



System Setup and Software Installation Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.1.x

First Published: 2020-01-29

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface vii

Obtaining Documentation and Submitting a Service Request vii

Changes to This Document vii

CHAPTER 1

New and Changed Feature Information 1

New and Changed System Setup Features 1

CHAPTER 2

Cisco ASR 9000 System Features 3

Cisco ASR 9000 Product Overview 3

Virtual Machine based Routing and System Administration 4

Command Modes 5

CHAPTER 3

Bring-up the Router 7

Boot the Router 7

Boot the Router Using USB 8

Boot the Router Using iPXE 10

Setup Root User Credentials 12

Access the System Admin Console 14

Configure the Management Port 15

Perform Clock Synchronization with NTP Server 16

CHAPTER 4

Perform Preliminary Checks 19

Verify Software Version 19

Verify Active VMs 20

Verify Status of Hardware Modules 22

Verify Firmware Version 22

Verify SDR Information 24

Verify Interface Status 26

CHAPTER 5 **Create User Profiles and Assign Privileges 27**

Create User Groups 28

Configure User Groups in XR VM 29

Create a User Group in System Admin VM 30

Create Users 32

Create a User Profile in XR VM 32

Create a User Profile in System Admin VM 34

Create Command Rules 36

Create Data Rules 38

Change Disaster-recovery Username and Password 41

Recover Password using PXE Boot 42

CHAPTER 6 **Perform System Upgrade and Install Feature Packages 43**

Upgrading the System 43

Upgrading Features 44

Workflow for Install Process 44

Install Packages 44

Install Prepared Packages 48

Uninstall Packages 51

CHAPTER 7 **Manage Automatic Dependency 53**

Update RPMs and SMUs 54

Upgrade Base Software Version 54

Downgrade an RPM 55

CHAPTER 8 **Customize Installation using Golden ISO 59**

Limitations 59

Golden ISO Workflow 60

Build Golden ISO 61

Build Golden ISO Using Script 61

Install Golden ISO 64

Install Replace with Golden ISO	67
---------------------------------	----



Preface

This preface contains these sections:

- [Obtaining Documentation and Submitting a Service Request, on page vii](#)
- [Changes to This Document, on page vii](#)

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation*, at: <http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.

Changes to This Document

This table lists the technical changes made to this document since it was first released.

Date	Summary
January 2020	Initial release of this document



CHAPTER 1

New and Changed Feature Information

This table summarizes the new and changed feature information for the *System Setup and Software Installation Guide for Cisco ASR 9000 Series Routers* .

- [New and Changed System Setup Features, on page 1](#)

New and Changed System Setup Features

Feature	Description	Changed in Release	Where Documented
None	No new features introduced	Not applicable	Not applicable



CHAPTER 2

Cisco ASR 9000 System Features

The topics covered in this chapter are:

- [Cisco ASR 9000 Product Overview, on page 3](#)
- [Virtual Machine based Routing and System Administration, on page 4](#)
- [Command Modes, on page 5](#)

Cisco ASR 9000 Product Overview

The Cisco ASR 9000 series routers are next-generation edge access routers that are optimized for service provider applications. These routers are designed to fulfill various roles in:

- Layer 2 and Layer 3 Ethernet aggregation
- Subscriber-aware broadband aggregation

The Cisco ASR 9000 series routers meet carrier-class requirements for redundancy, availability, packaging, power, and other requirements traditional to the service provider.

The Cisco ASR 9000 series consists of the following routers:

