Perform System Upgrade and Install Feature Packages

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

The topics covered in this chapter are:

Upgrade the System

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

System upgrade is done by installing a base package–Cisco IOS XR Core Bundle plus Manageability Package. The file name for this bundle is *ncs1001-mini-x-7.10.1.iso*. Install this ISO image using **install** commands. For more information about the install process, see Workflow for Install Process, on page 14.

Software Upgrade and Downgrade Matrix

The following table lists the upgrade and downgrade paths supported for Cisco NCS 1001.

Upgrade Path		Downgrade Path	
Source Release	**Destination Release**	**Source Release**	**Destination Release**
R7.3.1, R7.5.2, R7.7.1, R7.8.1, R7.9.1	R7.10.1	R7.10.1	R7.9.1, R7.8.1, R7.7.1, R7.5.2, R7.3.1

Software Compatibility Matrix

The following table describes the software compatibility for all firmware.

Table 1: Software Compatibility Matrix

FPD	R6.2.1	R6.2.2	R6.3.1	R6.3.2	R6.5.1	R6.5.2	R7.0.0	R7.0.1	R7.1.1	R7.1.2	R7.2.1	R7.3.1 and R7.3.2	R7.5.1, R7.5.2, and R7.7.1	R7.8.1, R7.9.1, and 7.10.1
FW_PSMv1	1.38	1.38	1.43	1.45	1.45	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51
FW_PSMv2	0.09	0.09	0.12	0.14	0.14	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16
FW_PSMv3														1.64 only for 7.10.1
Control_ BKP	1.07	1.09	1.09	1.09	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10
Control_ FPGA	1.07	1.09	1.09	1.09	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10
FW_EDFAv1	1.39	1.39	1.43	1.43	1.43	1.51	1.54	1.55	1.56	1.56	1.56	1.60	1.60 for R7.5.1 and 1.61 for R7.5.2 and R7.7.1	1.61
FW_EDFAv2	0.12	0.12	0.28	0.31	0.37	0.37	0.39	0.40	0.40	0.40	0.40	0.43	0.43	0.45
FW_OTDR_p	NA	NA	NA	NA	5.02	5.02	6.02	6.02	6.02	6.02	6.02	6.03	6.03	6.03
FW_OTDR_s	NA	NA	NA	NA	1.47	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51
PO-PriMCU (AC)	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.0	4.0	4.0	4.0

FPD	R6.2.1	R6.2.2	R6.3.1	R6.3.2	R6.5.1	R6.5.2	R7.0.0	R7.0.1	R7.1.1	R7.1.2	R7.2.1	R7.3.1 and R7.3.2	R7.5.1, R7.5.2, and R7.7.1	R7.8.1, R7.9.1, and 7.10.1
PO-PriMCU (DC)	NA	1.10	1.10	1.10	1.10	1.10	1.10	1.14	1.14	1.14	1.14	1.14	1.14	2.01
BIOS_ Backup	13.50	13.60	13.80	14.20	14.20	14.20	14.20	14.50	14.60	14.60	15.10	15.10	15.10	15.10
BIOS_ Primary	13.50	13.60	13.80	14.20	14.20	14.20	14.20	14.50	14.60	14.60	15.10	15.10	15.10	15.10 for 7.8.1 and 7.9.1; 15.30 for 7.10.1
Daisy_ Duke_ BKP	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.20	0.20	0.20	0.20	0.20	0.20	0.20
Daisy_ Duke_ FPGA	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.20	0.20	0.20	0.20	0.20	0.20	0.20

Upgrade the Firmware

Use the following procedure to upgrade the firmware.

Attention If you are upgrading FPDs from R7.8.1 to later releases, refer to Upgrade FPDs, on page 10 for serialized upgrade.

Note The output of this procedure is related to R7.7.1. The FPDs to be installed for later releases are different. Refer to Software Compatibility Matrix, on page 2 for FPD versions.

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

Procedure

Perform the procedures in the following sequence.

a. Upgrade BIOS and Daisy Duke FPDs, on page 4

b. Upgrade Control FPGA and Control BKP FPDs, on page 6

c. Upgrade PSM, on page 7

d. Upgrade EDFA, on page 9

Upgrade BIOS and Daisy Duke FPDs

Attention This procedure is valid only until Release 7.7.1.

From R7.0.1 and later, the upgrade procedure also updates the secure boot keys.

