade the system if a new version of the system is available to avail the latest features on the NCS 1010.

For more information about upgrading the software version, see ../upgrade/c-upgrade-software.xml.

The **show version** only displays the IOS XR version in the label field if modific ations are made to the running software on the booted ISO image during installation of a newer version.

The verify the software version task is complete.

Verify hardware modules

Use this procedure to verify hardware modules.

Cisco NCS 1010 have various hardware modules such as processors, line cards, fan trays, and power modules installed on the NCS 1010. Ensure that the firmware on various hardware components of the NCS 1010 is compatible with the installed Cisco IOS XR image. You also must verify that all the installed hardware and firmware modules are operational.

Procedure

Step 1 Verify the status of the hardware modules.

show platform

Example:

```
RP/0/RP0/CPU0:ios#show platform
Wed Apr 27 08:43:40.130 UTC
Node               Type                      State                 Config state
--------------------------------------------------------------------------------
0/RP0/CPU0         NCS1010-CNTLR-K9(Active)  IOS XR RUN            NSHUT,NMON
0/PM0              NCS1010-AC-PSU            OFFLINE               NSHUT,NMON
0/PM1              NCS1010-AC-PSU            OPERATIONAL           NSHUT,NMON
0/FT0              NCS1010-FAN               OPERATIONAL           NSHUT,NMON
0/FT1              NCS1010-FAN               OPERATIONAL           NSHUT,NMON
0/0/NXR0           NCS1K-OLT-C               OPERATIONAL           NSHUT,NMON
0/1                NCS1K-BRK-SA              OPERATIONAL           NSHUT,NMON
0/1/0              NCS1K-BRK-8               OPERATIONAL           NSHUT,NMON
```

```
0/1/1             NCS1K-BRK-8             OPERATIONAL             NSHUT,NMON
0/1/2             NCS1K-BRK-24            OPERATIONAL             NSHUT,NMON
0/1/3             NCS1K-BRK-24            OPERATIONAL             NSHUT,NMON
0/2               NCS1K-MD-32E-C          OPERATIONAL             NSHUT,NMON
0/3               NCS1K-MD-32O-C          OPERATIONAL             NSHUT,NMON
```

Step 2 View the list of hardware and firmware modules that are detected on the NCS 1010.

Note
From R26.2.1, the SSD FPD name displayed in the output of show hw-module fpd command is changed from a vendor or model-specific name to a generic format: 'CPU-SSD' for RP locations and 'CHASSIS-SSD' for rack locations.

show hw-module fpd command output until R26.2.1

Example:

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

Mon Dec 22 10:01:56.338 UTC
Auto-upgrade:Enabled,PM excluded
Attribute codes: B golden, P protect, S secure, A Anti Theft aware
                                                                          FPD Versions
                                                                          ==============
Location   Card type             HWver FPD device       ATR Status    Running  Programd   Reload Loc
-------------------------------------------------------------------------------------------------
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   ADMCONFIG              CURRENT      1.00     1.00        NOT
REQ
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   BIOS             S     CURRENT      6.10     6.10         0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   BIOS-Golden      BS    CURRENT              1.30         0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   CpuFpga          S     CURRENT      1.12     1.12         0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   CpuFpgaGolden    BS    CURRENT              0.07         0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   SsdMicron5300    S     CURRENT      0.01     0.01         0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   TamFw            S     CURRENT      9.07     9.07         0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   TamFwGolden      BS    CURRENT              9.05         0/RP0
0/PM0      NCS1K4-AC-PSU-2       1.0   PO-PriMCU              NOT READY
N/A
0/PM0      NCS1K4-AC-PSU-2       1.0   PO-SecMCU              CURRENT      1.05     1.05        NOT
REQ
0/PM1      NCS1K4-AC-PSU-2       1.0   PO-PriMCU              CURRENT      1.03     1.03        NOT
REQ
0/PM1      NCS1K4-AC-PSU-2       1.0   PO-SecMCU              CURRENT      1.05     1.05        NOT
REQ
0/0/NXR0   NCS1K-E-ILA-RE-C      0.1   ILA              S     CURRENT      3.44     3.44        NOT
REQ
0/0/NXR0   NCS1K-E-ILA-RE-C      0.1   Raman-E-1        S     CURRENT      4.04     4.04        NOT
REQ
0/1/NXR0   NCS1K-E2-OLT-RE-C    0.1   E2_OLT           S     CURRENT      4.01     4.01        NOT
REQ
0/1/NXR0   NCS1K-E2-OLT-RE-C    0.1   Raman-E-1        S     CURRENT      4.04     4.04        NOT
REQ
0/Rack     NCS1020-SA            0.1   ADMCONFIG              CURRENT      1.00     1.00        NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaLow        S     CURRENT      1.12     1.12        NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaLowGolden  BS    CURRENT              0.07        NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaUp         S     CURRENT      1.10     1.10        NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaUpGolden   BS    CURRENT              0.06        NOT
REQ
0/Rack     NCS1020-SA           0.1   SsdIntelSC2KB    S     CURRENT      1.30     1.30        0/Rack
```

show hw-module fpd command output from R26.2.1

Example:

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

Mon Dec 22 09:27:51.696 UTC
Auto-upgrade:Enabled,PM excluded
Attribute codes: B golden, P protect, S secure, A Anti Theft aware
                                                                                FPD Versions
                                                                                ============
Location   Card type             HWver FPD device       ATR Status    Running  Programd   Reload Loc
-------------------------------------------------------------------------------------------------
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   ADMCONFIG             CURRENT      1.00     1.00         NOT
REQ
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   BIOS             S   CURRENT      6.10     6.10          0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   BIOS-Golden      BS  CURRENT               1.30          0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   CPU-SSD          S   CURRENT      0.01     0.01          0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   CpuFpga          S   CURRENT      1.12     1.12          0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   CpuFpgaGolden    BS  CURRENT               0.07          0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   TamFw            S   CURRENT      9.07     9.07          0/RP0
0/RP0/CPU0 NCS1010-CNT-B-K9      0.1   TamFwGolden      BS  CURRENT               9.05          0/RP0
0/PM0      NCS1K4-AC-PSU-2       1.0   PO-PriMCU             NOT READY
N/A
0/PM0      NCS1K4-AC-PSU-2       1.0   PO-SecMCU             CURRENT      1.05     1.05         NOT
REQ
0/PM1      NCS1K4-AC-PSU-2       1.0   PO-PriMCU             CURRENT      1.03     1.03         NOT
REQ
0/PM1      NCS1K4-AC-PSU-2       1.0   PO-SecMCU             CURRENT      1.05     1.05         NOT
REQ
0/0/NXR0   NCS1K-E-ILA-RE-C      0.1   ILA              S    CURRENT      3.44     3.44         NOT
REQ
0/0/NXR0   NCS1K-E-ILA-RE-C      0.1   Raman-E-1        S    CURRENT      4.04     4.04         NOT
REQ
0/1/NXR0   NCS1K-E2-OLT-RE-C     0.1   E2_OLT           S    CURRENT      4.01     4.01         NOT
REQ
0/1/NXR0   NCS1K-E2-OLT-RE-C     0.1   Raman-E-1        S    CURRENT      4.04     4.04         NOT
REQ
0/Rack     NCS1020-SA            0.1   ADMCONFIG             CURRENT      1.00     1.00         NOT
REQ
0/Rack     NCS1020-SA           0.1   CHASSIS-SSD      S   CURRENT      1.30     1.30         0/Rack
0/Rack     NCS1020-SA            0.1   IoFpgaLow        S    CURRENT      1.12     1.12         NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaLowGolden  BS   CURRENT               0.07         NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaUp         S    CURRENT      1.10     1.10         NOT
REQ
0/Rack     NCS1020-SA            0.1   IoFpgaUpGolden   BS   CURRENT               0.06         NOT
REQ
```

