Customize Installation using Golden ISO

Golden ISO (GISO) is a customized ISO that a user can build to suit the installation requirement. The user can customize the installable image to include the standard base image with the basic functional components, and add additional RPMs, SMUs and configuration files based on requirement.

The ease of installation and the time taken to seamlessly install or upgrade a system plays a vital role in a cloud-scale network. An installation process that is time-consuming and complex affects the resiliency and scale of the network. The GISO simplifies the installation process, automates the installation workflow, and manages the dependencies in RPMs and SMUs automatically.

GISO is built using a build script `gisobuild.py` available on the box in `/pkg/bin` location in XR domain. For more information about the build script and the steps to build GISO, see Build Golden ISO, on page 4.

When a system boots with GISO, additional SMUs and RPMs in GISO are installed automatically, and the router is pre-configured with the XR configuration in GISO. For more information about downloading and installing GISO, see Install Golden ISO, on page 7.

The capabilities of GISO can be used in the following scenarios:

- Migration from IOS XR 32-bit to IOS XR 64-bit
- Initial deployment of the router
- Software disaster recovery
- System upgrade from one base version to another
- System upgrade from same base version but with additional SMUs
- Install update to identify and update dependant packages

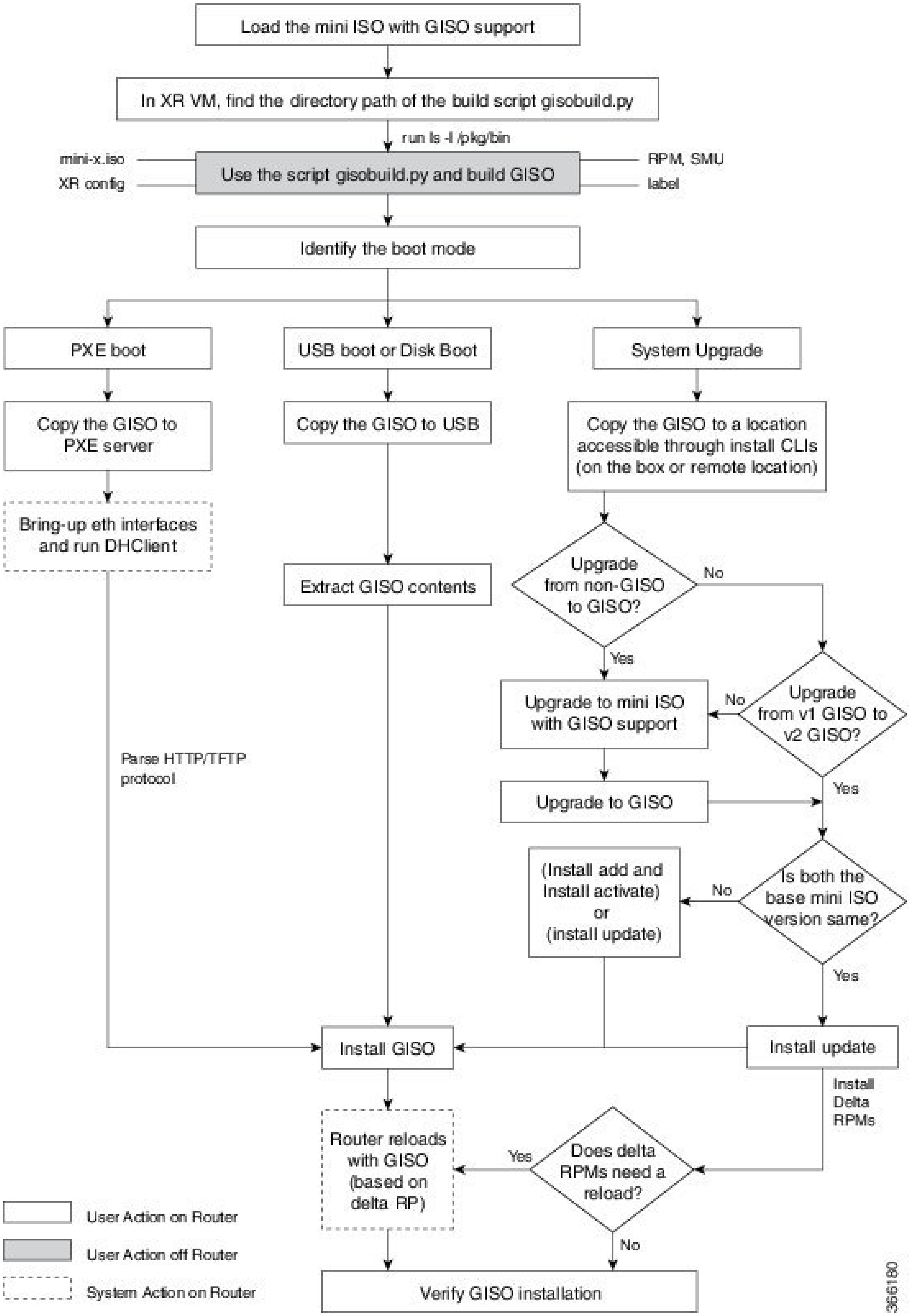
Limitations

The following are the known problems and limitations with the customized ISO:

- Building and booting GISO for asynchronous package (a package of different release than the ISO) is not supported.
- GISO build script `gisobuild.py` does not support verifying the XR configuration.
- Renaming a GISO build and then installing from the renamed GISO build is not supported.

Golden ISO Workflow

The following image shows the workflow for building and installing golden ISO.