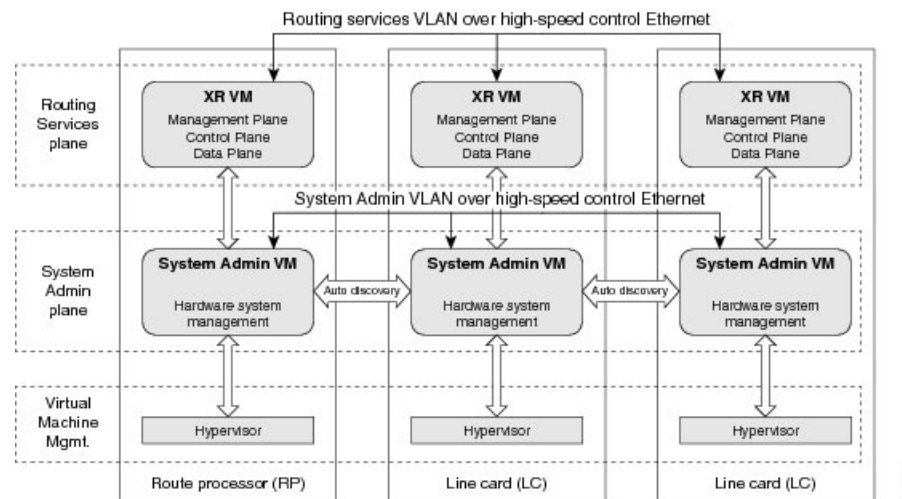
- Cisco ASR 9001 Router (32-bit)
- Cisco ASR 9001-S Router
- Cisco ASR 9006 Router
- Cisco ASR 9010 Router
- Cisco ASR 9901 Router
- Cisco ASR 9904 Router
- Cisco ASR 9906 Router
- Cisco ASR 9910 Router
- Cisco ASR 9912 Router
- Cisco ASR 9922 Router

Virtual Machine based Routing and System Administration

On the Cisco ASR 9000 series router running 64-bit IOS XR, the routing functions and the System Administration functions are run on separate virtual machines (VMs) over a Linux host operating system. The VMs simulate individual physical computing environments over a common hardware. Available hardware resources like processor, memory, hard disk, and so on, are virtualized and allocated to individual virtual machines by the hypervisor.

The VM topology on the Cisco ASR 9000 series router running 64-bit IOS XR is shown in this figure.

Figure 1: Virtualized IOS XR on Cisco ASR 9000 Series Router



Implementation of Virtualized IOS XR on Cisco ASR 9000 Series Router

- The hypervisor creates and manages individual VM environments.
- On every route processor (RP) there are two VMs; one for system administration (System Admin VM) and one for managing the routing functions (XR VM).
- The two VMs on each node operate on their respective planes. On each plane, the VMs are connected to each other using a dedicated VLAN over a high-speed Control Ethernet connection.
- The System Admin VMs can detect each other's presence by auto discovery and thus maintain complete system awareness.

To access the XR VM, connect to the XR VM console port on the RP. To access the System Admin VM, in the XR VM CLI, execute the **admin** command.



Note

In 32-bit IOS XR OS, the management interfaces are available from XR VM. In 64-bit IOS XR OS, the Management ports on the RP/RSP are available as follows:

- MGT LAN 0 is available in XR VM.
- MGT LAN 1 is available in Admin VM.

Advantages of Virtualized IOS XR on the Router

- **Faster boot time**—Because the System Admin functions are on a dedicated VM, the boot time is considerably reduced.
- **Independent upgrades**—Software packages can be independently installed on the System Admin VM and the XR VM, resulting in minimal system downtime.
- **Self-starting VMs**—Both the System Admin VM and the XR VM are automatically launched during router boot-up without any user intervention. They have a default set-up that is ready for use.
- **System redundancy**—In spite of their interconnectivity, there is also a level of isolation between the VMs. Therefore, if a particular VM experiences any issues, it does not affect the functioning of other VMs.

Command Modes

This table lists the command modes:

Command Mode	Description
XR VM Execution Mode	Run commands on the XR VM to display the operational state of the router. Example: RP/0/RP0/CPU0:router#
XR VM Global Configuration	Perform security, routing, and other XR feature configurations on the XR VM. Example: RP/0/RP0/CPU0:router#configure RP/0/RP0/CPU0:router(config)#
System Admin VM Execution Mode	Run commands on the System Admin VM to display and monitor the operational state of the router hardware. The chassis or individual hardware modules can be reloaded from this mode. Example: RP/0/RP0/CPU0:router#admin sysadmin-vm:0_RP0#
System Admin VM Configuration Mode	Run configuration commands on the System Admin VM to manage and operate the hardware modules of the entire chassis. Example: RP/0/RP0/CPU0:router#admin sysadmin-vm:0_RP0#config sysadmin-vm:0_RP0(config)#



CHAPTER 3

Bring-up the Router

After installing the hardware, boot the router. Connect to the XR console port and power on the router. The router completes the boot process using the pre-installed operating system (OS) image. If no image is available within the router, the router can be booted using iPXE boot or an external bootable USB drive.