From the **show hw-module fpd** output, verify that all hardware modules that are installed on the chassis are listed. An unlisted module indicates that the module is either malfunctioning, or has not been installed properly. You must remove and reinstall the hardware module.

The fields in the **show hw-module fpd** output are:

- **FPD Device**: Name of the hardware component, such as IO FPGA, or BIOS. The Golden FPDs are not field upgradable.
- **Running**: Current version of the firmware running on the FPD.
- **Programd**: Version of the FPD programmed on the module

- **Status**: Upgrade status of the firmware. The different states are:

Table 4: Status and Description of the Firmware Upgrade

Status	Description
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 new firmware version is available in the installed image. Cisco recommends that you perform an upgrade of the firmware version.
RLOAD REQ	The upgrade is complete, 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 corrupt. Reinstall the firmware.
UPGD SKIP	The upgrade is skipped because the installed firmware version is higher than the one available in the image.

Step 3 Upgrade the required firmware as required,.

upgrade hw-module location all fpd all

Example:

```
RP/0/RP0/CPU0:ios#upgrade hw-module location all fpd all
Alarms are created showing all modules that needs to be upgraded.

Active Alarms
------------------------------------------------------------------------------------------------------------
Location     Severity  Group      Set Time                  Description
------------------------------------------------------------------------------------------------------------
0/6/CPU0     Major     FPD_Infra  09/16/2019 12:34:59 UTC   One Or More FPDs Need Upgrade Or Not In
 Current State
0/10/CPU0    Major     FPD_Infra  09/16/2019 12:34:59 UTC   One Or More FPDs Need Upgrade Or Not In
 Current State
0/RP0/CPU0   Major     FPD_Infra  09/16/2019 12:34:59 UTC   One Or More FPDs Need Upgrade Or Not In
 Current State
0/RP1/CPU0   Major     FPD_Infra  09/16/2019 12:34:59 UTC   One Or More FPDs Need Upgrade Or Not In
 Current State
0/FC0        Major     FPD_Infra  09/16/2019 12:34:59 UTC   One Or More FPDs Need Upgrade Or Not In
 Current State
0/FC1        Major     FPD_Infra  09/16/2019 12:34:59 UTC   One Or More FPDs Need Upgrade Or Not In
 Current State
```

Note
The BIOS and IOFPGA upgrades require a restart of the NCS 1010 for the new version to take effect.

Step 4 Verify status of the modules after upgrade.

show hw-module fpd REQ

Example:

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

```
Wed Jun 29 08:50:21.057 UTC

Auto-upgrade:Disabled
                                                                                FPD Versions
                                                                               ==============
Location   Card type             HWver FPD device       ATR Status    Running   Programd  Reload Loc

-------------------------------------------------------------------------------------------------
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   ADMConfig            CURRENT    3.40    3.40       NOT REQ
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   BIOS              S  CURRENT    4.10    4.10         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   BIOS-Golden      BS  CURRENT            4.10         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   CpuFpga           S  CURRENT    1.02    1.02         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   CpuFpgaGolden    BS  CURRENT            1.01         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   SsdIntelS4510     S  CURRENT   11.32   11.32         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   TamFw             S  CURRENT    6.13    6.13         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.0   TamFwGolden      BS  CURRENT            6.11         0/RP0
0/PM0      NCS1010-AC-PSU        0.0   AP-PriMCU            CURRENT    1.03    1.03       NOT REQ
0/PM0      NCS1010-AC-PSU        0.0   AP-SecMCU            CURRENT    2.01    2.01       NOT REQ
0/PM1      NCS1010-AC-PSU        0.0   AP-PriMCU            CURRENT    1.03    1.03       NOT REQ
0/PM1      NCS1010-AC-PSU        0.0   AP-SecMCU            CURRENT    2.01    2.01       NOT REQ
0/0/NXR0   NCS1K-ILA-C           1.0   ILA               S  CURRENT    1.00    1.00       NOT REQ
0/Rack     NCS1010-SA            1.0   EITU-ADMConfig       CURRENT    2.10    2.10       NOT REQ
0/Rack     NCS1010-SA            1.0   IoFpga            S  CURRENT    1.04    1.04       NOT REQ
0/Rack     NCS1010-SA            1.0   IoFpgaGolden     BS  CURRENT            1.01       NOT REQ
0/Rack     NCS1010-SA            1.0   SsdIntelS4510     S  CURRENT   11.32   11.32        0/Rack
```

The status of the upgraded nodes shows that a reload is required.

Step 5 Reload the individual nodes that require an upgrade.

reload location node-location

Example:

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

Step 6 Verify that all nodes that had required an upgrade now shows an updated status of CURRENT with an updated FPD version.

Example:

```
Thu Mar  2 12:35:06.602 IST

Auto-upgrade:Enabled
Attribute codes: B golden, P protect, S secure, A Anti Theft aware
                                                                                FPD Versions
                                                                               ==============
Location   Card type             HWver FPD device       ATR Status   Running Programd  Reload Loc
-------------------------------------------------------------------------------------------------
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  ADMConfig            CURRENT    3.40    3.40       NOT REQ
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  BIOS              S  CURRENT    4.20    4.20         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  BIOS-Golden      BS  CURRENT            4.10         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  CpuFpga           S  CURRENT    1.11    1.11         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  CpuFpgaGolden    BS  CURRENT            1.01         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  SsdIntelS4510     S  CURRENT   11.32   11.32         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  TamFw             S  CURRENT    6.13    6.13         0/RP0
0/RP0/CPU0 NCS1010-CNTLR-K9      1.11  TamFwGolden      BS  CURRENT            6.11         0/RP0
0/PM0      NCS1010-AC-PSU        0.0   AP-PriMCU            CURRENT    1.03    1.03       NOT REQ
0/PM0      NCS1010-AC-PSU        0.0   AP-SecMCU            CURRENT    2.01    2.01       NOT REQ
0/PM1      NCS1010-AC-PSU        0.0   AP-PriMCU            CURRENT    1.03    1.03       NOT REQ
0/PM1      NCS1010-AC-PSU        0.0   AP-SecMCU            CURRENT    2.01    2.01       NOT REQ
0/0/NXR0   NCS1K-OLT-L           1.0   OLT               S  CURRENT    1.02    1.02       NOT REQ
0/Rack     NCS1010-SA            2.1   EITU-ADMConfig       CURRENT    2.10    2.10       NOT REQ
```

```
0/Rack     NCS1010-SA            2.1   IoFpga           S   CURRENT    1.12    1.12       NOT REQ
0/Rack     NCS1010-SA            2.1   IoFpgaGolden     BS  CURRENT            1.01       NOT REQ
0/Rack     NCS1010-SA            2.1   SsdIntelS4510    S   CURRENT   11.32   11.32        0/Rack
```