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

Note

- BIOS downgrade is not supported once BIOS FPD is upgraded to 15.10.
- The 15.10 BIOS FPD version does not have issues for software images prior to R7.2.1. If the user needs to downgrade the software image prior to R7.2.1, the BIOS FPDs always show the status as "NEED UPGRADE".

Before you begin

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

Procedure

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

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                        FPD Versions
                                                                      =================
Location   Card type             HWver FPD device       ATR Status   Running Programd
-----------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   NEED UPGD          1.09
0/0        NCS1001-K9            0.1   Control_FPGA         NEED UPGD  1.09    1.09
0/1        NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31    0.31
0/2        NCS1K-PSM             0.0   FW_PSMv2             NEED UPGD  0.12    0.12
0/3        NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31    0.31
0/RP0      NCS1K-CNTLR2          0.1   BIOS_Backup      BS  NEED UPGD         13.80
0/RP0      NCS1K-CNTLR2          0.1   BIOS_Primary      S  NEED UPGD 13.80   13.80
0/RP0      NCS1K-CNTLR2          0.1   Daisy_Duke_BKP   BS  NEED UPGD          0.17
0/RP0      NCS1K-CNTLR2          0.1   Daisy_Duke_FPGA   S  NEED UPGD  0.17    0.17
```

Step 2 Enter the **upgrade hw-module location 0/RP0 fpd all** command from IOS XR console to upgrade the BIOS and Daisy Duke FPDs.

Step 3 Enter the **show hw-module fpd** command again from IOS XR console to view the status of BIOS and Daisy Duke FPDs.

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                      FPD Versions
                                                                    =================
Location   Card type             HWver FPD device       ATR Status   Running Programd
-----------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   NEED UPGD          1.09
0/0        NCS1001-K9            0.1   Control_FPGA         NEED UPGD  1.09    1.09
0/1        NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31    0.31
0/2        NCS1K-PSM             0.0   FW_PSMv2             NEED UPGD  0.12    0.12
0/3        NCS1K-EDFA            0.0   FW_EDFAv2            NEED UPGD  0.31    0.31
0/RP0      NCS1K-CNTLR2          0.1   BIOS_Backup      BS  RLOAD REQ         15.10
0/RP0      NCS1K-CNTLR2          0.1   BIOS_Primary      S  RLOAD REQ 13.80   15.10
0/RP0      NCS1K-CNTLR2          0.1   Daisy_Duke_BKP   BS  CURRENT            0.20
0/RP0      NCS1K-CNTLR2          0.1   Daisy_Duke_FPGA   S  RLOAD REQ  0.17    0.20
```

Step 4 Enter the **admin** command to change to admin console.

Step 5 Enter the **hw-module location 0/RP0 reload** command to reload NCS 1001.

The system reboots within few seconds.

Step 6 Enter the **show hw-module fpd** command from the admin console to view the status of BIOS and Daisy Duke FPDs.

Example:

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

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

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

What to do next

Upgrade Control FPGA and Control BKP FPDs, on page 6

Upgrade Control FPGA and Control BKP FPDs

Attention This procedure is valid only until Release 7.7.1.

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

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

Procedure

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

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                      FPD Versions
                                                                     =================
Location   Card type             HWver FPD device       ATR Status   Running Programd
-----------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   NEED UPGD          1.09
0/0        NCS1001-K9            0.1   Control_FPGA         NEED UPGD  1.09    1.09
```

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

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

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

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                       FPD Versions
                                                                     =================
Location   Card type             HWver FPD device       ATR Status   Running Programd
-----------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   NEED UPGD          1.09
0/0        NCS1001-K9            0.1   Control_FPGA         RLOAD REQ  1.09    1.09
```

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

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

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

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                       FPD Versions
                                                                     =================
Location   Card type              HWver FPD device       ATR Status    Running Programd
------------------------------------------------------------------------------------------
0/0        NCS1001-K9             0.1   Control_BKP      B   CURRENT            1.10
0/0        NCS1001-K9             0.1   Control_FPGA         RLOAD REQ  1.09    1.10
```

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

The system reboots within few seconds.

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

Example:

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

What to do next

Upgrade PSM, on page 7

Upgrade PSM

Attention This procedure is valid only until Release 7.7.1.

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

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

Perform the following steps to upgrade PSM in section protection configuration.

Procedure

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

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                        FPD Versions
                                                                     =================
Location   Card type             HWver FPD device       ATR Status   Running Programd
-----------------------------------------------------------------------------------------
0/2        NCS1K-PSM             0.0   FW_PSMv1         NEED UPGD       a.bc   x.yz
```

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

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

Step 3 **config**

Step 4 Enter the **no hw-module location 0/RP0/CPU0 slot 2 psm lockout-from WORKING** to move the PSM state to **Ots0_2_0_1 Working Active** without any lockout in place.