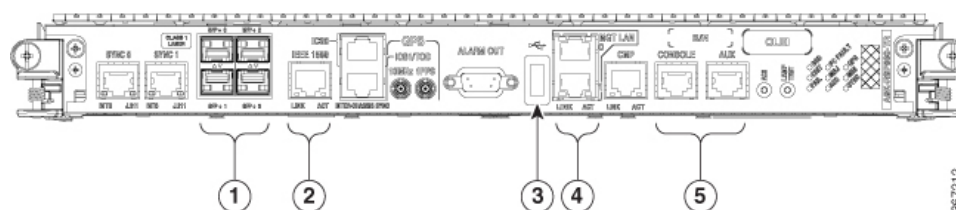
After booting is complete, create the root username and password, and then use it to log on to the XR console and get the router prompt. The first user created in XR console is synchronized to the System Admin console. From the XR console, access the System Admin console to configure system administration settings.

For more information about completing the hardware installation, see [Cisco ASR 9000 Series Aggregation Services Router Hardware Installation Guide](#).

- [Boot the Router, on page 7](#)
- [Boot the Router Using USB, on page 8](#)
- [Boot the Router Using iPXE, on page 10](#)
- [Setup Root User Credentials, on page 12](#)
- [Access the System Admin Console, on page 14](#)
- [Configure the Management Port, on page 15](#)
- [Perform Clock Synchronization with NTP Server, on page 16](#)

Boot the Router

Use the console port on the Route Processor (RP) to connect to a new router. The console port connect to the XR console by default. If required, subsequent connections can be established through the management port, after it is configured.



1	SFP/SFP+ ports
2	Service LAN port
3	External USB port

4	Management LAN ports
5	Console and Auxiliary (AUX) ports

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

Step 2 Start the terminal emulation program on your workstation.

For chassis with RSP4, RP2 cards, the console settings are baud rate 9600 bps, no parity, 2 stop bits and 8 data bits. The user can change this baud rate. For next generation RP3, RSP5 cards, the console settings are baud rate 115200 bps, no parity, 2 stop bits and 8 data bits.

Step 3 Power on the router.

Connect the power cord to Power Entry Module (PEM) and the router boots up. The boot process details is displayed on the console screen of the terminal emulation program.

Step 4 Press **Enter**.

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

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

What to do next

Specify the root username and password.

Boot the Router Using USB

The bootable USB drive is used to re-image the router for the purpose of system upgrade, password recovery or boot the router in case of boot failure. The USB on router is mounted as disk 2.

Before you begin

Ensure you have completed the following prerequisites:

- You have access to a USB drive with a storage capacity that is between 8GB (min) and 32 GB (max). USB 2.0 and USB 3.0 are supported.
- Copy the compressed boot file from the [Software Download Center](#) to your local machine. The file name for the compressed boot file is in the format asr9k-x64-usb_boot-<release_number>.zip.

Step 1 Create a bootable USB drive.

Note The content of the zipped file ("EFI" and "boot" directories) should be extracted directly into root of the USB drive. If the unzipping application places the extracted files in a new folder, move the "EFI" and "boot" directories to root of the USB drive.

- Connect the USB drive to your local machine and format it with FAT32 or MS-DOS file system using the Windows Operating System or Apple MAC Disk Utility.
- Copy the `asr9k-x64-usb_boot-<release_number>.zip` compressed boot file to the USB drive.
- Verify that the copy operation is successful. To verify, compare the file size at source and destination. Additionally, verify the MD5 checksum value.
- Extract the content of the compressed boot file by unzipping it inside the USB drive. This converts the USB drive to a bootable drive.
- Eject the USB drive from your local machine.

Step 2 Insert the USB on the active RP, and reload or reset the power of the router.