The verify hardware modules task is complete.

Verify interface status

Use this procedure to verify interface status.

All available interfaces must be discovered by the system after booting the Cisco NCS 1010. Interfaces not discovered might indicate a malfunction in the unit.

Procedure

View the interfaces discovered by the system.

show ipv4 interfaces brief

Example:

```
RP/0/RP0/CPU0:ios#show ipv4 interfaces brief
Wed May 25 11:50:28.438 UTC

               Intf     Intf        LineP              Encap         MTU        BW
               Name    State       State               Type        (byte)    (Kbps)
-------------------------------------------------------------------------------------------------------

                Lo0       up          up              Loopback       1500          0
                Lo3       up          up              Loopback       1500          0
                Nu0       up          up               Null           1500          0
          Gi0/0/0/0       up          up               ARPA           1514    1000000
        Mg0/RP0/CPU0/0     up          up               ARPA           1514    1000000
       Mg0/RP0/CPU0/1    admin-down  admin-down        ARPA  1514     1000000
       Mg0/RP0/CPU0/2    admin-down  admin-down        ARPA  1514     1000000
       PT0/RP0/CPU0/0    admin-down  admin-down        ARPA  1514     1000000
       PT0/RP0/CPU0/1    admin-down  admin-down        ARPA  1514     1000000
```

Example:

```
RP/0/RP0/CPU0:ios#show ipv4 interfaces brief
Tue Jul 12 07:32:42.390 UTC

Interface                      IP-Address      Status          Protocol Vrf-Name
Loopback0                      198.51.100.1       Up            Up       default
Loopback3                      203.0.113.1        Up            Up       default
GigabitEthernet0/0/0/0         192.0.2.1          Up            Up       default
MgmtEth0/RP0/CPU0/0            192.0.2.255        Up            Up       default
PTP0/RP0/CPU0/0                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/1            unassigned      Down            Down     default
PTP0/RP0/CPU0/1                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/2            unassigned      Down            Down     default
```

When a NCS 1010 is turned ON for the first time, all interfaces are in the **unassigned** state.

Ensure that the total number of interfaces that are displayed in the result matches with the actual number of interfaces present on the NCS 1010, and that the interfaces are created according to the type of line cards displayed in **show platform** command.

The verify interface status task is complete.

Verify node status

Use this procedure to verify node status.

A node can be a specified location, or the complete hardware module in the system. You must verify that the software state of all route processors, line cards, and the hardware state of fabric cards, fan trays, and power modules are listed, and their state is OPERATIONAL. This indicates that the IOS XR console is operational on the cards.

Procedure

Verify the operational status of the node.

show platform

Example:

```
RP/0/RP0/CPU0:ios#show platform
Wed Apr 27 08:43:40.130 UTC
Node               Type                      State                 Config state
--------------------------------------------------------------------------------
0/RP0/CPU0         NCS1010-CNTLR-K9(Active)  IOS XR RUN            NSHUT,NMON
0/PM0              NCS1010-AC-PSU            OFFLINE               NSHUT,NMON
0/PM1              NCS1010-AC-PSU            OPERATIONAL           NSHUT,NMON
0/FT0              NCS1010-FAN               OPERATIONAL           NSHUT,NMON
0/FT1              NCS1010-FAN               OPERATIONAL           NSHUT,NMON
0/0/NXR0           NCS1K-OLT-C               OPERATIONAL           NSHUT,NMON
0/1                NCS1K-BRK-SA              OPERATIONAL           NSHUT,NMON
0/1/0              NCS1K-BRK-8               OPERATIONAL           NSHUT,NMON
0/1/1              NCS1K-BRK-8               OPERATIONAL           NSHUT,NMON
0/1/2              NCS1K-BRK-24              OPERATIONAL           NSHUT,NMON
0/1/3              NCS1K-BRK-24              OPERATIONAL           NSHUT,NMON
0/2                NCS1K-MD-32E-C            OPERATIONAL           NSHUT,NMON
0/3                NCS1K-MD-32O-C            OPERATIONAL           NSHUT,NMON
```

Example:

```
RP/0/RP0/CPU0:ios#show platform
Thu Mar  2 12:35:01.883 IST
Node               Type                      State                 Config state
--------------------------------------------------------------------------------
0/RP0/CPU0         NCS1010-CNTLR-K9(Active)  IOS XR RUN            NSHUT,NMON
0/PM0              NCS1010-AC-PSU            OPERATIONAL           NSHUT,NMON
0/PM1              NCS1010-AC-PSU            OFFLINE               NSHUT,NMON
0/FT0              NCS1010-FAN               OPERATIONAL           NSHUT,NMON
0/FT1              NCS1010-FAN               OPERATIONAL           NSHUT,NMON
0/0/NXR0           NCS1K-OLT-L               OPERATIONAL           NSHUT,NMON
0/3                NCS1K-BRK-24              OPERATIONAL           NSHUT,NMON
```

The verify node status task is complete.

What to do next

This completes verification of the basic NCS 1010 setup. You can now complete the post-setup tasks where you manage user profiles and groups.

Verify hardware inventory

Use this procedure to verify hardware inventory.

The **show inventory** command displays details of the hardware inventory of NCS 1010.

To verify the inventory information for all the physical entities, perform this procedure.

Procedure

Verify hardware inventory.

show inventory

Displays the details of the physical entities of NCS 1010 along with the details of SFPs.

