Install Cisco IOS XR7 on NCS 540 Series Routers

This section describes the concepts and procedures for upgrading or downgrading your system, installing optional packages, and obtaining bug fixes for the Cisco NCS 540 series routers.

Cisco NCS 540 series routers use the Cisco IOS XR7 framework. This framework refers to a set of architectural enhancements to the Cisco IOS XR software around the capabilities of modularity, simplified platform infrastructure, and programmability at various software layers.

The Cisco IOS XR software is composed of a base image (ISO) that provides the XR infrastructure, and a TAR file. The TAR file is made up of a set of packages (also called RPMs). These packages comprise mandatory and optional RPMs that can be deployed based on specific requirements. This software modularity approach provides a flexible consumption model that allows you to install a subset of IOS XR packages on devices based on your individual requirements. All critical components are modularized as packages so that you can select the features that you want to run on your router. For example, components like CDP and Telnet are modularized as packages and separated from the base image. These packages can be individually installed, upgraded or removed based on your requirements.

Cisco IOS XR7 install is Dandified Yum- or DNF-based software package manager that is used to install, update, and remove packages on the RPM-based Linux distributions. The package manager is used to automatically compute dependencies and determine the actions required to install packages.

Note For information on how to download the system upgrade procedures document, see the *About Cisco IOS XR Software Upgrade and Downgrade Guide*.

The Cisco NCS 540 router variants that support XR7 in Cisco IOS XR Release 7.0.1 are:

- N540-28Z4C-SYS-A
- N540-28Z4C-SYS-D
- N540X-16Z4G8Q2C-A
- N540X-16Z4G8Q2C-D
- N540-12Z20G-SYS-A
- N540-12Z20G-SYS-D
- N540X-12Z16G-SYS-A
- N540X-12Z16G-SYS-D

The Cisco NCS 540 router variants that support XR7 in Cisco IOS XR Release 7.3.1 are:

- N540X-6Z18G-SYS-A
- N540X-6Z18G-SYS-D
- N540X-8Z16G-SYS - A
- N540X-8Z16G-SYS - D

The Cisco NCS 540 router variants that support XR7 in Cisco IOS XR Release 7.4.1 are:

- N540X-4Z14G2Q-SYS-A
- N540X-4Z14G2Q-SYS-D
- N540-24Q8L2DD-SYS

The following Cisco NCS 540 Series Routers support XR7 in Cisco IOS XR Release 7.10.1:

Table 1: Cisco NCS 540 Series Router and PIDs

Cisco NCS 540 Router Type	Product Identifier
Small Density Routers	• N540X-6Z18G-SYS-A • N540X-6Z18G-SYS-D • N540X-8Z16G-SYS-A • N540X-8Z16G-SYS-D • N540X-4Z14G2Q-A • N540X-4Z14G2Q-D • N540-6Z18G-SYS-A • N540-6Z18G-SYS-D • N540-6Z14S-SYS-D
Medium Density Routers Supporting the XR7 Operating System	• N540-28Z4C-SYS-A • N540-28Z4C-SYS-D • N540X-16Z4G8Q2C-A • N540X-16Z4G8Q2C-D • N540X-16Z8Q2C-D • N540-12Z20G-SYS-A • N540-12Z20G-SYS-D • N540X-12Z16G-SYS-A • N540X-12Z16G-SYS-D

Cisco NCS 540 Router Type	Product Identifier
Large Density Routers	N540-24Q8L2DD-SYS
Fronthaul Routers	• N540-FH-CSR-SYS • N540-FH-AGG-SYS

Supported Packages

The base ISO image is contained within a `.tar` file. Additional optional packages (RPMs) are provided as modular software deliverables to align with diverse use cases and their deployments across the network.

Note You can create a golden ISO (GISO) with optional packages and bug fixes based on your requirement. Contact Cisco Support to build a GISO.

The software deliverables include:

- ISO image containing the base install image - `ncs540l-x64-7.0.1.iso`
- Tar file containing optional RPMs - `NCS540l-iosxr-7.0.1.tar`
- ZIP file for USB boot - `ncs540l-usb_boot-7.0.1.zip`

The software deliverables can be downloaded from Cisco Software Download center.

Optional Package	Included in ISO by Default
ncs540l-netflow	Yes
ncs540l-mcast	Yes
BGP	Yes
CDP	No
EIGRP	No
IPSLA	Yes
IS-IS	Yes
LLDP	Yes

Optional Package	Included in ISO by Default
MCAST	Yes
MPLS-OAM	Yes
Netflow	Yes
OSPF	Yes
Perfmgmt	Yes
RIP	No
Telnet	No
Track	Yes

Note The telnet package is not part of the ISO image. You must manually install the telnet optional package to use telnet for client or server. This applies to all packages that are not part of the ISO image.

SSH is part of the ISO image.

Install operation over IPv6 is not supported.

Supported Packages for NCS 540 Small-Density Routers

Effective Cisco IOS XR Release 7.3.1, the following variants of the Cisco NCS 540 routers form the small-density routers:

- N540X-6Z18G-SYS-A
- N540X-6Z18G-SYS-D
- N540X-8Z16G-SYS-A
- N540X-8Z16G-SYS-D

The software deliverables include:

- ISO image containing the base install image —`ncs540l-aarch64-7.3.1.iso`
- Tar file containing optional RPMs—`NCS540l-iosxr-7.3.1.tar`
- ZIP file for USB boot—`ncs540l-usb_boot-7.3.1.zip`

Software Deliverables and Terminologies

This section provides an understanding of the terms that are associated with installing the software.

- **Package:** The primary mechanism for changing the install image on a system. A package, also known as an RPM, contains the software and metadata. A package is in `.rpm` format. A package can be mandatory or optional. Mandatory packages are part of the install image and cannot be removed. Optional packages

are not required for the software to work, but can be installed to provide additional functionalities, and can be installed or removed based on requirement.

- **ISO image:** A bootable image that contains the installable files of the base operating system (OS). The image contains the IOS XR (XR7) infrastructure for fixed and distributed platforms in the form of base ISO image, mandatory RPMs. An ISO image is in `.iso` format.
- **Golden ISO (GISO):** A customizable ISO image that is built to contain preferable packages to suit diverse installation requirements. GISO can be customized to include a standard base image with the basic functional components, additional RPMs, bug fixes, and configuration files based on your requirement. GISO can also include a custom image version. From IOS XR Release 7.5.x and later, you can build your GISO image without support from Cisco by using the Build Golden ISO (GISO) Using `gisobuild.py` Tool feature.
- **Source:** A location where packages can be installed from. The source can be a repository, local directory or a local tar file.
- **Repository:** A directory of RPMs and their metadata that a package manager uses to query the packages.
- **Active package:** A package whose software is currently running on the system.
- **Committed package:** A package that is committed and remains active following a system reload.
- **Atomic Change:** Every packaging operation is contained within an atomic change. Atomic changes may contain multiple packaging operations. During an atomic change, any changes to install IOS XR software will not be visible to the system. To make the changes visible to the system, the atomic change must be applied.
- **Top-level package:** Each block of software has a top-level package and various partition-level packages. The top-level package can be installed or upgraded directly, whereas the partition-level packages cannot be changed directly. The partition-level packages are installed or upgraded automatically as dependencies of the top-level package. The top-level package has the name format `xr-<feature>-<release>.x86_64.rpm`, whereas the dependent partition-level packages have the longer name format containing information about the partition. You can also use the standard RPM commands to check the summary or description metadata of the package, which will identify whether it is a top-level or a partition-level package.
- **Package manager:** An entity that handles the semantics to resolve dependencies in packaging operations.
- **Packaging operations:** The actions performed to change the packages that are installed on the system. The semantics are inherited from the underlying package manager. Examples of packaging operations are upgrade, downgrade, replace, add, or remove packages.
- **Synchronous action:** Synchronous action requests are supported for install actions using CLI command. Specify `synchronous` keyword in the install commands, and the prompt will only be returned when either the request has completed, `Ctrl + C` keys are pressed or a reload occurs. Pressing `Ctrl + C` keys during a synchronous action request will return the prompt to the user but will not halt the install operation. During the synchronous action request, the user is updated with the status of the request whenever it changes.
- **Transaction:** All atomic changes occur within a transaction. If the system reloads during an install transaction, the running software will be reverted to its previous state before the transaction was started. To maintain the software changes carried out during a transaction, you must commit the transaction.
- A complete install operation to modify the system's software requires three phases:
    - Packaging operation

- **Apply:** This is required to complete an atomic change and make the software change visible to the system.
- **Commit:** This is required to end a transaction and ensure that all software changes will still be present on router reload.

Note If you perform a manual or automatic system reload without completing the transaction with the **install commit** command, 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'.

Note In a multinode system, any node reloads that occur during a transaction that are not initiated as part of the install 'apply by reload' phase can result in the reloaded node being in BOOT HOLD state. The node continues to be in the BOOT HOLD state until the transaction is either committed or cancelled.

Workflow for Installing Cisco IOS XR7 Software

The router is shipped with a pre-installed version of the Cisco IOS XR (XR7) software. When the router is powered ON for the first time, the pre-installed software starts functioning automatically. You configure the router for network capabilities. When a new version of the software is available, you can upgrade the system using these tasks:

Note For instructions to upgrade image-specific software, navigate to the CCO Software Download portal, select the product and refer to the `ncs540-x64-<version>.docs.tar` file for the release.

Obtain Data Models for Install Operation

You can use YANG data models to install and upgrade XR7 software. The data models are packaged with the release image in the `/pkg/yang` directory.