Note Use this procedure only on active RP; the standby RP must either be removed from the chassis, or stopped at the boot menu. After the active RP is installed with images from USB, boot the standby RP.

Step 3 On active XR console, press CTRL-C to view BIOS menu. From the menu, select IOS-XR 64 bit Local boot using front panel USB media.

```

Got EMT Mode as Disk Boot
Set OS type None, Received OS type=0
Got Boot Mode as Disk Boot

Booting IOS-XR 64 bit Boot previously installed image - Press Ctrl-c to stop

Please select the operating system and the boot device:
  1) Boot to ROMMON
  2) IOS-XR 64 bit Boot previously installed image
  3) IOS-XR 64 bit Mgmt Network boot using DHCP server
  4) IOS-XR 64 bit Mgmt Network boot using local settings (iPXE)
  (Press 'p' for more option)
Selection [1/2/3/4]: p
Please select the operating system and the boot device:
  1) Boot to ROMMON
  2) IOS-XR 64 bit Boot previously installed image
  3) IOS-XR 64 bit Mgmt Network boot using DHCP server
  4) IOS-XR 64 bit Mgmt Network boot using local settings (iPXE)
  5) IOS-XR 64 bit Internal network boot from RSP/RP
  6) IOS-XR 64 bit Local boot using embedded USB media
  7) IOS-XR 64 bit Local boot using front panel USB media
  8) Change baud rate and continue booting
Selection [1/2/3/4/5/6/7/8]: 7
Selected IOS-XR 64 bit Local boot using front panel USB media, Continue ? Y/N: y

Set CBC OS type IOS-XR 64 bit, EMT USB Boot to CBC
Sending boot success notification

Selected boot option - EFI USB Device 1 (SanDisk Cruzer)
Verifying image signature...
Image signature verified successfully
Image Verification Passed

```

522185

If active and standby RPs are not stopped at the boot menu, the previously used boot option is used. If the system is inactive in the boot menu for 30 minutes, the system resets automatically.

Step 4 If standby RP is present and it was stopped in step 2, boot the standby RP after the active RP starts to boot. From the boot options select IOS-XR 64 bit Internal network boot from RSP/RP.

Example:

```

Please select the operating system and the boot device:
  1) IOS-XR (32 bit Classic XR)
  2) IOS-XR 64 bit Boot previously installed image

```

```

3) IOS-XR 64 bit Mgmt Network boot using DHCP server
4) IOS-XR 64 bit Mgmt Network boot using local settings (iPXE)
5) IOS-XR 64 bit Internal network boot from RSP/RP
6) IOS-XR 64 bit Local boot using embedded USB media
7) IOS-XR 64 bit Local boot using front panel USB media
Selection [1/2/3/4/5/6/7]:

```

Select option 5 and proceed with the boot up. After the router boots up, specify the root username and password.

Boot the Router Using iPXE

iPXE is a pre-boot execution environment that is included in the network card of the management interfaces and works at the system firmware (UEFI) level of the router. iPXE is used to re-image the system, and boot the router in case of boot failure or in the absence of a valid bootable partition. iPXE downloads the ISO image, proceeds with the installation of the image, and bootstraps within the new installation.

iPXE acts as a boot loader and provides the flexibility to choose the image that the system will boot based on the Platform Identifier (PID), the Serial Number, or the management mac-address. iPXE must be defined in the DHCP server configuration file.



Note

PID and serial number is supported only if iPXE is invoked using the command (admin) `hw-module location all bootmedia network reload all`. If iPXE is selected manually from BIOS, PID and serial number is not supported.



Note

Cisco ASR 9901 — By default, iPXE uses the previous attempted boot method on the next reload. If the Network option was previously used, the iPXE register will be set to 1 (IPXE_PREF=1). To boot an Cisco ASR 9901 router via DHCP on the next reload, you must set the IPXE_PREF settings to 0 (IPXE_PREF=0).