Load the mini ISO with GISO support
In XR VM, find the directory path of the build script gisobuild.py
run ls -l /pkg/bin
mini-x.iso
XR config
Use the script gisobuild.py and build GISO
RPM, SMU
label
Identify the boot mode
PXE boot
USB boot or Disk Boot
System Upgrade
Copy the GISO to PXE server
Copy the GISO to USB
Copy the GISO to a location accessible through install CLIs (on the box or remote location)
Bring-up eth interfaces and run DHClient
Extract GISO contents
Upgrade from non-GISO to GISO?
No
Yes
Upgrade to mini ISO with GISO support
No
Upgrade from v1 GISO to v2 GISO?
Parse HTTP/TFTP protocol
Upgrade to GISO
Yes
(Install add and Install activate) or (install update)
No
Is both the base mini ISO version same?
Yes
Install GISO
Install update
Install Delta RPMs
Router reloads with GISO (based on delta RP)
Yes
Does delta RPMs need a reload?
No
Verify GISO installation
User Action on Router
User Action off Router
System Action on Router
366180

Build Golden ISO

The customized ISO is built using Cisco Golden ISO (GISO) build script `gisobuild.py` available on the box in `/pkg/bin` location in XR domain.

The GISO build script 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 mini-x.iso version.
- Skips and removes third-party RPMs that are not SMUs of already existing third-party base package in mini-x.iso.
- Displays an error and exits build process if there are multiple base RPMs of same release but different versions.
- Performs compatibility check and dependency check for all the RPMs. For example, the child RPM ncs5500-mpls-te-rsvp is dependent on the parent RPM ncs5500-mpls . If only the child RPM is included, the Golden ISO build fails.

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

- Base mini-x.iso (mandatory)
- XR configuration file (optional)
- one or more Cisco-specific SMUs for host, XR and System admin
- one or more third-party SMUs for host, XR and System admin
- Label for golden ISO

Note Golden ISO can be built only from mini ISO. The full or fullk9 bundle ISO is not supported.