The models are in the `.yang` format. Each data model can be identified as one of the following functionalities:

- `-oper` in the model name indicates an operational model. For example, `Cisco-IOS-XR-install-oper.yang` and `Cisco-IOS-XR-install-augmented-oper.yang` are operational models for the install operation.
- `-cfg` indicates a configuration model. For example, `Cisco-IOS-XR-install-cfg.yang` is a configuration model for the install operation.

- `-act` indicates an action model. For example, `Cisco-IOS-XR-install-augmented-act.yang` and `Cisco-IOS-XR-install-act.yang` are action models for the install operation.

Procedure

Step 1 View the install-related data models on the router. You can also access the supported data models to install Cisco IOS XR software from the Github repository.

Example:

```
node0_RP0_CPU0:/pkg/yang]$ls -ltr *install*
-rw-r--r--. 1 root root  8646 Jul  2 01:59 Cisco-IOS-XR-install-act.yang
-rw-r--r--. 1 root root  7267 Jul  2 01:59 Cisco-IOS-XR-install-search-act.yang
-rw-r--r--. 1 root root 10664 Jul  2 01:59 Cisco-IOS-XR-install-augmented-act.yang
-rw-r--r--. 1 root root  2511 Jul  2 02:00 Cisco-IOS-XR-um-install-cfg.yang
-rw-r--r--. 1 root root  2270 Jul  2 02:04 Cisco-IOS-XR-install-cfg.yang
-rw-r--r--. 1 root root  6222 Jul  2 02:04 Cisco-IOS-XR-install-oper.yang
-rw-r--r--. 1 root root 14009 Jul  2 02:04 Cisco-IOS-XR-install-augmented-oper.yang
-rw-r--r--. 1 root root 13726 Jul  2 02:04 Cisco-IOS-XR-install-oper-sub1.yang
-rw-r--r--. 1 root root  2440 Jul  2 02:04 Cisco-IOS-XR-install-augmented-oper-sub2.yang
-rw-r--r--. 1 root root 59866 Jul  2 02:04 Cisco-IOS-XR-install-augmented-oper-sub1.yang
```

The following table describes the function of the install-related data models:

Data Model	Description
Cisco-IOS-XR-um-install-cfg model	Unified data model that contains a collection of YANG definitions for Cisco IOS XR install package configuration, and augments the modules with configuration data
Cisco-IOS-XR-install-oper	Operational data model to view details that are related to basic package information, active and committed packages, and fixes
Cisco-IOS-XR-install-cfg	Configuration data model to specify the location of the install source
Cisco-IOS-XR-install-act	Action model to perform basic install operations and software upgrade
Cisco-IOS-XR-install-search-act	Action model that contains a collection of YANG definitions for install actions related to searching for package information
Cisco-IOS-XR-install-augmented-oper	Augmented operational model that displays information about packaging, atomic changes, and history of the install operation on the router
Cisco-IOS-XR-install-augmented-act	Action model to perform flexible install operations, including controlling the exact timing of system reloads and rolling back to a previous commit
Cisco-IOS-XR-shellutil-copy-act	Action model to copy files on the router from a source location

Step 2 Use a manageability agent like NETCONF or gRPC to connect and communicate with the router. You can send Remote Procedure Calls (RPC) requests to configure or retrieve operational data from the router. The router processes the request and responds to the request through an RPC response. You use the RPCs to send requests to install the software by populating the relevant container and leaf in the data model.

For more information about understanding the data model structure and using data models, see the *Programmability Configuration Guide for Cisco NCS 540 Series Routers*.

Create Repository to Access Files for Installing IOS XR Software

Note If only Golden ISO (GISO) is used, you do not need to create a repository.

To install packages (RPM), code upgrades, and updates in XR7, you need a repository of RPMs for the router to download the RPMs for installation. The repository can be local to the router, or accessed remotely through FTP, HTTP, or HTTPS.

Important The repository must be created specific to each platform and release. Do not create repositories with a mix of platforms and releases.

When the repository is accessed 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
- (Optional) Virtual Routing and Forwarding (VRF) name

The repository can be configured to be reachable using a non-default VRF table. If the repository is reachable through an address in a VRF, specify the name of the VRF.

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

- FTP: `ftp://<server>[;<vrf>]/<path-to-repository>`
- HTTP: `http://<server>[;<vrf>]/<path-to-repository>`
- HTTPS: `https://<server>[;<vrf>]/<path-to-repository>`
- Local: `file:///<path-to-repository>`. 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 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.

This section provides the procedure for setting up a local RPM repository on the router.

Procedure

Step 1 Create a directory locally on the router's `/harddisk:`. Copy the required RPMs and ISO files (using copy or scp command) from the server to the local directory on the router.

Step 2 Access the shell of the router using `run` command and untar the RPMs.

Example:

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

Step 3 Exit from the shell.

Step 4 Configure the local repository.

Example:

```
Router#config
Router(config)#install repository local-repo url file:///harddisk:/<directory-with-rpms>
Router(config)#commit
<data and time stamp> UTC: config[67543]: Configuration committed by user.
Router(config)#end
```

where, `local-repo` is the repository name, `file:///harddisk:/<directory-with-rpms>` is the local repository URL.

Step 5 Check the contents of the repository.

Example:

```
Router#show install available
Trying to access repositories...
Package                Architecture          Version          Repository
--------------------  ---------------  ------------------ ---------------

xr-ncs540l-core       x86_64        7.0.1v1.0.1-1      local-repo

xr-core               x86_64        7.0.1v1.0.1-1      local-repo
```

Note
Only the top-level packages are displayed. The contents of the repository is 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 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, updates) are made available. This is the recommended method to setup a repository.

Note For release 7.0.1, 7.0.2, and 7.1.x, the external repository is available only through the Management Ethernet interface.

Before you begin

Ensure that you have completed the following tasks:

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

Procedure

Step 1 Create a 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 using `createrepo` utility on the Linux server. This creates a directory named `repodata` with the metadata of all the RPMs.

Example:

```
[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.

Example:

```
Router#config
Router(config)#install repository remote-repo url http://10.194.88.104/<directory-with-rpms>

Router(config)#commit
<data and time stamp> UTC: config[67542]: Configuration committed by user 'cisco'.
Router(config)#end
```

For FTP, the repository is configured as follows:

```
Router#config
Router(config)#install repository remote-repo url ftp://10.194.88.104/<directory-with-rpms>
Router(config)#commit
<data and time stamp> UTC: config[67543]: Configuration committed by user 'cisco'.
Router(config)#end
```

where, `remote-repo` is the repository name, `http://10.194.88.104/<directory-with-rpms>` is the HTTP repository URL, and `ftp://10.194.88.104/<directory-with-rpms>` is the FTP repository URL.

Step 4 Verify connectivity to the server, and check the contents of the repository.

Example:

```
Router#show install available
Trying to access repositories...
Package                 Architecture          Version          Repository
--------------------  ---------------  ------------------ ---------------

xr-ncs540l-core       x86_64        7.0.1v1.0.1-1      remote-repo

xr-core               x86_64        7.0.1v1.0.1-1      remote-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 with the updated metadata are present in the repository. It displays only the packages that are available in the repository and not part of active system.

Upgrade the Current Active Version of Cisco IOS XR Software

This section shows replacing the current software version with `.iso` image. The instructions in this section also apply to system downgrade.

Upgrade the Current Active Version

In this scenario, you replace the current software with image, apply the changes, and commit the install operation. Committing the changes indicates the end of the current transaction. The updated software is used after the changes are applied, before the install transaction is committed.

A reboot is not always necessary. Bug fixes can be applied using a GISO that contains the current running software and the additional bugfix. These fixes may not require a reboot. Reboots are required for a system version change, if too many processes need restarting, there is configuration in the GISO to apply or if the bugfix is marked as requiring a reload.

From Cisco IOS XR Release 7.5.1, you can use the Automatic Install of Bridging Bug Fix RPMs feature to install any bridging bug fix RPMs that are prerequisite for a system upgrade or a downgrade.

You need to add the required Bridging Bug Fix RPMs into the customized ISO built using Cisco Golden ISO (GISO) build script **gisobuild.py**, by going through the steps in Build a Golden ISO, on page 23. Ensure that you add the required bridging bug fix RPMs into your Golden ISO. The GISO can include bridging Bug Fix RPMs for multiple releases, and installs only the specific bridging Bug Fix RPMs required for the target release. The bridging bug fix RPMs can be used in the following scenarios:

- To resolve a bug that might stop upgrade.
- The latest version has new prerequisite requirements that are not met by the earlier version.

The **install replace** command adds and applies the highest version of any bridging bugfix RPMs for the currently active XR version.

Cisco IOS XR Software Release 7.5.1 introduces **iso-config [ignore | replace]** keywords to the **install replace** and **install package replace** commands. Before this release, the configuration file in GISO is ignored by default. If the GISO has a configuration file and it is not ignored using the **iso-config ignore** option, then the configuration in the file is applied, and the install operation requires a reboot. With the **iso-config** keyword, the default value is set to **replace** to replace the running configuration with the configuration in the file.

You can extract the configuration file from GISO using `image.py` script that is embedded in the ISO. This script interprets the GISO, and serves as an API to the ISO content at the filesystem level. The extracted configuration file replaces the running configuration after the router reloads.

Table 2: Configuration File in GISO

Scenario	iso-config replace	iso-config ignore	Nothing Specified
Configuration is present in GISO	Extracts the configuration and replaces the running configuration after the changes are applied via router reload. The output of the **show install log** command displays `ISO config replace requested (ISO config present).`	Does not extract or apply the configuration. The output of the **show install log** command does not display any information that is related to the configuration.	Extracts the configuration and replaces the running configuration after the changes are applied via router reload. The behavior is similar to **iso-config replace** operation.
Configuration is not present in GISO	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The output of the **show install log** command displays `ISO config replace requested (ISO config not present).`	Does not extract or apply the configuration. The output of the **show install log** command does not display any information that is related to the configuration.	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The behavior is similar to **iso-config replace** operation.