Example:

```
RP/0/RP0/CPU0:ios#show inventory
Wed Apr 27 08:43:44.222 UTC

NAME: "Rack 0", DESCR: "NCS1010 - Shelf Assembly"
PID: NCS1010-SA        , VID: V00, SN: FCB2504B0X4

NAME: "0/RP0/CPU0", DESCR: "Network Convergence System 1010 Controller"
PID: NCS1010-CNTLR-K9  , VID: V00, SN: FCB2506B0NX

NAME: "0/1", DESCR: "NCS 1000 shelf for 4 passive modules"
PID: NCS1K-BRK-SA      , VID: V00 , SN: FCB2534B0GR

NAME: "0/1/0", DESCR: "NCS 1000 MTP/MPO to 8 port passive breakout module"
PID: NCS1K-BRK-8       , VID: V00 , SN: MPM25401005

NAME: "0/1/1", DESCR: "NCS 1000 MTP/MPO to 8 port passive breakout module"
PID: NCS1K-BRK-8       , VID: V00 , SN: MPM25401003

NAME: "0/1/2", DESCR: "NCS 1000 MTP/MPO to 24 colorless chs passive breakout module"
PID: NCS1K-BRK-24      , VID: V00 , SN: MPM25141004

NAME: "0/1/3", DESCR: "NCS 1000 MTP/MPO to 24 colorless chs passive breakout module"
PID: NCS1K-BRK-24      , VID: V00 , SN: MPM25371005

NAME: "0/2", DESCR: "NCS 1000 32 chs Even Mux/Demux Patch Panel - 150GHz - C-band"
PID: NCS1K-MD-32E-C    , VID: V00 , SN: ACW2529YE13

NAME: "0/3", DESCR: "NCS 1000 32 chs Odd  Mux/Demux Patch Panel - 150GHz - C-band"
PID: NCS1K-MD-32O-C    , VID: V00 , SN: ACW2529YA13

NAME: "0/FT0", DESCR: "NCS1010 - Shelf Fan"
PID: NCS1010-FAN       , VID: V00, SN: FCB2504B0W3

NAME: "0/FT1", DESCR: "NCS1010 - Shelf Fan"
```

```
PID: NCS1010-FAN       , VID: V00, SN: FCB2504B0U8

NAME: "0/PM0", DESCR: "NCS 1010 - AC Power Supply Unit"
PID: NCS1010-AC-PSU    , VID: V00, SN: APS244700D0

NAME: "0/PM1", DESCR: "NCS 1010 - AC Power Supply Unit"
PID: NCS1010-AC-PSU    , VID: V00, SN: APS244700BY
```

The verify hardware inventory task is complete.

Verify management interface status

Use this procedure to verify management interface status.

To verify the management interface status, perform this procedure.

Procedure

Step 1 Verify the management interface configuration.

show interfaces MgmtEth 0/RP0/CPU0/0

Displays the management interface configuration.

Example:

```
RP/0/RP0/CPU0:ios#show interfaces MgmtEth 0/RP0/CPU0/0
Wed May 25 11:49:18.118 UTC
MgmtEth0/RP0/CPU0/0 is up, line protocol is up
  Interface state transitions: 1
  Hardware is Management Ethernet, address is 38fd.f866.0964 (bia 38fd.f866.0964)
  Internet address is 192.0.2.254/16
  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 15:05:21
  ARP type ARPA, ARP timeout 04:00:00
  Last input never, output 00:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     53138 packets input, 6636701 bytes, 0 total input drops
     0 drops for unrecognized upper-level protocol
     Received 12145 broadcast packets, 40082 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     217288 packets output, 60964220 bytes, 0 total output drops
     Output 1 broadcast packets, 15 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions
```

Step 2 Verify the management interface status.

show interfaces summary

show interfaces brief

show ipv4 interfaces brief

Verifies the management interface status.

Example:

```
RP/0/RP0/CPU0:ios#show interfaces summary
Mon Nov  4 18:10:14.996 IST
Interface Type          Total    UP       Down     Admin Down
--------------          -----    --       ----     ----------
ALL TYPES               9        7        0        2
--------------
IFT_GETHERNET           1        1        0        0
IFT_LOOPBACK            1        1        0        0
IFT_ETHERNET            4        4        0        0
IFT_NULL                1        1        0        0
IFT_PTP_ETHERNET        2        0        0        2
```

Example:

```
RP/0/RP0/CPU0:ios#show interfaces brief
Mon Nov  4 18:11:37.222 IST

               Intf       Intf        LineP              Encap  MTU        BW
               Name       State       State               Type (byte)    (Kbps)
--------------------------------------------------------------------------------
                Lo0          up          up           Loopback  1500          0
                Nu0          up          up               Null  1500          0
          Gi0/0/0/0          up          up               ARPA  1514     100000
     Mg0/RP0/CPU0/0          up          up               ARPA  1514    1000000
     Mg0/RP0/CPU0/1          up          up               ARPA  1514    1000000
     Mg0/RP0/CPU0/2          up          up               ARPA  1514    1000000
     PT0/RP0/CPU0/0  admin-down  admin-down               ARPA  1514    1000000
     PT0/RP0/CPU0/1  admin-down  admin-down               ARPA  1514    1000000
    Mg0/RP0/RCOM0/0          up          up               ARPA  1514    1000000
```

Example:

```
RP/0/RP0/CPU0:ios#show ipv4 interfaces brief
Mon Nov  4 18:12:32.082 IST

Interface                      IP-Address      Status          Protocol Vrf-Name
Loopback0                      192.0.2.1       Up              Up       default
GigabitEthernet0/0/0/0         192.0.2.1       Up              Up       default
MgmtEth0/RP0/CPU0/0            192.0.2.254     Up              Up       default
PTP0/RP0/CPU0/0                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/1            203.0.113.1     Up              Up       default
PTP0/RP0/CPU0/1                unassigned      Shutdown        Down     default
MgmtEth0/RP0/CPU0/2            192.0.2.255     Up              Up       default
MgmtEth0/RP0/RCOM0/0           unassigned      Up              Up       default
```

The verify management interface status task is complete.

Verify alarms

Use this procedure to verify alarms.

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

Procedure

Verify active alarms.

show alarms brief system active

Displays alarms in brief or detail.