From the system admin console, enter the **run chvrf 0 ssh rp0_admin /opt/cisco/calvados/bin/nvram_dump -s IPXE_PREF=0** command twice. After entering this command the first time, the host is added to the known list of hosts.

```

sysadmin-vm:0_RP0# run chvrf 0 ssh rp0_admin /opt/cisco/calvados/bin/nvram_dump -s IPXE_PREF=0
Sat May  2 10:39:52.740 UTC+00:00
Warning: Permanently added 'rp0_admin' (ECDSA) to the list of known hosts.
sysadmin-vm:0_RP0# run chvrf 0 ssh rp0_admin /opt/cisco/calvados/bin/nvram_dump -s IPXE_PREF=0
Sat May  2 10:39:54.995 UTC+00:00
sysadmin-vm:0_RP0# hw-module location all bootmedia network

```

iPXE boot can be performed during the following scenarios:

- migration from 32-bit to 64-bit using migration script
- recover password
- boot-up failure with 64-bit image

Before you begin

Take a backup of configuration to a TFTP or FTP path to load the configuration back after the iPXE boot.

Step 1 Login to the system admin console.

Example:

```
sysadmin-vm:0_RSP0# hw-module location all reload
Tue Mar  6 08:12:47.605 UTC
Reload hardware module ? [no,yes] yes
result Card graceful reload request on all acknowledged.
sysadmin-vm:0_RSP0#
```

Step 2 If the router is unable to boot, press Ctrl +C to stop the boot process when the following information is displayed.

Note Use this procedure only on active RP; the standby RP must either be removed from the chassis, or stopped at the boot menu. After the active RP is installed with images from iPXE boot, boot the standby RP.

Example:

```
System Bootstrap, Version 10.57 [ASR9K x86 ROMMON],
Copyright (c) 1994-2018 by Cisco Systems, Inc.
Compiled on Mon 01/09/2017 17:15:01.98
BOARD_TYPE           : 0x100317
Rommon               : 10.57 (Primary)
Board Revision       : 4
PCH EEPROM           : 3.4
IPU FPGA(PL)         : 0.40.0 (Backup)
IPU INIT(HW_FPD)     : 0.30.0
IPU FSBL(BOOT.BIN)   : 1.19.0
IPU LINUX(IMAGE.FPD) : 1.21.0
OPTIMUS FPGA         : 0.12.0
OMEGA FPGA           : 0.13.0
ALPHA FPGA           : 0.14.0
CHA FPGA             : 0.5.1
CBC0                 : Part 1=34.38, Part 2=34.38, Act Part=2
Product Number       : A9K-RSP880-SE
Chassis              : ASR-9904-AC
Chassis Serial Number : FOX1936GBDD
Slot Number          : 1
Pxe Mac Address LAN 0 : 70:e4:22:06:13:40
Pxe Mac Address LAN 1 : 70:e4:22:06:13:41
=====
Got EMT Mode as 3
Got Boot Mode as 0
Booting IOS-XR (32 bit Classic XR) - Press Ctrl-c to stop
```

Step 3 Choose option 4 for iPXE boot.

Example:

```
Please select the operating system and the boot device:
 1) IOS-XR (32 bit Classic XR)
 2) IOS-XR 64 bit Boot previously installed image
 3) IOS-XR 64 bit Mgmt Network boot using DHCP server
 4) IOS-XR 64 bit Mgmt Network boot using local settings (iPXE)
 5) IOS-XR 64 bit Internal network boot from RSP/RP
 6) IOS-XR 64 bit Local boot using embedded USB media
 7) IOS-XR 64 bit Local boot using front panel USB media
Selection [1/2/3/4/5/6/7]:
```

Step 4 Manually update iPXE ROMMON details before booting using FTP or TFTP.

Example:

```
iPXE>set cisco/cisco-server-url:string tftp://<path>/asr9k-mini-x64.iso
iPXE>set cisco/cisco-ipv4-address:string 1.3.24.202
iPXE>set cisco/cisco-netmask-address:str 255.255.0.0
iPXE>set cisco/cisco-gateway-address:str 1.3.0.1
```

Step 5 Open the connected management port (0/1).

Example:

```
iPXE>ifclose net0
iPXE>ifclose net1
iPXE>ifopen net1
```

where net0 and net1 represents management port0 and port1 respectively.