Step 5 **commit**

Step 6 Enter the **hw-module slot 2 manual-switch-to WORKING** command from IOS XR console to switch the traffic manually from the protect to the working path.

PSM switch causes traffic loss for less than 50 ms.

Step 7 Enter the **show controllers ots 0/2/0/* summary** command to check the output.

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

Step 8 Enter the **show controllers ots 0/2/0/* summary** command on the far-end node to verify that **Ots0_2_0_1 Working** port is reported as **Active**.

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

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

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

Step 9 Enter the **upgrade hw module location 0/2 fpd FW_PSMv1** command from IOS XR console to upgrade the PSMv1 FPD.

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

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

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

Step 10 Enter the **show hw-module fpd** command from the admin console to view the status of PSM FPD.

Example:

```
RP/0/RP0/CPU0:ios# show hw-module fpd
                                                                       FPD Versions
                                                                     =================
Location   Card type             HWver FPD device       ATR Status   Running Programd
-----------------------------------------------------------------------------------
0/0        NCS1001-K9            0.1   Control_BKP      B   CURRENT            1.10
0/0        NCS1001-K9            0.1   Control_FPGA         CURRENT    1.10    1.10
0/1        NCS1K-EDFA            0.0   FW_EDFAv2            CURRENT    0.43    0.43
0/2        NCS1K-PSM             0.0   FW_PSMv2             CURRENT    0.16    0.16
0/3        NCS1K-EDFA            0.0   FW_EDFAv1            CURRENT    1.60    1.60
0/3        NCS1K-EDFA            0.0   FW_EDFAv1            CURRENT    1.60    1.60
```

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

What to do next

Upgrade EDFA, on page 9

Upgrade EDFA

⚠

Attention This procedure is valid only until Release 7.7.1.

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

Procedure

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

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

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

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

Example:

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

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

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

Step 3 (Only for FW_EDFAv1 FPD) Enter the **admin** command to change to admin console.

Step 4 (Only for FW_EDFAv1 FPD) Enter the **hw-module location 0/slot reload** command to perform hardware reset of the EDFA module.

Step 5 (Only for FW_EDFAv1 FPD) After the reload, check the state of EDFA modules using the **show controllers ots 0/slot/0/0** and **show controllers ots 0/slot/0/0** commands.

Example:

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

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

Upgrade FPDs

Table 2: Feature History

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

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

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

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

The FPD devices upgrade in the following sequence:

1. **Control_BKP**

   Control_FPGA

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

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

   BIOS_Primary

   Daisy_Duke_BKP

   Daisy_Duke_FPGA

Procedure

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

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

The following output shows the details of the FPD devices.

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

0/0          NCS1001-K9   0.1     Control_FPGA                 NEED UPGD     1.09

0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD     1.43
1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    NEED UPGD     6.02
6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    NEED UPGD     1.47
1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD     1.43
1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     NEED UPGD     13.80

0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      NEED UPGD     13.80
13.80
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     NEED UPGD     0.20
```

```
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      NEED UPGD     0.20
0.20
```

Step 2 To upgrade all the FPD devices, execute the following command:

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

Step 3 To check the status of the FPD devices, execute the following command:

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

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

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

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

0/0          NCS1001-K9   0.1     Control_FPGA                 RLOAD REQ     1.10

0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD     1.43
1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    NEED UPGD     6.02
6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    NEED UPGD     1.47
1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    NEED UPGD     1.43
1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     NEED UPGD     13.80

0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      NEED UPGD     13.80
13.80
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     NEED UPGD     0.20

0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      NEED UPGD     0.20
0.20
```

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

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

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

0/0          NCS1001-K9   0.1     Control_FPGA                 Current       1.10

0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    Current       1.43
1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    Current       6.02
6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    Current       1.47
1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    Current       1.43
1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     NEED UPGD     13.80

0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      NEED UPGD     13.80
13.80
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     NEED UPGD     0.20
```

```
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      NEED UPGD     0.20
0.20
```

Step 5 To see if all the FPD devices are upgraded, use the following command:

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

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

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

0/0          NCS1001-K9   0.1     Control_FPGA                 Current       1.10

0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    Current       1.43
1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    Current       6.02
6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    Current       1.47
1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    Current       1.43
1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     RLOAD REQ     15.10

0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      RLOAD REQ     15.10
15.10
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     RLOAD REQ
0.20
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_FPGA       S      RLOAD REQ     0.20
0.20
```