Scenario	iso-config replace	iso-config ignore	Nothing Specified
Configuration extraction is not supported in target GISO (downgrade)	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The output of the **show install log** command does not display any information that is related to the configuration.	Does not extract or apply the configuration. The output of the **show install log** command does not display any information that is related to the configuration.	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The behavior is similar to **iso-config replace** operation.
Configuration is supported but the extraction fails	Attempts to extract the configuration. But the operation fails with error `Failed to extract the gISO config file from the ISO: <path to ISO>`.	Does not extract or apply the configuration. The output of the **show install log** command does not display any information related to the configuration.	The behavior is similar to **iso-config replace** operation.

Procedure

Step 1 Copy the ISO (or GISO) image to the `/harddisk:` location on the router.

Step 2 Upgrade the system to replace the current software with the .iso image.

Example:

```
Router#install package replace /harddisk:/ncs540l-x64-x.x.x.iso
```

Step 3 Activate the new .iso image on the router by applying the changes.

Example:

```
Router#install apply {reload | restart} [noprompt]
```

If you are using a configuration file in GISO, use the **install apply reload** command for the configuration to take effect.

Note
You can use a single command to perform both the packaging operation and activating the applying the changes using **install replace /harddisk:/ncs540l-x64-x.x.x.iso noprompt** command.

To identify whether a reload is required or only process restart is needed, use either **show install history last transaction verbose** command or **show install request** command.

Include the keyword `noprompt` in the command to enable the system to bypass your permission to reload the router.

Applying the change gives you the flexibility to test the operation of the new software before committing the changes. If you reload the router, the router reverts the software to its previous software state.

All operations that automatically apply the new software are prohibited when an atomic change is already in progress. You must address the current atomic-change before performing this operation. To address the change, apply the current atomic-change, or cancel it with the **install package abort all-since-apply** command.

Step 4 View the install log.

Example:

```
Router#show install log detail
Fri Nov 12 09:43:49.702 UTC
2021-11-12 09:33:47 UTC    Transaction 1 started
2021-11-12 09:33:47 UTC    Atomic change 1.1 started
2021-11-12 09:33:47 UTC    Packaging operation 1.1.1 started
2021-11-12 09:33:47 UTC    Replace


2021-11-12 09:35:58 UTC    Packaging operation 1.1.1 success
2021-11-12 09:36:04 UTC    Apply by reload started
2021-11-12 09:38:48 UTC    Atomic change 1.1 successfully applied by reload
```

If configuration file is present in GISO, the output shows that the replace operation for ISO configuration is requested.

Step 5 Verify that the image is activated successfully.

Example:

```
Router#show install request
```

Step 6 Commit the transaction.

Example:

```
Router#install commit
```

Note
Any action requests may be run synchronously from the CLI. During this request, you are updated with the status of the request whenever it changes. The following example shows the output from a synchronous action request:

```
Router#install commit synchronous
Starting:
  install commit
Transaction 1
The install operation will continue in the background
Press Ctrl-C to return to the exec prompt. This will not cancel the install operation

Current activity: Initializing ....
Current activity: Commit transaction ......

Transaction 4: 'install commit' completed without error
```

Upgrade the System and Install RPMs

In this scenario, you replace the current software with the `.iso` image and have the possibility to install or remove optional RPMs before applying the changes. You can perform this operation while an atomic-change is already in progress. However, all packaging operations before this command are discarded. The installed software is an exact copy of the software in the ISO after this packaging operation is complete. You can

perform all additional packaging operations after this operation and before applying and committing the changes.

Procedure

Step 1 Copy the ISO (or GISO) image to the `/harddisk:` directory on the router.

Step 2 Upgrade the system to replace the current software with the .iso image.

Example:

```
Router#install package replace /harddisk:/ncs540l-x64-x-x-x.iso
```

Step 3 Install other RPMs (packages) after the system upgrade operation.

a) Configure a repository on the router. For instructions to create a local or a remote repository, see Create Repository to Access Files for Installing IOS XR Software, on page 8.

b) Check the available packages in the repository.

Example:

```
Router#show install available
```

c) Install the RPMs.

Example:

```
Router#install package add <pkg1> <pkg2> <pkgn>
```

Step 4 Check the status of install operation.

Example:

```
Router#show install request

User request: install package add xr-bgp

State: In progress since <date and timestamp>

Current activity: Package add or other package operation
Next activity: Await user input
Time started: <date and timestamp>
Timeout in: 35m 8s
Locations responded: 0/1

Location    Packaging operation stage  Notification Phase Clients responded
----------  ------------------------- ------------------- ------------------
0/RP0/CPU0   Package operations         None in progress    N/A
```

Note
The operation ID is a unique ID for each user request. This ID is constructed from the transaction ID, atomic change ID and packaging operation ID that was already used in the commands. For example, if the request is `install commit`, the operation ID is the transaction ID. If the request includes applying an atomic change but not committing the transaction (for example, `install replace /harddisk:/ncs540_x64.iso`), the operation ID is the atomic change ID. An operation ID of 4.2 indicates a second atomic change in the fourth transaction.

This operation ID is also returned in the action RPC. If an error occurs while the request is initiated, an empty string is returned instead of an operation ID.

When the `State` changes to `Success`, activate the new image.

```
Router#show install request
Wed Sep 14 02:53:21.525 PDT

User request: install package abort latest
Operation ID: 2.1.2
State:        Failure since 2022-09-14 02:48:15 UTC-07:00
Disk space check failed on nodes: 0/0/CPU0. Query 'show install history id 2.1.2 errors'
for more details and next steps

Current activity:    Await user input
Time started:        2022-09-14 02:48:20 UTC-07:00

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

Note
The **install apply restart** method has the least impact.

Step 5 Activate the new .iso image or RPM on the router by applying the changes.

Example:

```
Router#install apply {reload | restart} [noprompt]
```

To identify whether a reload is required or only process restart is needed, use either **show install history last transaction verbose** command or **show install request** command.

Include the keyword `noprompt` in the command to enable the system to bypass your permission to reload the router.

Step 6 Verify the image and packages that are activated as part of `install package add` operation is activated successfully.

Example:

```
Router#show install request
```

Step 7 Commit the transaction.

Example:

```
Router#install commit
```

To perform the same step using data models, use the `install-package-replace` RPC on the Cisco-IOS-XR-install-augmented-act data model.

```
<install-replace>
   <file>iso-name</file>
   <source-type>local</source-type>
   <source>directory-containing-iso</source>
   <commit>true</commit>
</install-replace>
```

Upgrade QDD Optical Modules Through CLI

The QDD optics firmware file needs to be copied to the router manually. Contact Cisco Support to check the QDD firmware version, IOS XR release compatibility, and to obtain the QDD optics firmware file.

Limitation: When ports share a common management interface, IOS XR serializes the firmware upgrade. Serializing and deserializing may delay the upgrade process.

Procedure

Step 1 From the router, copy the QDD firmware file to the hard disk using the following command:

Example: `scp user@10.1.1.1:/home/user/filename harddisk:/`

- When you are using VRF, use the following sample command:

```
scp user@10.1.1.1:/home/user/cl1.bin vrf MGMT harddisk:/

Tue Jan 25 02:57:22.762 UTC
Connecting to 10.1.1.1...
Password:
  Transferred 1484800 Bytes
  1484800 bytes copied in 0 sec (22161194)bytes/sec

RP/0/RP0/CPU0:8808#dir harddisk:/cl1.bin
Tue Jan 25 03:00:47.835 UTC

Directory of harddisk:/cl1.bin
35 -rw-r--r--. 1 1484800 Jan 25 02:57 dp04qsdd_dp04sfp8_161_10_01.ackit

53461500 kbytes total (42983204 kbytes free)
```

- When you are not using VRF, remove the `vrf MGMT` command:

```
scp user@10.1.1.1:/home/user/cl1.bin harddisk:/
```

Step 2 Run the following commands to upgrade the FPD for QDD optical modules:

Multiport upgrade:`upgrade optics port 0,1,2,3,4 filename /harddisk:/cl1.bin location 0/1/CPU0`

Single port upgrade: `upgrade optics port 0 filename /harddisk:/cl1.bin location 0/1/CPU0`

You can check the firmware upgrade progress using the following command: `show optics firmware upgrade port 0,1,1,2,3,4 location 0/1/CPU0`

Install Optional Packages to Provide Additional Functionality

You can install one or more packages (RPM) that are not already present on the system. The packages are not mandatory for the software to function, but provide additional functionality. Based on your requirement, you can install or remove these optional packages. The source file can be a repository name, repository url, local filepath, or path to a tar file.

You must specify only the top-level package name that you want to install. The associated dependencies of this package, in the form of card and partition-specific packages, are included automatically. By default, the

latest available version of each package is installed. You can also explicitly install a specific version of a package.

Note All Cisco IOS XR images are signed to ensure the authenticity of the software.

This example shows the options to install the optional package `xr-telnet-7.0.1v1.0.1-1.x86_64.rpm`.

Before you begin

If you are installing the packages from a local directory, ensure that the TAR file `ncs540l-iosxr-7.0.1.tar` is copied to the `harddisk:/` on the router. If you are installing the packages from an RPM repository, ensure you have configured the repository. For more information, see Create Repository to Access Files for Installing IOS XR Software, on page 8.

Procedure

Step 1 Install one or more optional packages using one of the following options:

- **Option 1:** Install the package from the local directory:

```
Router# install source /harddisk:/files xr-telnet-7.0.1
v1.0.1-1.x86_64.rpm
```

Note
The `install source` command automatically applies the changes. Use this command to install optional packages. To upgrade existing packages, see Upgrade the System to Obtain Bug Fixes, on page 30.

- **Option 2:** Install the package from a configured remote repository:

```
Router#install source install-repo xr-telnet
```

Here, `install-repo` is the name of the repository. For repository configuration, see Create Repository to Access Files for Installing IOS XR Software, on page 8.

- **Option 3:** Install the package from a repository URL:

```
Router#install source http://72.16.0.0:3333/remote-repo xr-telnet
```

- **Option 4:** Add the package and apply the change. The package must be available in the repository.