Example:

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

Thu Apr 28 06:16:50.524 UTC

------------------------------------------------------------------------------------
Active Alarms
------------------------------------------------------------------------------------
Location        Severity     Group            Set Time                    Description
------------------------------------------------------------------------------------
0/RP0/CPU0      Major        Ethernet         04/28/2022 06:03:39 UTC    RP-SW: SPI flash config is
 incorrect
0/PM0           Major        Environ          04/28/2022 06:03:50 UTC    Power Module Error
(PM_VIN_VOLT_OOR)
0/PM0           Major        Environ          04/28/2022 06:03:50 UTC    Power Module Output Disabled
 (PM_OUTPUT_DISABLED)
0               Major        Environ          04/28/2022 06:03:50 UTC    Power Group redundancy lost
0/PM0           Major       FPD_Infra        04/28/2022 06:04:08 UTC    One Or More FPDs Need Upgrade
 Or Not In Current State
0/PM1           Major       FPD_Infra        04/28/2022 06:04:09 UTC    One Or More FPDs Need Upgrade
 Or Not In Current State
0/0             Major        Controller       04/28/2022 06:05:12 UTC    Osc0/0/0/0 - Provisioning
Failed
0/0             Major        Controller       04/28/2022 06:05:12 UTC    Osc0/0/0/2 - Provisioning
Failed
0/0             Major        Controller       04/28/2022 06:05:12 UTC    Ots0/0/0/0 - Provisioning
Failed
0/0             Major        Controller       04/28/2022 06:05:12 UTC    Ots0/0/0/2 - Provisioning
Failed
```

Note
In the maintenance mode, all the alarms are moved from active to suppressed and the **show alarms** command does not display the alarms details.

The verify alarms task is complete.

Verify environmental parameters

Use this procedure to verify environmental parameters.

The **show environment** command displays the environmental parameters of NCS 1010.

To verify that the environmental parameters are as expected, perform this procedure.

Procedure

Step 1 Verify fan environmental parameters.

show environment fan

Example:

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

```
RP/0/RP0/CPU0:ios#show environment fan
Thu May 26 04:15:37.765 UTC
==============================================================================
                                          Fan speed (rpm)
Location      FRU Type                    FAN_0    FAN_1   FAN_2
------------------------------------------------------------------------------

0/PM0         NCS1010-AC-PSU               5368
0/PM1         NCS1010-AC-PSU               5336
0/FT0         NCS1010-FAN                 10020   10020   10020
0/FT1         NCS1010-FAN                 10020   10020    9960

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

Displays the environmental parameters for the selected command output.

Step 2 Verify route processor temperature parameters.

show environment temperature location 0/RP0

Example:

This example shows a sample output of the **show environment** command with the **temperatures** keyword for *0/RP0 location*.

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

Thu May 26 04:16:39.832 UTC
=============================================================================================================================
Location  TEMPERATURE                          Value     Crit    Major    Minor    Minor    Major    Crit
          Sensor                             (deg C)     (Lo)     (Lo)     (Lo)     (Hi)     (Hi)    (Hi)
-----------------------------------------------------------------------------------------------------------------------------
0/RP0/CPU0
          RP_TEMP_PCB                          30      -10       -5        0       70       75      80
          RP_TEMP_HOT_SPOT                     33      -10       -5        0       70       75      80
          RP_TEMP_LTM4638                      49      -10       -5        0       80       85      90
          RP_TEMP_LTM4644_0                    36      -10       -5        0       80       85      90
          RP_TEMP_LTM4644_1                    39      -10       -5        0       80       85      90
          RP_JMAC_1V0_VCCP_TMON                33      -10       -5        0       80       85      90
          RP_JMAC_1V0_VNN_TMON                 33      -10       -5        0       80       85      90
          RP_JMAC_1V0_VCC_RAM_TMON             32      -10       -5        0       80       85      90
          RP_JMAC_1V2_DDR_VDDQ_TMON            33      -10       -5        0       80       85
```

```
    90
==============================================================================================================
```

Step 3 Verify NXR temperature parameters.

show environment temperature location 0/0/NXR0

Example:

This example shows a sample output of the **show environment** command with the **temperatures** keyword for *0/0/NXR0 location*.

```
RP/0/RP0/CPU0:ios#show environment temperature location 0/0/NXR0

Thu May 26 04:16:39.832 UTC
==============================================================================================================
Location  TEMPERATURE                            Value      Crit    Major    Minor    Minor    Major
  Crit
          Sensor                               (deg C)      (Lo)     (Lo)     (Lo)     (Hi)     (Hi)
  (Hi)
--------------------------------------------------------------------------------------------------------------
0/0/NXR0
          OLTC_LT_P0_iEDFA0                      24       18       19       20       30       31
    32
          OLTC_LT_P0_iEDFA1                      25       18       19       20       30       31
    32
          OLTC_LT_P0_iEDFA2                      24       18       19       20       30       31
    32
          OLTC_LT_P2_iEDFA0                      25       18       19       20       30       31
    32
          OLTC_LT_P3_iEDFA0                      25       18       19       20       30       31
    32
          OLTC_LT_P0_eEDFA0                      24       18       19       20       30       31
    32
          OLTC_CT_1                              32      -10       -7       -5       75       77
    80
          OLTC_LT_P0_eEDFA1                      24       18       19       20       30       31
    32
          OLTC_CT_2                              27      -10       -7       -5       70       73
    75
          OLTC_CT_3                              30      -10       -7       -5       70       73
    75
          OLTC_CT_4                              30      -10       -7       -5       70       73
    75
          OLTC_FT_P0_iEDFA0                      60       55       57       58       62       64
    65
          OLTC_FT_P2_iEDFA0                      60       55       57       58       62       64
    65
          OLTC_FT_P3_iEDFA0                      60       55       57       58       62       64
    65
          OLTC_FT_P0_eEDFA0                      60       55       57       58       62       64
    65
==============================================================================================================
```

Step 4 Verify power environmental parameters.

show environment power

Example:

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

```
RP/0/RP0/CPU0:ios#show environment power
Thu May 26 04:17:55.592 UTC
================================================================================
```

```
CHASSIS LEVEL POWER INFO: 0
================================================================================
   Total output power capacity (Group 0 + Group 1) :    1050W +     1050W
   Total output power required                     :     700W
   Total power input                               :     228W
   Total power output                              :     140W

Power Group 0:
================================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type            Volts     Amps    Volts     Amps
================================================================================
   0/PM0       NCS1010-AC-PSU  228.5     0.5     12.1      5.6      OK

Total of Group 0:              114W/0.5A         67W/5.6A

Power Group 1:
================================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type            Volts     Amps    Volts     Amps
================================================================================
   0/PM1       NCS1010-AC-PSU  228.5     0.5     12.1      6.1      OK

Total of Group 1:              114W/0.5A         73W/6.1A

================================================================================
   Location     Card Type               Power       Power        Status
                                        Allocated   Used
                                        Watts       Watts
================================================================================
   0/RP0/CPU0   NCS1010-CNTLR-K9        90          14           ON
   0/FT0        NCS1010-FAN             110         17           ON
   0/FT1        NCS1010-FAN             110         15           ON
   0/0/NXR0     NCS1K-OLT-C             350         61           ON
   0/Rack       NCS1010-SA              40          19           ON


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

Step 5 Verify voltage environmental parameters.

show environment voltage location 0/RP0

Example:

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

```
RP/0/RP0/CPU0:ios#show environment voltage location 0/RP0
Thu May 26 04:19:16.636 UTC
================================================================================================
Location  VOLTAGE                                Value      Crit    Minor    Minor     Crit
          Sensor                                  (mV)       (Lo)    (Lo)     (Hi)      (Hi)
------------------------------------------------------------------------------------------------
0/RP0/CPU0
          RP_ADM1266_12V0                      12094    10800    11280    12720    13200
          RP_ADM1266_1V8_CPU                    1806     1670     1750     1850     1930
          RP_ADM1266_1V24_VCCREF                1238     1150     1200     1280     1330
          RP_ADM1266_1V05_CPU                   1047      980     1020     1080     1120
          RP_ADM1266_1V2_DDR_VDDQ               1204     1120     1160     1240     1280
          RP_ADM1266_1V0_VCC_RAM                 988      650      700     1250     1300
          RP_ADM1266_1V0_VNN                     869      550      600     1250     1300
          RP_ADM1266_1V0_VCCP                   1018      450      500     1250     1300
          RP_ADM1266_0V6_DDR_VTT                 599      560      580      620      640
          RP_ADM1266_3V3_STAND_BY               3301     3070     3200     3400     3530
          RP_ADM1266_5V0                        5004     4650     4850     5150     5350
```

```
          RP_ADM1266_3V3                      3325     3070     3200     3400     3530
          RP_ADM1266_2V5_PLL                  2489     2330     2430     2580     2680
          RP_ADM1266_2V5_FPGA                 2502     2330     2430     2580     2680
          RP_ADM1266_1V2_FPGA                 1202     1120     1160     1240     1280
          RP_ADM1266_3V3_CPU                  3332     3070     3200     3400     3530
          RP_ADM1266_2V5_CPU                  2498     2330     2430     2580     2680

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

Step 6 Verify current environmental parameters.

show environment current

Example:

This example shows a sample output of the **show environment** command with the **current** keyword.

```
RP/0/RP0/CPU0:P2C_DT_02#show environment current
Tue Jul  5 08:36:22.132 UTC
==============================================================================================
Location  CURRENT                                 Value
          Sensor                                  (mA)
----------------------------------------------------------------------------------------------
0/RP0/CPU0
          RP_CURRMON_LTM4638                       395
          RP_CURRMON_LTM4644_0                     179
          RP_CURRMON_LTM4644_1                     307
          RP_JMAC_1V0_VCCP_IMON                    187
          RP_JMAC_1V0_VNN_IMON                      62
          RP_JMAC_1V0_VCC_RAM_IMON                   0
          RP_JMAC_1V2_DDR_VDDQ_IMON                187
0/Rack
          SA_ADM1275_12V_MOD0_IMON                4154
          SA_ADM1275_12V_MOD1_IMON                  43
          SA_ADM1275_12V_MOD2_IMON                  18
          SA_ADM1275_12V_FAN0_IMON                1356
          SA_ADM1275_12V_FAN1_IMON                1517
          SA_INA230_5V0_IMON                       129
          SA_INA230_3V3_IMON                      2998
          SA_INA230_1V0_XGE_CORE_IMON             2464
          SA_INA230_1V0_FPGA_CORE_IMON             787
          SA_ADM1275_12V_SA_IMON                  1668
          SA_ADM1275_12V_CPU_IMON                 1147
```

Step 7 Verify altitude environmental parameters.

show environment altitude

Example:

This example shows a sample output of the **show environment** command with the **altitude** keyword.

```
RP/0/RP0/CPU0:P2C_DT_02#show environment altitude
Tue Jul  5 08:36:51.710 UTC
================================================================================
Location     Altitude Value (Meters)    Source
--------------------------------------------------------------------------------
0            760                    sensor
```

Step 8 Verify all environmental parameters.

show environment all

Example:

This example shows a sample output of the **show environment** command with the **all** keyword.

```
RP/0/RP0/CPU0:P2C_DT_02#show environment all

Tue Jul  5 08:37:28.412 UTC
==============================================================================================================
Location  TEMPERATURE                            Value     Crit    Major    Minor    Minor    Major
  Crit
          Sensor                               (deg C)     (Lo)     (Lo)     (Lo)     (Hi)     (Hi)
  (Hi)
--------------------------------------------------------------------------------------------------------------
0/RP0/CPU0
          RP_TEMP_PCB                             29      -10       -5        0       70       75
    80
          RP_TEMP_HOT_SPOT                        32      -10       -5        0       70       75
    80
          RP_TEMP_LTM4638                         45      -10       -5        0       80       85
    90
          RP_TEMP_LTM4644_0                       35      -10       -5        0       80       85
    90
          RP_TEMP_LTM4644_1                       38      -10       -5        0       80       85
    90
          RP_JMAC_1V0_VCCP_TMON                   30      -10       -5        0       80       85
    90
          RP_JMAC_1V0_VNN_TMON                    29      -10       -5        0       80       85
    90
          RP_JMAC_1V0_VCC_RAM_TMON                30      -10       -5        0       80       85
    90
          RP_JMAC_1V2_DDR_VDDQ_TMON               31      -10       -5        0       80       85
    90
0/PM0
          Ambient Temp                            29      -10       -5        0       55       60
    65
          Secondary HotSpot Temp                  50      -10       -5        0       85       90
    95
          Primary HotSpot Temp                    41      -10       -5        0       65       70
    75
0/0/NXR0
          ILAC_LT_P0_eEDFA0                       25       18       19       20       30       31
    32
          ILAC_LT_P0_eEDFA1                       25       18       19       20       30       31
    32
          ILAC_LT_P0_eEDFA2                       25       18       19       20       30       31
    32
          ILAC_LT_P2_eEDFA0                       25       18       19       20       30       31
    32
          ILAC_LT_P2_eEDFA1                       25       18       19       20       30       31
    32
          ILAC_LT_P2_eEDFA2                       25       18       19       20       30       31
    32
          ILAC_CT_1                               29      -10       -7       -5       75       77
    80
          ILAC_CT_2                               26      -10       -7       -5       70       73
    75
          ILAC_CT_3                               28      -10       -7       -5       70       73
    75
          ILAC_CT_4                               28      -10       -7       -5       70       73
    75
          ILAC_FT_P0_eEDFA0                       59       55       57       58       62       64
    65
          ILAC_FT_P0_eEDFA1                       59       55       57       58       62       64
    65
0/Rack
          SA_TEMP_AIR_INLET0                      25      -10       -5        0       45       55
    60
          SA_TEMP_AIR_INLET1                      25      -10       -5        0       45       55
```

```
    60
          SA_TEMP_AIR_EXAUST0                  27      -10       -5        0        75        85
    90
          SA_TEMP_AIR_EXAUST1                  26      -10       -5        0        75        85
    90
          SA_TEMP_PCB_HOT_SPOT0                28      -10       -5        0        80        85
    90
          SA_TEMP_PCB_HOT_SPOT1                32      -10       -5        0        80        85
    90
          SA_TEMP_PCB_HOT_SPOT2                28      -10       -5        0        80        85
    90
          SA_TEMP_PCB_HOT_SPOT3                30      -10       -5        0        80        85
    90