Use the following naming convention when building GISO:

GISO Build	Format	Example

GISO without k9sec RPM	`<platform-name>-golden-x.iso-<version>.<label>`
	`<platform-name>-golden-x-<version>.iso.<label>`
		<platform-name>-golden-x64.iso-<version>.v1
		<platform-name>-golden-x64-<version>.iso.v1

GISO with k9sec RPM	`<platform-name>-goldenk9-x.iso-<version>.<label>`
	`<platform-name>-goldenk9-x-<version>.iso.<label>`
		<platform-name>-goldenk9-x64.iso-<version>.v1
		<platform-name>-goldenk9-x64-<version>.iso.v1

Note To successfully add k9sec RPM to GISO, change the permission of the file to `644` using the **chmod** command.

```
chmod 644 [k9 sec rpm]
```

To build GISO, perform the following steps:

Before You Begin

- To upgrade from non-GISO to GISO version, it is mandatory to first upgrade to mini ISO with GISO support. For NCS 5500 series routers, upgrade to release 6.2.2 or later.
- The system where GISO is built must meet the following requirements:
  - System must have Python version 2.7 and later.
  - System must have free disk space of minimum 3 to 4 GB.
  - Verify that the Linux utilities `mount`, `rm`, `cp`, `umount`, `zcat`, `chroot`, `mkisofs` are present in the system. These utilities will be used by the script. Ensure privileges are available to execute all of these Linux commands.
  - Kernel version of the system must be later than 3.16 or later than the version of kernel of Cisco ISO.
  - Verify that a `libyaml rpm` supported by the Linux kernel is available to successfully `import yaml` in the tool.
  - User should have proper permission for security rpm(k9sec-rpm) in rpm repository, else security rpm would be ignored for Golden ISO creation.
- The system from where the gisobuild script is executed must have root credentials.

Step 1 From the XR prompt, find the directory path of the build script `gisobuild.py`.

```
Router# run ls -l /pkg/bin | grep giso
```

Step 2 Copy the script `gisobuild.py` from the router to an offline system or external server where the GISO will be built. Ensure that this system meets the pre-requisites described above in the *Before You Begin* section.

a) Go to shell prompt of IOS XR and copy the script `gisobuild.py` to harddisk:

```
Router#run
Mon Jul 10 13:14:43.606 UTC
[Router]$cp /pkg/bin/gisobuild.py /harddisk\:/
[Router]$exit
logout
```

b) Go to EXEC mode of IOS XR and check the script `gisobuild.py` in harddisk:

```
Router#dir harddisk:/ | inc gisobuild.py
Mon Jul 10 13:15:20.542 UTC

     13 -rwxr-xr-x 1 77203 Jul 10 13:14 gisobuild.py
```

c) Copy the script `gisobuild.py` from harddisk:/ to any desired location (outside the router where ISO will be built) that is supported. Use any of the supported protocol.

```
Router#copy harddisk:/ ?
  WORD              Copy to file
  apphost:          Copy to apphost: file system
  config:           Copy to config: file system
  disk0:          Copy to disk0: file system
  ftp:            Copy to ftp: file system
  harddisk:       Copy to harddisk: file system
  http:           Copy to http: file system
  https:          Copy to https: file system
  location        location to copy from
  rootfs:         Copy to rootfs: file system
  running-config  Update (merge with) current system configuration
  tftp:           Copy to tftp: file system
```

The following example shows the script copied from one harddisk to another with a new name `gisobuild_new.py`:

```
Router#copy harddisk:/gisobuild.py harddisk:/gisobuild_new.py
Mon Jul 10 13:19:28.912 UTC
Destination filename [/harddisk:/gisobuild_new.py]?
CCCCC
77203 bytes copied in      0 sec
Router#dir harddisk:/ | inc gisobuild_new.py
Mon Jul 10 13:19:48.557 UTC
     14 -rwxr-xr-x 1 77203 Jul 10 13:19 gisobuild_new.py
```

Step 3 Run the script `gisobuild.py` as a root user and provide parameters to build the golden ISO off the router. Ensure that all RPMs and SMUs are present in the same directory. The number of RPMs and SMUs that can be used to build the Golden ISO is 128.

Note The `-i` option is mandatory, and either or both `-r` or `-c` options must be provided.