```
Router#install package add xr-telnet-7.0.1v1.0.1-1.x86_64.rpm
Router#install apply {restart | reload}
```

More than one package can be installed using a single packaging operation. Use the following command:

```
Router#install source <path-to-source> <package 1> <package 2> … <package n>
```

For example,

```
Router#install source /harddisk:/files xr-telnet-7.0.1v1.0.0-1.x86_64.rpm
xr-mcast-7.0.1v1.0.0-1.x86_64.rpm
```

To perform this task using data models, use the `install` RPC on the Cisco-IOS-XR-install-act data model. Here is an example with an HTTP repository:

```
<install>
   <packages>
```

```
      <packagename>pkg1</packagename>
      <packagename>pkg2</packagename>
      …
      <packagename>pkgn</packagename>
   </packages>
   <source-type>http</source-type>
   <source><path-to-source></source>
</install>
```

Step 2 Commit the operation.

Example:

```
Router#install commit
```

Step 3 Check the status of install operation.

Example:

Delete Optional Packages

You can remove optional packages that you no longer require. An optional package is not mandatory for the operating system to function, and based on your requirement, it can be installed or removed.

Procedure

Step 1 Remove the optional package.

Example:

```
Router#install package remove <optional-package-name>
```

Step 2 Apply the changes to make the change active.

Example:

```
Router#install apply [reload | restart]
```

Attention
To identify whether to reload or restart the system after applying the changes, use either **show install history last transaction verbose** command or **show install request** command.

Step 3 Commit the changes to make the change persistent after a reload operation.

Example:

```
Router#install commit
```

Installing the Image Over the Network

Table 3: Feature History Table

Feature Name	Release Information	Feature Description
Support for loading an image over the network	Release 7.4.1	There is no longer a need to copy an ISO image on the router to install XR7. Starting with this release, you can fetch an ISO or tar file of RPMs over the network as part of Install operation. You can also directly install the image from an FTP or HTTP server that hosts the image. The install source can also be a URL to a tar file (a file with extensions, .tar, .tgz, or .tar.gz). This feature modifies the following commands: • **install replace** • **install package replace** • **install source** • **install package add source**

The following commands include the **ftp** and **http** keywords to enable you to install an ISO image stored at the specified FTP or HTTP server.

- The **install replace** command is used to install an ISO image and apply this new installation.

Note This command requires a manual reload of the system.

```
RP/0/RP0/CPU0:ios#install replace ?
WORD Full local path to ISO - must be located in or under /var/xr/disk1/, /harddisk:/,
 or /misc/disk1/
ftp: Remote ISO ftp://<server>[;<vrf>]/<remote_path>
http: Remote ISO http://<server>[;<vrf>]/<remote_path>
```

- The **install package replace** command is used to install an ISO image; however, it does not automatically apply the software change.

```
RP/0/RP0/CPU0:ios#install package replace ?
  WORD   Full local path to ISO - must be located in or under /var/xr/disk1/, /harddisk:/,
 or /misc/disk1/
  ftp:   Remote ISO ftp://<server>[;<vrf>]/<remote_path>
  http:  Remote ISO http://<server>[;<vrf>]/<remote_path>
```

- The **install source** command is used to install and apply packages from a specified source. Effective Cisco IOS XR Release 7.4.1, you can use this command to install packages from a tarball located on a remote FTP or HTTP server.

```
RP/0/RP0/CPU0:ios#install source ?
  WORD            Configured DNF repository name to use as the source
  WORD            Full path to a local directory to use as the source - must be a
subdirectory of /var/xr/disk1/, /harddisk:/, or /misc/disk1/
  WORD            Full path to a local tar file to use as the source - must be located
 in or under /var/xr/disk1/, /harddisk:/, or /misc/disk1/
  any-configured  Use any configured repository to obtain packages
  ftp:            Remote repo or tar file ftp://<server>[;<vrf>]/<remote_path>
  http:           Remote repo or tar file http://<server>[;<vrf>]/<remote_path>
```

- The **install package add source** command is used to install packages; however, it does not automatically apply them. Effective Cisco IOS XR Release 7.4.1, you can use this command to install packages from a tarball located on a remote FTP or HTTP server.

Additional Install Operations

After you upgrade your system, based on your requirement, you can perform additional install operations:

View the Version of Installed Packages

The router is shipped with a pre-installed operating system. You can view the version of the installed software and the active packages. If you have upgraded your system, installed additional packages or bug fixes, you can view the version of the committed packages.

Review the software version information:

- Package name and version
- User who built the package
- Time the package was built
- Build workspace
- Build host
- ISO label:
  - Label is present if GISO boots using PXE boot
  - Label is present if GISO is installed using the `install replace` method
  - Label reverts to default (only release version) if there is any change since the time the image with the label was installed.
  - Label is nullified and reverts to default if an RPM is added or removed on top of an existing GISO.
  - Label is repopulated if an RPM from the GISO is added or removed and a rollback operation is performed.
- Copyright information

- Hardware information

Procedure

Step 1 View the version of the Cisco IOS XR software, and its various software components that are installed on the router.

Example:

The following example shows the version information for a non-GISO image:

```
Router#show version
Cisco IOS XR Software, Version 7.0.1 LNT
Copyright (c) 2013-2019 by Cisco Systems, Inc.

Build Information:
 Built By     : xyz
 Built On     : Sat Jun 29 22:45:27 2019
 Build Host   : iox-lnx-064
 Workspace    : ../7.0.1
                7.3.1/
                NCS540L/ws/
 Version      : 7.0.1
 Label        : 7.0.1

cisco NCS540L
System uptime is 41 minutes
```

The following example shows the version information for a GISO image. The customer label is appended to the `Label` field in the GISO image:

```
Router#show version
Cisco IOS XR Software, Version 7.0.1 LNT
Copyright (c) 2013-2019 by Cisco Systems, Inc.

Build Information:
 Built By     : xyz
 Built On     : Sat Jun 29 22:45:27 2019
 Build Host   : iox-lnx-064
 Workspace    : ../7.0.1
                /NCS540L/ws/
 Version      : 7.0.1
 Label        : 7.0.1-CUSTOMER_LABEL

cisco NCS540L
System uptime is 41 minutes
```

You can also use the `get` RPC on the `install.version` data model.

Step 2 View the active packages.

Example:

```
Router#show install active [summary]
Active Packages:    XR: 112    All: 1088
Label:              7.0.1

Optional Packages                          Version
-------------------------           ----------------------
xr-ncs540l-mcast                        7.0.1v1.0.0-1
xr-ncs540l-netflow                      7.0.1v1.0.0-1
```