==============================================================================================================
Location  VOLTAGE                                Value      Crit     Minor     Minor     Crit
          Sensor                                 (mV)       (Lo)     (Lo)      (Hi)      (Hi)
--------------------------------------------------------------------------------------------------------------
0/RP0/CPU0
          RP_ADM1266_12V0                      12094    10800    11280    12720    13200
          RP_ADM1266_1V8_CPU                    1801     1670     1750     1850     1930
          RP_ADM1266_1V24_VCCREF                1238     1150     1200     1280     1330
          RP_ADM1266_1V05_CPU                   1054      980     1020     1080     1120
          RP_ADM1266_1V2_DDR_VDDQ               1207     1120     1160     1240     1280
          RP_ADM1266_1V0_VCC_RAM                 988      650      700     1250     1300
          RP_ADM1266_1V0_VNN                     858      550      600     1250     1300
          RP_ADM1266_1V0_VCCP                   1008      450      500     1250     1300
          RP_ADM1266_0V6_DDR_VTT                 603      560      580      620      640
          RP_ADM1266_3V3_STAND_BY               3310     3070     3200     3400     3530
          RP_ADM1266_5V0                        4996     4650     4850     5150     5350
          RP_ADM1266_3V3                        3328     3070     3200     3400     3530
          RP_ADM1266_2V5_PLL                    2489     2330     2430     2580     2680
          RP_ADM1266_2V5_FPGA                   2500     2330     2430     2580     2680
          RP_ADM1266_1V2_FPGA                   1197     1120     1160     1240     1280
          RP_ADM1266_3V3_CPU                    3332     3070     3200     3400     3530
          RP_ADM1266_2V5_CPU                    2502     2330     2430     2580     2680
0/Rack
          SA_ADM1266_12V_BUS_EITU              12057    10800    11280    12720    13200
          SA_ADM1266_5V0                        5022     4650     4800     5200     5350
          SA_ADM1266_1V8_ZARLINK_DPLL           1806     1670     1730     1870     1930
          SA_ADM1266_1V0_PHY                    1009      930      960     1040     1070
          SA_ADM1266_1V0_ALDRIN_CORE             982      910      930     1070     1090
          SA_ADM1266_1V0_ALDRIN_SERDES          1007      930      960     1040     1070
          SA_ADM1266_1V0_FPGA                   1008      930      960     1040     1070
          SA_ADM1266_1V2_FPGA                   1205     1120     1150     1250     1280
          SA_ADM1266_1V8                        1804     1670     1730     1870     1930
          SA_ADM1266_2V5                        2505     2330     2400     2600     2680
          SA_ADM1266_3V3                        3323     3070     3170     3430     3530
          SA_ADM1275_12V_SA_BP                 12058    10800    11280    12720    13200
          SA_ADM1275_12V_CPU_BP                12032    10800    11280    12720    13200
          SA_ADM1275_12V_MOD0_BP               12063    10800    11280    12720    13200
          SA_ADM1275_12V_MOD1_BP               12048    10800    11280    12720    13200
          SA_ADM1275_12V_MOD2_BP               12027    10800    11280    12720    13200
          SA_ADM1275_12V_FAN0_BP               12032    10800    11280    12720    13200
          SA_ADM1275_12V_FAN1_BP               12042    10800    11280    12720    13200
==============================================================================================================
Location  CURRENT                                Value
          Sensor                                 (mA)
--------------------------------------------------------------------------------------------------------------
0/RP0/CPU0
          RP_CURRMON_LTM4638                     395
          RP_CURRMON_LTM4644_0                   179
          RP_CURRMON_LTM4644_1                   307
          RP_JMAC_1V0_VCCP_IMON                  125
```

```
          RP_JMAC_1V0_VNN_IMON                    62
          RP_JMAC_1V0_VCC_RAM_IMON                 0
          RP_JMAC_1V2_DDR_VDDQ_IMON              156
0/Rack
          SA_ADM1275_12V_MOD0_IMON              3412
          SA_ADM1275_12V_MOD1_IMON                30
          SA_ADM1275_12V_MOD2_IMON                43
          SA_ADM1275_12V_FAN0_IMON              1418
          SA_ADM1275_12V_FAN1_IMON              1394
          SA_INA230_5V0_IMON                     129
          SA_INA230_3V3_IMON                    3020
          SA_INA230_1V0_XGE_CORE_IMON           2464
          SA_INA230_1V0_FPGA_CORE_IMON           787
          SA_ADM1275_12V_SA_IMON                1640
          SA_ADM1275_12V_CPU_IMON               1157
==============================================================================
                                          Fan speed (rpm)
Location      FRU Type                    FAN_0    FAN_1   FAN_2
------------------------------------------------------------------------------

0/PM0         NCS1010-AC-PSU               5424
0/FT0         NCS1010-FAN                  9960    9960    9960
0/FT1         NCS1010-FAN                 10020   10020   10020
================================================================================
Location     Altitude Value (Meters)    Source
--------------------------------------------------------------------------------
0            760                    sensor

================================================================================
CHASSIS LEVEL POWER INFO: 0
================================================================================
   Total output power capacity (Group 0 + Group 1) :    1050W +        0W
   Total output power required                     :     700W
   Total power input                               :     159W
   Total power output                              :     129W

Power Group 0:
================================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type           Volts     Amps    Volts     Amps
================================================================================
   0/PM1       NCS1010-AC-PSU  0.0       0.0     0.0       0.0      OFFLINE

Total of Group 0:               0W/0.0A         0W/0.0A

Power Group 1:
================================================================================
   Power       Supply         ------Input----   ------Output---     Status
   Module      Type           Volts     Amps    Volts     Amps
================================================================================
   0/PM0       NCS1010-AC-PSU  228.5     0.7     12.1      10.7     OK

Total of Group 1:              159W/0.7A         129W/10.7A

================================================================================
   Location     Card Type               Power       Power        Status
                                        Allocated   Used
                                        Watts       Watts
================================================================================
   0/RP0/CPU0   NCS1010-CNTLR-K9        90          14           ON
   0/FT0        NCS1010-FAN             110         17           ON
   0/FT1        NCS1010-FAN             110         16           ON
```

```
0/0/NXR0      NCS1K-ILA-C             350         54          ON
0/Rack        NCS1010-SA              40          19          ON
```

The command output shows the fan, temperature, power, voltage, current, altitude, and overall environmental status.

Verify core dump context

Use this procedure to verify core dump context.

The **show context** command displays core dump context information of NCS 1010. Core dump is a result of abnormal exit of any process running in the system.

Procedure

Verify the context.

show context

Displays the core dump context information of NCS 1010.

Example:

```
RP/0/RP0/CPU0:ios# show context
Mon Sep 27 17:21:59.219 UTC

node: node0_RP0_CPU0
------------------------------------------------------------------
No context
```

The command output is empty during system upgrade.

The verify core dump context task is complete.

Verify core files

Use this procedure to verify core files.

Use the **run** command to go to the hard disk location and check for the core dumps of NCS 1010.

Procedure

Run the shell.

run

Example:

```
RP/0/RP0/CPU0:ios# run
Mon Sep 27 17:29:11.163 UTC
```

```
[xr-vm_node0_RP0_CPU0:~]$cd /misc/disk1/
[xr-vm_node0_RP0_CPU0:/misc/disk1]$ls -lrt *.tgz
```

The verify core files task is complete.

Verify memory information

Use this procedure to verify memory information.

You can view the memory information using the show watchdog memory-state command.

Procedure

Verify memory information.

show watchdog memory-state location all

Displays memory snapshot in brief.

Example:

```
RP/0/RP0/CPU0:ios#show watchdog memory-state location all
Thu Jun 16 08:36:44.436 UTC
---- node0_RP0_CPU0 ----
Memory information:
    Physical Memory     : 31935.167 MB
    Free Memory         : 29236.0   MB
    Memory State        :   Normal
```

The verify memory information task is complete.

Cisco NCS 1010 post-setup tasks

You must create user profiles and user groups to manage your system, install software packages, and configure your network.

AAA services

Every user is authenticated using a username and a password.

The authentication, authorization, and accounting (AAA) commands help with these services:

- Create users, groups, command rules, or data rules
- Change the disaster-recovery password

User access behavior

IOS-XR and Linux have separate AAA services. IOS XR AAA is the primary AAA system.

- A user created through IOS-XR can log in directly to the EXEC prompt on the NCS 1010.

- A user created through Linux can connect to the NCS 1010 and log in to the bash prompt. The user must log in to IOS XR explicitly to access the IOS-XR EXEC prompt.

AAA authorization

Configure IOS-XR AAA authorization to restrict uncontrolled user access.

If AAA is not configured, the command rules and data rules that are associated with the assigned groups are ignored.

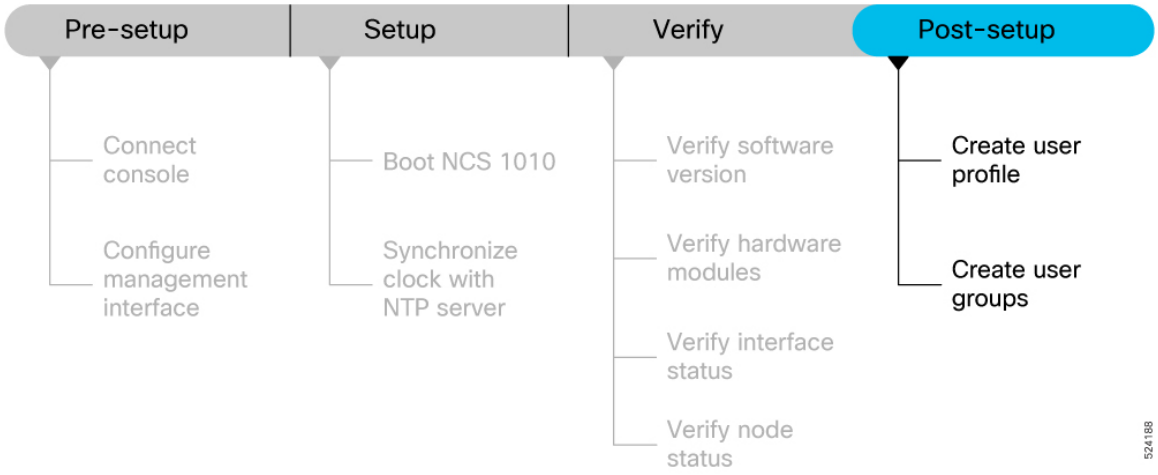
A user can then have full read and write access to IOS XR configuration through NETCONF, gRPC, or other YANG-based agents.

Enable AAA before you set up any configuration. For more information about AAA services, see AAA services.

Post-setup workflow

The image shows the tasks that are involved in the Cisco NCS 1010 Series NCS 1010 post-setup procedure.

Figure 4: Post-setup Workflow for the Cisco NCS 1010



Before you begin

Before you perform the post-setup tasks, complete these prerequisite tasks:

- Cisco NCS 1010 setup workflow
- Cisco NCS 1010 software and hardware verification tasks

Post-setup task functions

- Create a user profile: Create users and include the users in user groups with certain privileges.
- Create user groups: Associate command rules and data rules with a user group and enforce those rules on users in the group.

Create a user profile

Use this procedure to create a user profile.

You can create new users and include the user in a user group with certain privileges. The NCS 1010 supports a maximum of 1024 user profiles.

Create a user profile with these steps:

Procedure

Step 1 Create a user, provide a password, and assign the user to a group.

config

username *<user-name>*

password *password*

group root-lr

Example:

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

/* Create a new user */
ios(config)#username user1

/* Set a password for the new user */
ios(config-un)#password pw123

/* Assign the user to group root-lr */
RP/0/RP0/CPU0:ios(config-un)#group root-lr
```

All users have read privileges. The **root-lr** users inherit write privileges where users can create configurations, create new users, and so on.

Enable display of login banner: The US Department of Defense (DOD)-approved login banner provides information such as number of successful and unsuccessful login attempts, time stamp, login method, and so on.

The banner is displayed before granting access to devices and helps maintain privacy and security that is consistent with applicable federal laws.

The system tracks logins from system boot or from the time the user profile is created.

You can enable or disable the login banner by using the **login-history enable** and **login-history disable** commands.

Note
Login notifications get reset during a NCS 1010 reload.

Step 2 Verify the state of login banner.

show running-config username NAME1

Example:

```
RP/0/RP0/CPU0:ios(config-un)#show running-config username NAME1
 Fri Jan 29 13:55:28.261 UTC
 username NAME1
 group UG1
 secret * **********
```

```
password * ******
login-history enable
```

Step 3 Commit the configuration.

commit

Example:

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

The user profile is created and allowed access to the NCS 1010 based on the configured privileges.

The create a user profile task is complete.

Create user groups

Use this procedure to create user groups.

You can create a new user group to associate command rules and data rules with it. The command rules and data rules are enforced on all users that are part of the user group. The NCS 1010 supports a maximum of 32 user groups.

Before you begin

Ensure that you have created a user profile. See Create a user profile.

Procedure

Step 1 Create a new user group.

config

group group1

username user1

Example:

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

/* Create a new user group, group1 */
ios#(config)#group group1

/* Specify the name of the user, user1 to assign to this user group */
ios#(config-GRP)#username user1
```

Step 2 Commit the configuration.

commit

Example:

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

The create user groups task is complete.

What to do next

This completes the NCS 1010 setup and verification process. You can now proceed with upgrading the software, installing RPMs, SMUs and bug fixes based on your requirement.