Step 6 To reload the BIOS and Daisy_Duke FPDs, execute the following command:

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

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

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

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

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

0/0          NCS1001-K9   0.1     Control_FPGA                 Current       1.10

0/1          NCS1K-EDFA   0.0     FW_EDFAv1                    Current       1.43
1.43
0/2          NCS1K-OTDR   0.0     FW_OTDR_p                    Current       6.02
6.02
0/2          NCS1K-OTDR   0.0     FW_OTDR_s                    Current       1.47
1.47
0/3          NCS1K-EDFA   0.0     FW_EDFAv1                    Current       1.43
1.43
0/RP0        NCS1K-CNTLR2 0.1     BIOS_Backup           BS     Current       15.10

0/RP0        NCS1K-CNTLR2 0.1     BIOS_Primary          S      Current       15.10
15.10
0/RP0        NCS1K-CNTLR2 0.1     Daisy_Duke_BKP        BS     Current
0.20
```

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

Install Packages

Packages and software patches (SMU) can be installed on NCS 1001. Installing a package on NCS 1001 installs specific features that are part of that package. Cisco IOS XR software is divided into various software packages; this enables you to select the features to run on NCS 1001. Each package contains components that perform a specific set of NCS 1001 functions.

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

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

Workflow for Install Process

To install a package, see Install Packages, on page 17. To uninstall a package, see Uninstall Packages, on page 23. The workflow for installation and uninstallation processes are depicted in individual flowcharts in their respective subsections.

Creating Repository to Access Files for Upgrading IOS XR Software

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

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

- IP address of the server

- Port number of the server

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

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

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

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

Create and Configure a Local Repository

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

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

The following process explains how to set up a local RPM repository on the router.

Procedure

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

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

Step 2 Copy the required RPMs and ISO files (using the copy or scp command) from the server to the local directory on the router.

Step 3 Access the shell of the router using run the command and untar the RPMs.

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

Step 4 Exit from the shell.

Step 5 Configure the local repository.

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

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

Create and Configure an External Repository

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

Ensure that you have completed the following tasks:

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

Procedure

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

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

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

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

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

Step 3 Configure the external repository.

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

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

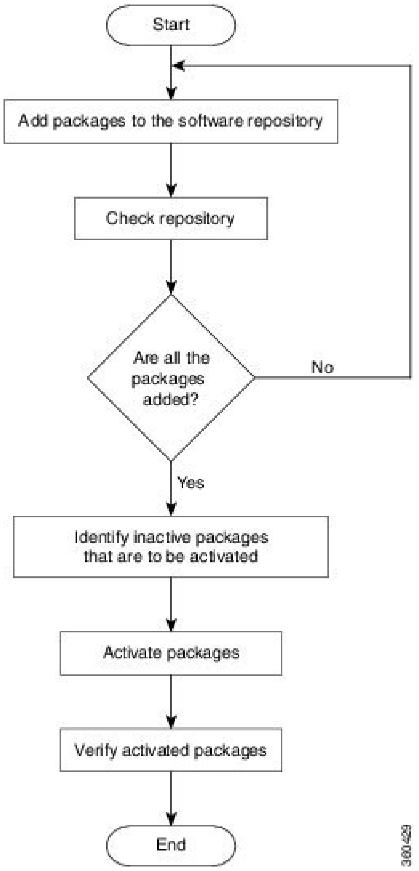
Install Packages

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

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

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

Figure 1: Installing Packages Workflow


Start
Add packages to the software repository
Check repository
Are all the packages added?
No
Yes
Identify inactive packages that are to be activated
Activate packages
Verify activated packages
End
360429

Note Disable auto-fpd upgrade before start of software upgrade.

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

Before you begin

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

Procedure

Step 1 Log into the Cisco software download site with your Cisco user ID and password.

Step 2 Click **Browse All** to see the product categories.

Step 3 Select the software either by searching across all product categories or by entering the complete product name or a partial string in the search box.

Example:

For example, NCS1001 software download page is available at https://software.cisco.com/download/home/286315098/type

Step 4 Download the Cisco IOS tar file either to FTP or TFTP server.

Step 5 Copy the tar file from the FTP or TFTP server either to router’s harddisk or external repository.

Example:

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

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

Step 6 Execute **install add** command to extract and install the packages from the tar file.

- **install add source** *<tftp transfer protocol>/ package_path/ filename1 filename2 ...*
- **install add source** *<ftp or sftp transfer protocol>// user@server:/ package_path/ filename1 filename2* ...

Example:

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

or

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


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

Note

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

Step 7 Execute **show install repository** to display packages that are added to the repository.

Example:

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

Step 8 (Optional) Execute the **show install request** command to display the operation ID of the add operation and its status. The operation ID can be used later to execute the activate command.

Example:

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


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

The install add operation 13 is 30% complete

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

Step 9 Execute **show install log** *operation ID* to display packages that are added to the repository.

Example:

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


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

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

Step 10 Execute **show install inactive** to display inactive packages that are present in the repository. Only inactive packages can be activated.

Example:

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


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

Step 11 Execute one of the following commands for the pre-activation checks and load individual components of the installable files onto the router setup.

- **install prepare**
- **install prepare id**

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

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

Step 12 Execute one of the following commands for the package activation. **install activate** *package_name*

- **Install activate** *package_name*
- **Install activate id** *operation_id*

Example:

```
RP/0/RP0/CPU0:ios#install activate ncs1001-k9sec-1.0.0.0-r7101.x86_64.rpm
ncs1001-mini-x-7.10.1.iso

Thu Jul 25 14:53:27.564 CEST

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

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

Install operation will continue in the background

RP/0/RP0/CPU0:ios#show install log 14
```

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

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

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

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

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

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

or

Using the install update command:

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

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

Step 13 Execute **show install active** to display active packages. You can verify from the result that the same image and package versions are active on all RPs and LCs

Example:

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


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

Displays packages that are active.

Step 14 Execute **install commit** to commit the Host, XR, and System Admin active software.

Example:

```
RP/0/RP0/CPU0:ios#install commit system


Thu Jul 25 17:05:27.364 CEST
Jul 25 17:05:30 Install operation 15 started by root:
  install commit system
Jul 25 17:05:31 Install operation will continue in the background

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

Note

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

Step 15 Execute **show install commit** command to display the committed packages.

Example:

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


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

What to do next

- After performing a system upgrade, upgrade FPD by using the **upgrade hw-module location all fpd all** command from the Cisco IOS XR mode. The progress of FPD upgrade process can be monitored using the **show hw-module fpd** command.
- Reload NCS 1001 if any FPD status is in RLOAD REQ state. If CTRL FPGA is in RLOAD REQ state, use the **hw-module location 0/0 reload** command. If Daisy Duke or BIOS is in RLOAD REQ state, use the **hw-module location 0/RP0 reload** command.
- Verify the installation using the **install verify packages** command.
- Uninstall the packages or SMUs if their installation causes any issues on the NCS 1001. See Uninstall Packages, on page 23.

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

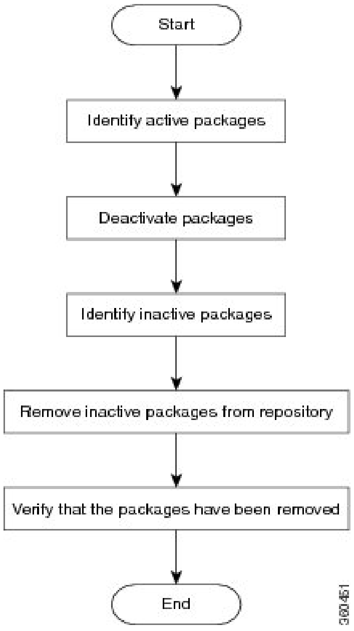
Uninstall Packages

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

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

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

Figure 2: Uninstalling Packages Workflow


Start
Identify active packages
Deactivate packages
Identify inactive packages
Remove inactive packages from repository
Verify that the packages have been removed
End
360451

Procedure

Step 1 **show install active**

Example:

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

Displays active packages. Only active packages can be deactivated.

Step 2 **show install repository**

Example:

```
show install repository


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

Step 3 Execute one of these commands:

- **install deactivate** *package_name*
- **install deactivate id** *operation_id*

Example:

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

or

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

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

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

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

Step 4 **show install inactive**

Example:

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


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

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

Step 5 **install remove** *package_name*

Example:

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


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

The inactive packages are removed from the repository.

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

Step 6 **show install repository**

Example:

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


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

Displays packages available in the repository. The package that are removed are no longer displayed in the result.

What to do next

Install required packages. See Install Packages, on page 17.