```
xr-bgp                                   7.0.1v1.0.0-1
xr-ipsla                                 7.0.1v1.0.0-1
xr-is-is                                 7.0.1v1.0.0-1
xr-lldp                                  7.0.1v1.0.0-1
xr-mcast                                 7.0.1v1.0.0-1
xr-mpls-oam                              7.0.1v1.0.0-1
xr-netflow                               7.0.1v1.0.0-1
xr-ospf                                  7.0.1v1.0.0-1
xr-perfmgmt                              7.0.1v1.0.0-1
xr-track                                 7.0.1v1.0.0-1
```

You can also use the `get` RPC on the `install.packages.active.node` and the `install.packages.active.node.summary` data models.

To understand the data model structure and its arguments, see the Obtain Data Models for Install Operation, on page 6 topic.

Build a Golden ISO

Table 4: Feature History Table

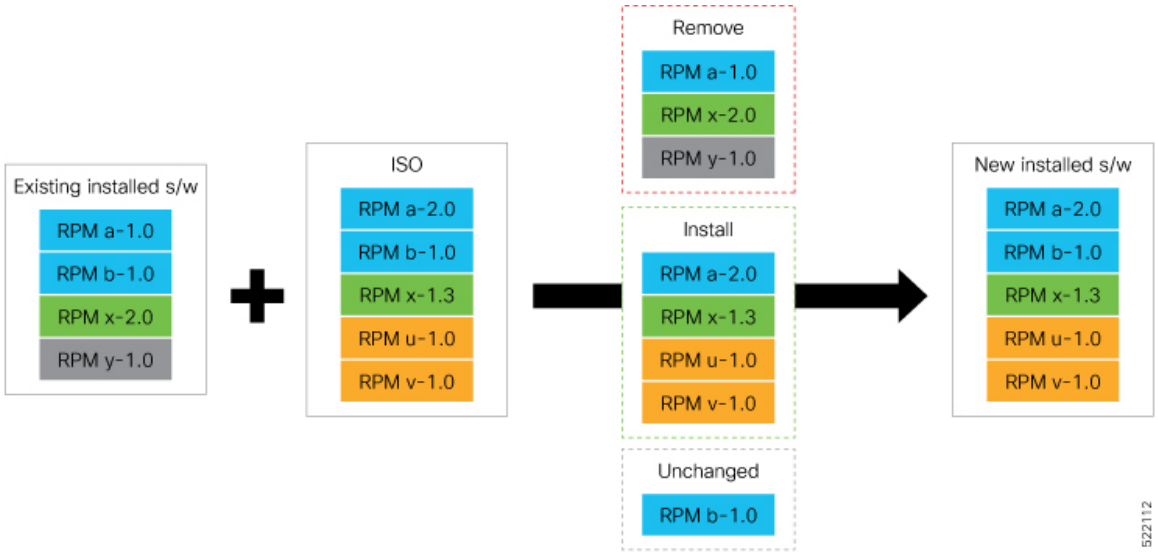
Feature Name	Release Information	Description
Build Golden ISO (GISO) Using `gisobuild.py` Tool	Release 7.6.1	This feature allows you to build your GISO image without support from Cisco. You can now select the install files, add your RPMs, repackage them as a custom image, and install the image. In previous releases, you had to contact Cisco to get your GISO built.

Golden ISO (ISO) upgrades the router to a version that has a predefined set of RPMs with a single operation. For example, you can create a customized ISO with the base OS package and specific optional RPMs based on your network requirements.

GISO supports automatic dependency management, and provides these functionalities:

- Builds RPM database of all the packages present in package repository.
- Skips and removes Cisco RPMs that do not match the base ISO version.
- Skips and removes third-party RPMs that are not part of already existing third-party base package in the base ISO.

Note Install operation over IPv6 is not supported.



Procedure

Step 1 Contact Cisco Support to build the GISO image with the set of packages based on your requirement.

Step 2 Build GISO image using `gisobuild.py` tool.

To build GISO, provide the following input parameters to the script:

- Base mini-x.iso (mandatory)
- Set of packages to install (Cisco signed packages)
- XR configuration file (optional)
- Label for golden ISO (optional)
- ZTP initialization `ztp.ini` file (optional)

Note
GISO build tool verifies the RPM dependecnies and RPM signatures. GISO build fails if the RPM is unsigned or incorrectly signed.

a) Copy the repository from the Github location to an offline system or external server where the GISO will be built.

b) Run the script `gisobuild.py` and provide parameters to build the GISO image. Ensure that all RPMs and SMUs are present in the same directory or on a repository.

Example:

```
$ ./giso/src/gisobuild.py --iso <input iso> --repo <rpm repo1 rpm_repo2> --pkglist <pkg1
 pkg2 pkg3>
    --xrconfig <config.cfg> --ztp-ini <ztp.ini> --label <label>
    --out-directory <out_directory> --clean
```

The following parameters can be provided as input to the GISO build tool:

- --iso: ISO path to mini.iso or full.iso file

- --xrconfig: XR configuration file
- --label: GISO label
- --repo: Path to repositories containing RPMs and tarballs
- --pkglist: Optional RPMs or SMUs to package
- --ztp-ini: Path to the ZTP initialization file
- --remove-packages: Remove RPMs from the GISO. To remove multiple RPMs, separate the RPM names using comma. For example, `--remove-packages xr-bgp,xr-mcast` command removes the xr-bgp and xr-mast packages from GISO
- --out-directory: Output directory to store output of the operations performed on the file
- --clean: Delete contents of the output directory
- --skip-dep-check: Skip dependency checking between files
- --version: Print version of the tool
- --pkglist: Optional RPM or SMU to package
- --yamlfile: Provide CLI arguments via YAML markup file
- --docker: Load and run pre-built docker image

The tool uses the input parameters to build the GISO image.

Step 3 Copy the GISO image to the `/harddisk:` location on the router.

Step 4 Upgrade the system to replace the current software with the <*platform-architecture*>.iso image, and install the RPMs.

Example:

```
Router#install replace <source location> <giso name.iso>
```

If you are using a configuration file in GISO, use the following command to extract and replace the configuration.

```
Router# install replace <source location> <GISO-with-cfg>-<platform>.iso
```

Note
The default option is to replace the existing configuration. The install operation applies the configuration from a GISO, the router reboots to activate the configuration.

Step 5 View the version information for the GISO image. You can include a label to indicate the runing software version on the router. For example, create a label v1 for the current GISO version. When you rebuild GISO with additional RPMs, you can create a label v2 to distinguish the builds.

Example:

```
Router#show version
Cisco IOS XR Software, Version 7.0.1 LNT
Copyright (c) 2013-2019 by Cisco Systems, Inc.

Build Information:
 Built By     : xyz
 Built On     : Sat Jun 29 22:45:27 2019
 Build Host   : iox-lnx-064
```

```
 Workspace    : ../7.0.1
                /NCS540L/ws/
 Version      : 7.0.1
 Label        : 7.0.1-CUSTOMER_LABEL

cisco NCS540L
System uptime is 41 minutes
```

Upgrade the System Using the IOS XR Configuration File in GISO

Table 5: Feature History Table

Feature Name	Release Information	Description
Upgrade the system using the IOS XR configuration file in GISO	Release 7.5.2	This feature identifies and extracts the required IOS XR configuration file in GISO, and automates the updation of IOS XR configuration files when the router is reloaded with the new GISO. This feature introduces **iso-config [ignore | replace]** keywords to the **install replace** and **install package replace** commands.

You can upgrade the system when a newer version is available. A system upgrade replaces the existing version of the software with a newer version. By keeping the software up to date, you can ensure that the device works with the latest features and bug fixes.

You can replace the currently active software on your system with the software from a specified ISO image or GISO image. Only a minimal set of changes is performed to upgrade to the new software. Packages are not removed and reinstalled if they have the same name and version. For example, an upgrade that differs by only one package, removes and install only that one modified package.

Note Even if the configuration file is bundled in a GISO, the file is not used during an upgrade of the system to Cisco IOS XR Software Release 7.5.2. When first moving the system to release 7.5.2, the configuration file is only used when the system is completely reimaged in case of disaster recovery. Upgrades from release 7.5.2 to a GISO based off release 7.5.2 (or a higher version) uses the configuration file if it is present in the GISO.

Note The instructions in this section also apply to system downgrade.

This section shows replacing the current software version with `.iso` image.

In this scenario, you replace the current software with image, apply the changes, and commit the install operation. Committing the changes indicates the end of the current transaction. The updated software is used after the changes are applied, before the install transaction is committed.

A reboot is not always necessary. Bugfixes can be applied using a GISO that contains the current running software and the additional bugfix. These fixes may not require a reboot. Reboots are required for a system

version change, if too many processes need restarting, there is configuration in the GISO to apply or if the bugfix is marked as requiring a reload.

Note The ztp.ini is an initialization file for ZTP that is provided when building a golden ISO (GISO). This file defines the GISO ZTP configuration. If the `.ini` file is used, the next time ZTP starts, this custom `.ini` file that was used is picked up. For example, if the install starts a reload operation or when ZTP is initiated manually, the ZTP runs with the custom `ini`. This file is carried over during install replace and install rollback, as long as both the `from-version` and the `to-version` are Cisco IOS XR, Release 7.3.1 or higher.

If you start an **install replace** or **install package replace** operation with a GISO that has a ztp.ini file, the GISO ZTP configuration is updated on both RPs when the operation is applied. To undo this install operation and also the change in GISO ZTP configuration, use **install package abort** command. If you replace the image using an ISO without a ztp.ini file, the GISO ZTP configuration is removed from both RPs, if it exists, when the operation is applied.

Cisco IOS XR Software Release 7.5.2 introduces **iso-config [ignore | replace]** keywords to the **install replace** and **install package replace** commands. Before this release, the configuration file in GISO is ignored by default. If the GISO has a configuration file and it is not ignored using the **iso-config ignore** option, then the configuration in the file is applied, and the install operation requires a reboot. With the **iso-config** keyword, the default value is set to **replace** to replace the running configuration with the configuration in the file.

You can extract the configuration file from GISO using `image.py` script that is embedded in the ISO. This script interprets the GISO, and serves as an API to the ISO content at the filesystem level. The extracted configuration file replaces the running configuration after the router reloads.

Table 6: Configuration File in GISO

Scenario	iso-config replace	iso-config ignore	Nothing Specified
Configuration is present in GISO	Extracts the configuration and replaces the running configuration after the changes are applied via router reload. The output of the **show install log** command displays `ISO config replace requested (ISO config present).`	Does not extract or apply the configuration. The output of the **show install log** command does not display any information related to the configuration.	Extracts the configuration and replaces the running configuration after the changes are applied via router reload. The behavior is similar to **iso-config replace** operation.
Configuration is not present in GISO	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The output of the **show install log** command displays `ISO config replace requested (ISO config not present).`	Does not extract or apply the configuration. The output of the **show install log** command does not display any information related to the configuration.	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The behavior is similar to **iso-config replace** operation.

Scenario	iso-config replace	iso-config ignore	Nothing Specified
Configuration extraction is not supported in target GISO (downgrade)	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The output of the **show install log** command does not display any information related to the configuration.	Does not extract or apply the configuration. The output of the **show install log** command does not display any information related to the configuration.	Attempts to extract the configuration, and the operation continues as if configuration is not requested. The behavior is similar to **iso-config replace** operation.
Configuration is supported but the extraction fails	Attempts to extract the configuration. But the operation fails with error `Failed to extract the gISO config file from the ISO: <path to ISO>`.	Does not extract or apply the configuration. The output of the **show install log** command does not display any information related to the configuration.	The behavior is similar to **iso-config replace** operation.

Procedure

Step 1 Copy the ISO (or GISO) image to the `/harddisk:` location on the router.

Step 2 Upgrade the system to replace the current software with the .iso image.

Example:

```
Router#install package replace /harddisk:/ncs540.iso
```

Step 3 Activate the new .iso image on the router by applying the changes.

Example:

```
Router#install apply {reload | restart} [noprompt]
```

If you are using a configuration file in GISO, use the **install apply reload** command for the configuration to take effect.

Note
You can use a single command to perform both the packaging operation and activating the applying the changes using **install replace /harddisk:/540.iso noprompt** command.

To identify whether a reload is required or only process restart is needed, use either **show install history last transaction verbose** command or **show install request** command.

Include the keyword `noprompt` in the command to enable the system to bypass your permission to reload the router.

Applying the change gives you the flexibility to test the operation of the new software before committing the changes. If you reload the router, the router reverts the software to its previous software state.

All operations that automatically apply the new software are prohibited when an atomic change is already in progress. You must address the current atomic-change before performing this operation. To address the change, apply the current atomic-change, or cancel it with the **install package abort all-since-apply** command.

Step 4 View the install log.

Example:

```
Router#show install log detail
Fri Nov 12 09:43:49.702 UTC
2021-11-12 09:33:47 UTC    Transaction 1 started
2021-11-12 09:33:47 UTC    Atomic change 1.1 started
2021-11-12 09:33:47 UTC    Packaging operation 1.1.1 started
2021-11-12 09:33:47 UTC    Replace


2021-11-12 09:35:58 UTC    Packaging operation 1.1.1 success
2021-11-12 09:36:04 UTC    Apply by reload started
2021-11-12 09:38:48 UTC    Atomic change 1.1 successfully applied by reload
```

If configuration file is present in GISO, the output shows that the replace operation for ISO configuration is requested.

Step 5 Verify that the image is activated successfully.

Example:

```
Router#show install request
```

Step 6 Commit the transaction.

Example:

```
Router#install commit
```

Note
Any action requests may be run synchronously from the CLI. During this request, you are updated with the status of the request whenever it changes. The following example shows the output from a synchronous action request:

```
Router#install commit synchronous
Starting:
  install commit
Transaction 1
The install operation will continue in the background
Press Ctrl-C to return to the exec prompt. This will not cancel the install operation

Current activity: Initializing ....
Current activity: Commit transaction ......

Transaction 4: 'install commit' completed without error
```

Upgrade the System to Obtain Bug Fixes

Table 7: Feature History Table

Feature Name	Release Information	Description
Install Cisco RPM Directly from TAR File	Release 7.3.1	Cisco RPMs can be installed from one of these sources - repository name, repository URL or a local file path. Previously, to install RPMs from a TAR file, you had to manually extract the TAR file and specify the path to the locally extracted file to install the RPM. This feature extends support to install the RPM directly from the TAR file. Bug fix RPMs are available as TAR files on the Software Download page. The RPMs can be installed from the TAR file using CLI or Yang data model.

You can upgrade the system to obtain all available bug fixes or choose specific bug fix using bug ID.

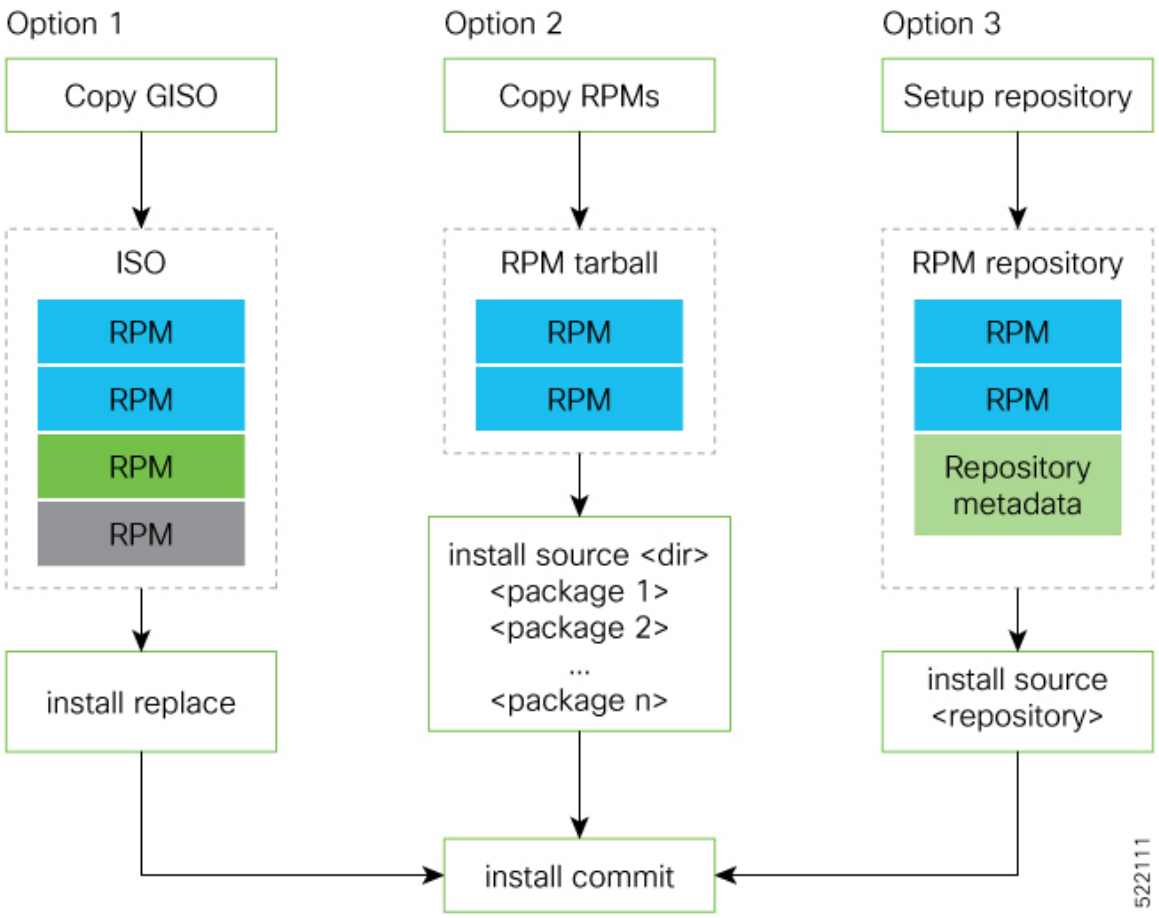
Bug fixes are available as optional RPMs. The bug fixes replace packages in the base ISO without adding up to the image size. For example, even if you add 20 RPMs to the GISO, the GISO file size will remain the same as the initial ISO.

You can download the bug fix RPMs from the Cisco Software Download portal .

From this page, download the latest bug fix RPMs as tarballs to the install repository. Untar the tarball into RPMs.

The `README` file provides the relevant information about the bug fix and also identifies the dependencies, if any, where other bug fix RPMs may be required for a complete fix.

The following image shows the options to install the bug fix RPMs.



The software is split into modular blocks and the package manager infrastructure computes the dependencies between the blocks. Each block of software has a top-level package and various partition-level packages. Bug fixes that span multiple blocks may lead to creating multiple dependent packages that are built as part of earlier bug fixes.

Note We recommend that you leverage the GISO workflow. Contact Cisco Support to build a new GISO with the required bug fixes. The RPMs are present in the initrd, which is the initial RAM disk for the boot loader, and this requires that the package is signed by Cisco. You can install GISO using a single `install replace` operation. For more information about GISO, see the *Build a Golden GISO* section.

However, if you do not prefer using the GISO, here are a few alternative ways to install bug fixes:

- Create a bug fix tarball to contain all the dependent packages. The tarball may be reasonably large for bug fixes that span mulitple blocks, but you can use this single tarball file directly to install the bug fixes without the need to check the individual packages within the file. You can also create a tarball containing all the dependent packages manually, but this method does not scale well.
- Create tarballs to install one or more bug fixes. For example, if you are installing bug fixes CSCxx11111, CSCyy22222 and CSCzz33333, you can use the individual tarball files and create a single tarball file.

- Use a Dandified Yum- or DNF repository to install, update, or remove relevant bug fixes. See, Create Repository to Access Files for Installing IOS XR Software, on page 8.

Note The RPMs can be installed from the TAR file using CLI or Yang data model.

- CLI—
  - Packaging operation: `install package add source /harddisk:/ncs540-<version>.tgz synchronous`
  - Packaging operation and atomic change: `install source /harddisk:/ncs540-<version>.tgz synchronous`
- Yang Data Model—

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:
base:1.0" message-id="101">
<install-package-add xmlns=
"http://cisco.com/ns/yang/Cisco-IOS-XR-install-augmented-act">
<source-type>tar</source-type>
<source>/misc/disk1/ncs540-<version>.tgz</source>
</install-package-add>
</rpc>
```

Note Use the RPM repository to harvest the benefits of package manager. The package manager queries the available packages, and downloads only those packages and their dependencies that are needed for installation.

Procedure

Step 1 View the list of available bug fixes.

Example:

```
Router#show install fixes available
Bug Id             Packages                      Repository
-----------    ----------------------------  ----------------
CSCxx12345     xr-540l-core-7.0.1v1.0.1-1      <repository-name>

               xr-core-7.0.1v1.0.1-1           <repository-name>
```

Step 2 Install the bug fix or package using one of the following options:

- Install the package where the bug fix is applied.

```
Router#install package upgrade xr-540l-core-7.0.1v1.0.1-1 xr-core-7.0.1v1.0.1-1
Packaging operation 1.1.1 started - xr-540l-core-7.0.1v1.0.1-1 xr-core-7.0.1v1.0.1-1
```

This task can also be accomplished using Cisco-IOS-XR-install-augmented-oper data model.