Note NCS5500 routers has two types of cards - x86_64 and arm. System Admin runs on both types of cards, whereas XR runs only on x86_64 card.

```
[directory-path]$ gisobuild.py [-h] [-i <mini-x.iso>] [-r <rpm repository>]
[-c <config-file>] [-l <giso label>] [-m] [-v]

System requirements check [PASS]

Platform: ncs5500 Version: 6.2.2

Scanning repository [repository-path]...

Building RPM Database...
Total 3 RPM(s) present in the repository path provided in CLI

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

        (+) ncs5500-mgbl-3.0.0.0-r622.x86_64.rpm

        ...RPM compatibility check [PASS]

Following SYSADMIN x86_64 rpm(s) will be used for building Golden ISO:

        (+) ncs5500-sysadmin-system-6.2.2-r622.CSCcv44444.x86_64.rpm

Following SYSADMIN arm rpm(s) will be used for building Golden ISO:

        (+) ncs5500-sysadmin-system-6.2.2-r622.CSCcv44444.arm.rpm
        ...RPM compatibility check [PASS]

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

XR rpms:
        ncs5500-mgbl-3.0.0.0-r622.x86_64.rpm

SYSADMIN rpms:
        ncs5500-sysadmin-system-6.2.2-r622.CSCcv44444.x86_64.rpm
        ncs5500-sysadmin-system-6.2.2-r622.CSCcv44444.arm.rpm

        ...Golden ISO creation SUCCESS.

Golden ISO Image Location: <directory-path>/ncs5500-golden-x.iso-6.2.2

Detail logs: <directory-path>/Giso_build.log-2017-07-10:18:17:43.395586
```

where:

- -i is the path to mini-x.iso
- -r is the path to RPM repository
- -c is the path to XR config file
- -l is the golden ISO label
- -h shows the help message
- -v is the version of the build tool `gisobuild.py`
- -m is to build the migration tar to migrate from IOS XR to IOS XR 64 bit

GISO is built with the RPMs placed in respective folders in the specified directory and also includes the log files `giso_summary.txt` and `gisobuild.log-<timestamp>`. The XR configuration file is placed as `router.cfg` in the directory.

Note The GISO script does not support verification of XR configuration.

What to Do Next

Install the golden ISO on the router.

Install Golden ISO

Golden ISO (GISO) automatically performs the following actions:

- Installs host and system admin RPMs.
- Partitions repository and TFTP boot on RP.

- Creates software profile in system admin and XR modes.
- Installs XR RPMs. Use **show instal active** command to see the list of RPMs.
- Applies XR configuration. Use **show running-config** command in XR mode to verify.

Step 1 Download GISO image to the router using one of the following options:

- **PXE boot:** when the router is booted, the boot mode is identified. After detecting PXE as boot mode, all available ethernet interfaces are brought up, and DHClient is run on each interface. DHClient script parses HTTP or TFTP protocol, and GISO is downloaded to the box.
- **USB boot or Disk Boot:** when the USB mode is detected during boot, and GISO is identified, the additional RPMs and XR configuration files are extracted and installed.
- **System Upgrade** when the system is upgraded, GISO can be installed using **install add**, **install activate**, or using **install update** commands.
    - **system upgrade from a non-GISO (image that does not support GISO) to GISO image:** If a system is running a version1 with an image that does not support GISO, the system cannot be upgraded directly to version2 of an image that supports GISO. Instead, the version1 must be upgraded to version2 mini ISO, and then to version2 GISO.
    - **system upgrade in a release from version1 GISO to version2 GISO:** If both the GISO images have the same base version but different labels, **install add** and **install activate** commands does not support same version of two images. Instead, using **install update** command installs only the delta RPMs. System reload is based on restart type of the delta RPMs.

```
Router#install update source <path-to-image> <platform-name-goldenk9-x64.iso-6.2.2>

Sat Dec  3 15:51:43.384 UTC

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

ISO <platform-name-goldenk9-x64.iso-6.2.2> in input package list. Going to upgrade the
system to version 6.2.2.

Updating contents of golden ISO

Scheme : localdisk

Hostname : localhost

Username : None

SourceDir : Dir

Collecting software state..

Fetching .... <ncs5500-name-goldenk9-x64.iso-6.2.2.v2>

LC/0/0/CPU0:Dec  3 15:53:41.580 : %PLATFORM-STATS_INFRA-3-ERR_STR_1 : ErrStr:Unable to map
 stats infra shared mem

Skipping openssh-scp-6.6p1.p1-r0.0.CSCtp12345.host.x86_64.rpm from GISO as it's active

Skipping <platform-name>-sysadmin-system-6.2.2-r622.CSCcv11111.x86_64.rpm from GISO as it's
 active

Skipping openssh-scp-6.6p1.p1-r0.0.CSCtp12345.admin.x86_64.rpm from GISO as it's active

Skipping openssh-scp-6.6p1-r0.0.host.x86_64.rpm from GISO (skipped SMUs base pkg)
```

```
Adding packages

        <platform-name>-sysadmin-shared-6.2.2-r622.CSCcv33333.x86_64.rpm

        <platform-name>-mpls-te-rsvp-x64-1.2.0.0-r622.x86_64.rpm

Router:Dec  3 15:54:12.308 : sdr_instmgr[1145]: %INSTALL-INSTMGR-2-OPERATION_SUCCESS :
Install operation 1 finished successfully

Dec 03 15:54:03 Install operation 1 started by root:

 install add source /misc/disk1/install_tmp_staging_area/6.2.2
ncs5500-sysadmin-shared-6.2.2-r622.CSCcv33333.x86_64.rpm
ncs5500-mpls-te-rsvp-x64-1.2.0.0-r622.x86_64.rpm

Dec 03 15:54:05 Install operation will continue in the background

Dec 03 15:54:13 Install operation 1 finished successfully

------Truncated
```

- **system upgrade across releases from version1 GISO to version2 GISO:** Both the GISO images have different base versions. Use **install add** and **install activate** commands, or **install update** command to perform the system upgrade. The router reloads after the upgrade with the version2 GISO image.

```
 install update source <path-to-image> <platform-name-goldenk9-x64.iso-6.2.2.opt

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
ISO <platform-name-goldenk9-x64.iso-6.2.2.opt> in input package list. Going to upgrade the
 system to version 6.2.2.
Scheme : localdisk
Hostname : localhost
Username : None
SourceDir : Dir
Collecting software state..
Getting platform
Getting supported architecture
Getting active packages from XR
Getting inactive packages from XR
Getting list of RPMs in local repo
Getting list of provides of all active packages
Getting provides of each rpm in repo
Getting requires of each rpm in repo
LC/0/0/CPU0: fab_si[193]: %PLATFORM-STATS_INFRA-3-ERR_STR_1 : ErrStr:Unable to map stats
infra shared mem
Fetching .... <platform-name-goldenk9-x64.iso-6.2.2.opt>
Adding packages
        <platform-name>-goldenk9-x64.iso-6.2.2.opt
.......
July 4 08:17:07 Install operation 2 started by root:
install add source <path>/6.2.2 <platform-name>-goldenk9-x64.iso-6.2.2.opt
July 4 08:17:08 Install operation will continue in the background
July 4 08:26:43 Install operation 2 finished successfully

Install add operation successful
Activating <platform-name>-goldenk9-x64-6.2.2
July 4 08:26:49 Install operation 3 started by root:
  install activate pkg <platform-name>-goldenk9-x64-6.2.2
July 4 08:26:49 Package list:
July 4 08:26:49     <platform-name>-goldenk9-x64-6.2.2
This install operation will reload the sdr, continue?
```

```
[yes/no]:[yes] Install operation will continue in the background
------Truncated
```