Apply the changes.

```
Router#install apply [reload | restart]
```

Note
To identify if you must reload or restart the system while applying the changes, use one of these two methods:

- History of last transaction

```
Router#show install history last transaction verbose
2019-09-11 17:01:46 UTC    Transaction 3 started
2019-09-11 17:01:46 UTC      Atomic change 3.1 started
2019-09-11 17:01:46 UTC      Packaging operation 3.1.1 started
2019-09-11 17:16:46 UTC    Transaction 3 complete

Least impactful apply method: process restart
```

  The command also displays the information about the changes to files and processes because of the install operation, and the package operations carried out on each node.

- Show install request

  Here, both `install apply restart` and `install apply reload` options are available. In this case, use `install apply restart` command because the impact on the system is the least. But when only an `install apply reload` option is available, then reload is the only option to apply the change.

- Install the optional package. Changes are applied automatically.

  Attention
  Automatic change may trigger a reload of the router depending on the package that is installed.

```
Router#install source /harddisk:/files xr-540l-core-7.0.1v1.0.1-1.x86_64.rpm
```

  Note
  Packages can also be installed using the package name.

```
Router#install source /harddisk:/files  xr-540l-core
```

- This task can also be performed using YANG data models. Use `install` RPC on the Cisco-IOS-XR-install-act data model. Here is an example usage with an HTTP repository:

```
<install>
   <packages>
     <packagename>pkg1</packagename>
   </packages>
   <source-type>http</source-type>
   <source><path-to-source></source>
</install>
```

Step 3 View the state of the packaging operation.

Example:

```
Router#show install request
User request: install package upgrade xr-<platform>-core-<version> xr-core-<version>
Operation ID: 2.1.2
State:        In progress since

Current activity:    Initiate operation
Next activity:       Begin transaction
Time started:        2019-06-25 07:41:06

No per-location information.
```

Step 4 View the log to ensure that the installation is successful.

Example:

```
Router#show install log
2019-06-25 07:41:06 UTC    Transaction 1 started
2019-06-25 07:45:08 UTC    Upgrade (Success)
2019-06-25 07:45:08 UTC    xr-<platform>-core-<version>
2019-06-25 07:45:08 UTC    xr-core-<version>
2019-06-25 07:57:02 UTC    Atomic change 1.1 successfully applied by reload
```

Step 5 View the history of the install operation.

Example:

```
Router#show install history table
  Transaction             Atomic Change          Packaging Operations
--------------------  --------------------------  ----------------------------------
Id   Status          Id  Method   Status      Id  Operation   Inputs Status
--------------------  --------------------------  ----------------------------------
 1   In progress     1   Reload   Success      1   Upgrade        1  Success
```

The command can also be used to view more details if there is a failed operation.

Use **show install history id <operation-id>** command to filter the history of install information by ID. IDs are of the form `<transaction id>.<atomic id>.<packaging id>`.

```
Router#show install history id ?
  WORD Specify an operation ID (e.g. 1, 1.2, 1.2.3)
```

Use **show install history last** command to view the last packaging operation, atomic change, or transaction.

```
Router#show install history last ?
     atomic-change  Show the last atomic change
     package        Show the last packaging operation
     transaction    Show the last transaction
```

Step 6 After the operation is complete, verify that the packages `xr-540l-core-7.0.1v1.0.1-1` and `xr-core-7.0.1v1.0.1-1` are installed and active.

Example:

```
Router#show install active summary
xr-ncs540l-bfd                                          7.0.1v1.0.0-1
xr-ncs540l-bmc                                          7.0.1v1.0.0-1
xr-ncs540l-bundles                                      7.0.1v1.0.0-1
xr-ncs540l-card-support                                 7.0.1v1.0.0-1
xr-ncs540l-core                                         7.0.1v1.0.1-1
xr-ncs540l-x64-core                                     7.0.1v1.0.1-1
xr-core                                                 7.0.1v1.0.1-1
xr-core-calv                                            7.0.1v1.0.0-1
xr-host-core                                            7.0.1v1.0.0-1
xr-ip-core                                              7.0.1v1.0.0-1
xr-spi-core                                             7.0.1v1.0.0-1
```

Example:

Example:

```
Router#show install active summary
```

The version has changed. The version `1.0.1-1` indicates that the bug fix is installed.

This task can also be accomplished using data models. Use the `get` RPC for `install.fixes.active` operation using Cisco-IOS-XR-install-augmented-oper data model.

Step 7 Commit the changes for the changes to persist after a reload operation.

Example:

```
Router#install commit
```

Step 8 View the list of bug IDs for which fixes are committed.

Example:

```
Router#show install fixes committed
```

This task can also be accomplished using data models. Use the `get` RPC for `install.fixes.committed` operation using Cisco-IOS-XR-install-augmented-oper data model.

Step 9 View the list of active bug fix RPMs.

Example:

```
Router#show install fixes active
```

This task can also be accomplished using data models. Use the `get` RPC for `install.fixes.active` operation using Cisco-IOS-XR-install-augmented-oper data model.

Roll Back Software to a Previously Saved Installation Point

Note This section does not apply to the following variants of the Cisco NCS 540 routers:

- N540X-6Z18G-SYS-A
- N540X-6Z18G-SYS-D
- N540X-8Z16G-SYS-A
- N540X-8Z16G-SYS-D
- N540X-4Z14G2Q-SYS-A
- N540X-4Z14G2Q-SYS-D
- N540-24Q8L2DD-SYS

Table 8: Feature History Table

Feature Name	Release Information	Feature Description
Roll back to a previously-saved installation point	Release 7.4.1	A maximum of 900MB of disk space is used to store all rollback points. You can delete older rollback points to make space for newer or smaller rollback points such as, SMUs. The following NCS 540 variants allow up to 900MB of disk space for rollback points: • N540-28Z4C-SYS-A/D • N540X-16Z4G8Q2C-A/D • N540-12Z20G-SYS-A/D • N540X-12Z16G-SYS-A/D This feature introduces the following commands: • **install commit description**—sets the description for a transaction ID • **install label id** *id* **description** *description*—adds a new description or overwrites the description of the specified transaction ID. The following command is updated to include the keyword, **oldest** *n*: • **clear install rollback oldest** *n*—deletes the specified number of rollback points

You can roll your system software back to a previous version. This could be used to discard an ongoing install operation, or to undo an install operation that has already been committed. After each commit operation, the system saves a record of the committed software packages. Each record is a restoration point, and is assigned a unique ID. This ID is known as a transaction ID. You can use the transaction ID to roll back the software to a restoration point associated with this ID. Up to 900MB of space is allowed for rollback points, instead of a specific number of rollback points.

Note

- Use transaction ID 0 to roll back to the software that was present after the system booted for the first time.
- If you commit an install transaction using **install commit** command, the GISO ZTP configuration is saved along with the rest of the software changes. This means that if you use the **install rollback** or **install package rollback** command to revert the software to the state of a previous transaction, the GISO ZTP configuration is also reverted to its previous state. To undo this install operation, as well as the change in GISO ZTP configuration, use **install package abort** command. If there is no GISO ZTP configuration saved at the end of the install transaction, the existing GISO ZTP configuration is reverted to the previous state when the rollback operation of that transaction is applied.

Procedure

Step 1 View the list of available transaction IDs.

Example:

```
Router# show install rollback list-ids
```

Step 2 Explore the main packages that can be installed if you roll the software back to the specific transaction ID.

Example:

```
Router# show install rollback id <id>
```

Step 3 View the relative changes that are made to the currently installed software if it is rolled back to a transaction ID.

Example:

```
Router# show install rollback id <id> changes
```

To perform these tasks using data models, use the `get` RPC on the Cisco-IOS-XR-install-augmented-oper data model.

```
<rpc>
  <get>
   <filter type="subtree">
      <install xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-install-augmented-oper">
      <rollback/>
     </install>
   </filter>
  </get>
</rpc>
```

Step 4 Roll back to the software associated with the specific transaction ID.

Example:

```
Router# install rollback <id> [commit]
```

If you want to apply the change and roll back to the associated transaction ID, commit the change. You can also include the keyword `noprompt` in the command to enable the system to bypass your permission to reload the router.

Attention
This roll back operation installs the previous software and also applies the change automatically. This may reload the router depending on the package that is rolled back.

Alternatively, use the **install package rollback** command to only roll back the package but not apply the changes. You can check whether the router will reload or restart if you apply the change using the **show install history last transaction verbose** command or **show install request** command. Based on the command output, you can take the appropriate action using **install apply reload | restart** command to either reload or restart the system. Use the **install commit** command to commit the transaction.

To perform this task using data models, use the `install-rollback` RPC on the Cisco-IOS-XR-install-augmented-oper data model.

```
<rpc>
 <install-rollback xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-install-augmented-act">
  <commit>true</commit>
   <transaction-id>0</transaction-id>
 </install-rollback>
</rpc>
```

To understand the data model structure and its arguments, see the Access the Install-Related Data Model.

Step 5 Commit the operation.

Example:

```
Router#install commit
```

Telemetry Support for Install Operations

Table 9: Feature History Table

Feature Name	Release Information	Description
Stream Telemetry Data about Install Operations	Release 7.5.2	You can stream telemetry data for install-related details such as active and committed packages, view the progress of install operations, retrieve the image version, and view the error messages with recovery information when an operation fails.

To stream telemetry data that is related to software installation, you must create subscriptions to the sensor paths in the YANG data models. See *Obtain Data Models for Install Operation* for the list of supported data models. For information about establishing a telemetry session and creating subscriptions, see the *Telemetry Configuration Guide for Cisco NCS 540 Series Routers*.

Stream Telemetry Data About	Description	YANG Path
Summary of active packages	Data is streamed after a successful **apply** operation. An active package is the software currently running on the system.	`Cisco-IOS-XR-install-oper: install/packages/active/summary`
Summary of committed packages	Data is streamed after a successful **commit** operation. A package that is committed remains active following a system reload.	`Cisco-IOS-XR-install-oper: install/packages/committed/summary`
Status of the last request operation	Data is streamed when starting a new request and also when entering an `idle` state. If the operation has failed, this includes error messages along with recovery state.	`Cisco-IOS-XR-install-oper: install/request`
Image version and GISO label	Data is streamed after a successful **apply** operation.	`Cisco-IOS-XR-install-oper: install/version`
Packaging information	Data is streamed at the start and end of a packaging operation.	`Cisco-IOS-XR-install-augmented-oper: install/history/latest-packaging-operation`
Atomic information	Data is streamed at the start and end of **apply** operation.	`Cisco-IOS-XR-install-augmented-oper: install/history/latest-atomic-change`
Transaction information	Data is streamed at the start, in progress, and end of a **commit** operation. **Note** After a transactional rollback, some of the data such as summary of active packages, image version can change. However, telemetry events are not sent after the reload operation.	`Cisco-IOS-XR-install-augmented-oper: install/history/latest-transaction`

Caveats

Release 7.10.1

The following upgrade caveats are applicable for Cisco NCS 540 Series Routers listed in the *Cisco NCS 540 Series Router and PIDs* table for Release 7.10.1 and later.

Table 10: Upgrade Caveats for Cisco NCS 540 Series Routers

From	To	Bridge SMUs Required (CSCwd71524)	Caveats
7.3.2	7.10.1 and later	Yes	1*,2*,3*,5*
7.5.2	7.10.1 and later	Yes	1*,2*,4*
7.6.1	7.10.1 and later	Yes	1*,2*,4*
7.6.2	7.10.1 and later	None	1*
7.7.1	7.10.1 and later	None	1*
7.7.2	7.10.1 and later	None	1*
7.8.1	7.10.1 and later	None	1*
7.8.2	7.10.1 and later	None	1*
7.9.1	7.10.1 and later	None	1*
7.9.2	7.10.1 and later	None	1*

1*: You can't roll back using the **install rollback** command.

2*: Ensure that a reload bridging SMU (CSCwd71524) is installed.

3*: Ensure that you install the bridge SMU (CSCwd71524) manually because even if it's available inside the GISO that's replacing the existing GISO, this SMU doesn't get installed automatically.

4*: It’s applicable to the Small Density Routers listed in the *Cisco NCS 540 Series Router and PIDs* table. Before replacing GISO, ensure that you install the bridge SMU (CSCwd71524) manually.

5*: Upgrade is not supported by the Small Density Routers listed in the *Cisco NCS 540 Series Router and PIDs* table.

Note CSCwd71524:

- When you upgrade from earlier than Release 7.10.1 to Release 7.10.1, system supports the installation process seemlessly.
- When you downgrade from Release 7.10.1, system preserves the present configuration and the install history from last transaction.

The following downgrade caveats are applicable for Cisco NCS 540 Series Routers listed in the *Cisco NCS 540 Series Router and PIDs* table for Release 7.10.1 and later.

Table 11: Downgrade Caveats for Cisco NCS 540 Series Routers

From	To	Target SMUs Required (CSCwd71524)	Caveats
7.10.1 and later	7.3.2	Yes	C*
7.10.1 and later	7.5.2	Yes	***, A*, B*, D*
7.10.1 and later	7.6.1	Yes	***, A*, B*, D*
7.10.1 and later	7.6.2	Yes	***, A*, B* , D*
7.10.1 and later	7.7.1	Yes	***, A*, B*, D*
7.10.1 and later	7.7.2	Yes	***, A*, B*, D*
7.10.1 and later	7.8.1	Yes	***, A*, B*, D*
7.10.1 and later	7.8.2	Yes	***, A*, B*, D*
7.10.1 and later	7.9.1	Yes	***, D*
7.10.1 and later	7.9.2	Yes	***, D*

***

- You don't need to run the **install commit** command after a downgrade operation because the operation is automatically committed.
- You can't roll back after a downgrade. To revert to the previous IOS XR previous version, replace or reimage to the relevant ISO.
- IOS XR configuration history is lost after a downgrade, but the router preserves the latest configuration.
- Install history from the last transaction is preserved after a downgrade operation.
- Downtime takes a longer time as the operation is performed through reimage.
- You can't downgrade using the **install package replace** command. Instead, use the **install replace** command to downgrade.
- Ensure that you reinstall third-party application once you complete the downgrade.
- Ensure that you disable reimage-protection feature before performing downgrade because the downgrade occurs through a reimage. You can disable the reimage-protection using the **platform security reimage-protection disable** command.

Note The reimage-protection feature applies to the Cisco NCS 540 Series Routers listed in the *Cisco NCS 540 Series Router and PIDs* table.

- PXE recovery is required if the image downgrading isn’t bootable.
- You must re-install the *Type 6 masterkey* and reapply the configuration encrypted by it because they are lost after the downgrade.

- You must regenerate crypto keys and certificates after a downgrade.

A*: You can't downgrade to the base ISO. You can downgrade to a GISO containing the bridge SMU (CSCwd71524).

B*: You must recover the router through PXE if a power cycle occurs during the downgrade.

C*: One-step downgrade isn't supported. You must use either PXE/USB to downgrade or perform a two-step downgrade through Release 7.9.1 or Release 7.5.4. The first-hop downgrade to Release 7.9.1 or Release 7.5.4 still carries the same caveats.

D*: FPD upgrade may cause your router to reload an extra time during boot-up.

E*: Traffic is impacted.

Use the **show install upgrade-matrix running** command to view the caveats.

Hardware MDB Profiles for Layer 2 and 3 Services

Table 12: Feature History Table

Feature Name	Release	Description
Hardware MDB profiles for Layer 2 and Layer 3 services overlay scale increased to 32000	Release 25.2.1	You can now configure hardware MDB profiles for Layer 2 and Layer 3 services together, over SRv6 underlay, enabling you to set overlay scales to 32000 in terms of the number of routers and sessions. This release adds the following hardware MDB profiles to the **hw-module profile mdb** command: • **l2max-srv6** • **l2max-se-srv6** • **l3max-srv6** • **l3max-se-srv6**
Hardware MDB Profiles for Layer 2 Services	Release 7.6.1	You can now configure hardware modular database (MDB) profiles for Layer 2 services, enabling you to achieve higher scales in terms of the number of routers and sessions. In earlier releases, you could configure hardware MDB profiles for Layer 3 services. This release adds the following hardware MDB profiles to the **hw-module profile mdb** command: • **l2max** • **l2max-se**

Hardware MDB profile is used to modify router resources for the specific needs during the router boot up time. You can configure the hardware MDB profile or you can view the default profile.

Hardware MDB profile feature was introduced in Cisco IOS XR software Release 7.4.1 with **l3max** and **l3max-se** configuration profile options.

Starting from Cisco IOS XR software Release 7.6.1, it supports **l3max**, **l3max-se**, **l2max**, and **l2max-se** profiles, and provides high scale layer 2 and layer 3 services. Router with base line cards supports **l2max** and **l3max** profiles, and routers with scale line cards support **l2max-se** and **l3max-se** profiles.

All features that are supported on **l3max** or **l3max-se** hardware profile are also supported on **l2max** or **l2max-se** hardware profile and the features that are supported on **l2max** or **l2max-se** hardware profile are also supported on **l3max** or **l3max-se** hardware profile respectively.

Use the **hw-module profile mdb** command in the Global Configuration mode to configure the hardware MDB profile options.

Table 13: MDB Hardware Module Profile for Fixed System of Cisco NCS 540 Series Routers

Base Profile	Scale Profile
• Default profile is **l3max**. • Use **hw-module profile mdb l2max** command to configure **l2max** profile.	• Default profile is **l3max-se**. • Use **hw-module profile mdb l2max-se** command to configure **l2max-se** profile.

Configure Hardware MDB Profiles for Layer 2 Services

Router supports configuration of hardware MDB profiles for layer 2 services

Prerequisites:

- Use **hw-module profile npu native-mode-enable** command to configure.

Configuration Example

This example shows how to configure the hardware MDB profile as **l3max** in the Cisco NCS 540 Series base line cards.

```
Router# configure
Router(config)# hw-module profile mdb l3max
Router(config)# commit
```

Verification

```
Router# show hw-module profile mdb-scale
MDB scale profile: l3max
```

Configuration Example

This example shows how to configure the hardware MDB profile as **l2max** in the Cisco NCS 540 Series base line cards.

```
Router# configure
Router(config)# hw-module profile mdb l2max
Router(config)# commit
```

Verification

```
Router# show hw-module profile mdb-scale
MDB scale profile: l2max
```

Configuration Example

This example shows how to configure the hardware MDB profile as **l3max-se** profile in the Cisco NCS 540 Series scale line cards.

```
Router# configure
Router(config)# hw-module profile mdb l3max-se
Router(config)# commit
```

Verification

```
Router# show hw-module profile mdb-scale
MDB scale profile: l3max-se
```

Configuration Example

This example shows how to configure the hardware MDB profile as **l2max-se** profile in the Cisco NCS 540 Series scale line cards.

```
Router# configure
Router(config)# hw-module profile mdb l2max-se
Router(config)# commit
```

Verification

```
Router# show hw-module profile mdb-scale
MDB scale profile: l2max-se
```