Step 2 Run the **show install repository all** command in System Admin mode to view the RPMs and base ISO for host, system admin and XR.

```
sysadmin-vm:0_RP0# show install repository all
Admin repository
---------------------
ncs5500-sysadmin-6.2.2
ncs5500-sysadmin-hostos-6.2.2-r622.CSCcv10001.admin.x86_64
ncs5500-sysadmin-hostos-6.2.2-r622.CSCcv10001.admin.arm
ncs5500-sysadmin-system-6.2.2-r622.CSCcv10005.x86_64
ncs5500-sysadmin-system-6.2.2-r622.CSCcv10005.arm
....
XR repository
------------------
ncs5500-iosxr-mgbl-3.0.0.0-r622.x86_64
ncs5500-xr-6.2.2
....
Host repository
---------------------
host-6.2.2
```

Step 3 Run the **show install package <golden-iso>** command to display the list of RPMs, and packages built in GISO.

```
Router#show install package ncs5500-goldenk9-x64-6.2.2

This may take a while ...
    ISO Name: ncs5500-goldenk9-x64-6.2.2
    ISO Type: bundle
    ISO Bundled: ncs5500-mini-x64-6.2.2
    Golden ISO Label: temp
    ISO Contents:
        ISO Name: ncs5500-xr-6.2.2
        ISO Type: xr
        rpms in xr ISO:
            iosxr-os-ncs5500-64-5.0.0.0-r622
            iosxr-ce-ncs5500-64-3.0.0.0-r622
            iosxr-infra-ncs5500-64-4.0.0.0-r622
            iosxr-fwding-ncs5500-64-4.0.0.0-r622
            iosxr-routing-ncs5500-64-3.1.0.0-r6122

        ISO Name: ncs5500-sysadmin-6.2.2
        ISO Type: sysadmin
        rpms in sysadmin ISO:
            ncs5500-sysadmin-topo-6.2.2-r622
            ncs5500-sysadmin-shared-6.2.2-r622
            ncs5500-sysadmin-system-6.2.2-r622
            ncs5500-sysadmin-hostos-6.2.2-r622.admin
        ...

        ISO Name: host-6.2.2
        ISO Type: host
```

```
rpms in host ISO:
    ncs5500-sysadmin-hostos-6.2.2-r622.host

Golden ISO Rpms:
    xr rpms in golden ISO:
        ncs5500-k9sec-x64-2.2.0.1-r622.CSCxr33333.x86_64.rpm
        openssh-scp-6.6p1.p1-r0.0.CSCtp12345.xr.x86_64.rpm
        openssh-scp-6.6p1-r0.0.xr.x86_64.rpm
        ncs5500-mpls-x64-2.1.0.0-r622.x86_64.rpm
        ncs5500-k9sec-x64-2.2.0.0-r622.x86_64.rpm

    sysadmin rpms in golden ISO:
        ncs5500-sysadmin-system-6.2.2-r622.CSCcv11111.x86_64.rpm
        ncs5500-sysadmin-system-6.2.2-r622.CSCcv11111.arm.rpm
        openssh-scp-6.6p1-r0.0.admin.x86_64.rpm
        openssh-scp-6.6p1-r0.0.admin.arm.rpm
        openssh-scp-6.6p1.p1-r0.0.CSCtp12345.admin.x86_64.rpm
        openssh-scp-6.6p1.p1-r0.0.CSCtp12345.admin.arm.rpm
        ncs5500-sysadmin-hostos-6.2.2-r622.CSCcv10001.admin.x86_64.rpm
        ncs5500-sysadmin-hostos-6.2.2-r622.CSCcv10001.admin.arm.rpm

    host rpms in golden ISO:
        openssh-scp-6.6p1-r0.0.host.x86_64.rpm
        openssh-scp-6.6p1-r0.0.host.arm.rpm
        openssh-scp-6.6p1.p1-r0.0.CSCtp12345.host.x86_64.rpm
        openssh-scp-6.6p1.p1-r0.0.CSCtp12345.host.arm.rpm
```

The ISO, SMUs and packages in GISO are installed on the router.