Step 6 Boot the required image from FTP or TFTP location.

Example:

```
iPXE>
iPXE> ifopen net0:
iPXE> boot tftp://<path>/asr9k-mini-x64-<release-number>.iso
tftp://<path>/asr9k-mini-x64-<release-number>.iso... 0%
Booting iso-image@0x83c525000(1135456256), bzImage@0x83c55f000(4526671)

.....BIOS CODE SIGN ENTRY ...
```

Step 7 After the active RP is up and running, boot the standby RP. From the boot options select IOS-XR 64 bit Internal network boot from RSP/RP.

Example:

```
Please select the operating system and the boot device:
 1) IOS-XR (32 bit Classic XR)
 2) IOS-XR 64 bit Boot previously installed image
 3) IOS-XR 64 bit Mgmt Network boot using DHCP server
 4) IOS-XR 64 bit Mgmt Network boot using local settings (iPXE)
 5) IOS-XR 64 bit Internal network boot from RSP/RP
 6) IOS-XR 64 bit Local boot using embedded USB media
 7) IOS-XR 64 bit Local boot using front panel USB media
Selection [1/2/3/4/5/6/7]:
```

Setup Root User Credentials

When the router boots for the first time, the system prompts the user to configure root credentials (username and password). These credentials are configured as the root user on the XR (root-lr) console, the System Admin VM (root-system), and as disaster-recovery credentials.

Before you begin

The boot process must be complete. For details on how to initiate the boot process, see [Bring-up the Router, on page 7](#).

SUMMARY STEPS

1. Enter root-system username: *username*

2. **Enter secret:** *password*
3. **Enter secret again:** *password*
4. **Username:** *username*
5. **Password:** *password*
6. (Optional) **show run username**

DETAILED STEPS

Step 1 **Enter root-system username:** *username*

Enter the username of the root user. The character limit is 1023. In this example, the name of the root user is "root".

Important The specified username is mapped to the "root-lr" group on the XR console. It is also mapped as the "root-system" user on the System Admin console.

When starting the router for the first time, or after a re-image, the router does not have any user configuration. In such cases, the router prompts you to specify the "root-system username". However, if the router has been configured previously, the router prompts you to enter the "username", as described in Step 4.

Step 2 **Enter secret:** *password*

Enter the password for the root user. The character range of the password is between 6 and 253 characters. The password you type is not displayed on the CLI for security reasons.

The root username and password must be safeguarded as it has the superuser privileges. It is used to access the complete router configuration.

Step 3 **Enter secret again:** *password*

Re-enter the password for the root user. The password is not accepted if it does not match the password entered in the previous step. The password you type is not displayed on the CLI for security reasons.

Step 4 **Username:** *username*

Enter the root-system username to login to the XR VM console.

Step 5 **Password:** *password*

Enter the password of the root user. The correct password displays the router prompt. You are now logged into the XR VM console.

Step 6 (Optional) **show run username**

Displays user details.

```
username root
group root-lr
group cisco-support
secret 5 $1$NBg7$fHs1inKPZVvzqxMv775UE/
!
```

What to do next

- Configure routing functions from the XR console.

- Configure system administration settings from the System Admin prompt. The System Admin prompt is displayed on accessing the System Admin console. For details on how to get the System Admin prompt, see [Access the System Admin Console, on page 14](#).

Access the System Admin Console

You must login to the System Admin console through the XR console to perform all system administration and hardware management setups.

SUMMARY STEPS

1. Login to the XR console as the root user.
2. (Optional) Disable the login banner on console port when accessing the System Admin mode from XR mode.
 - a) **configure**
 - b) **service sysadmin-login-banner disable**
3. **admin**
4. (Optional) **exit**

DETAILED STEPS

Step 1 Login to the XR console as the root user.

Step 2 (Optional) Disable the login banner on console port when accessing the System Admin mode from XR mode.

- a) **configure**
- b) **service sysadmin-login-banner disable**

Example:

```
RP/0/RSP0/CPU0:router(config)#service sysadmin-login-banner disable
```

Disable the login banner on console port in System Admin mode.

- c) **commit**
- d) **end**

Step 3 **admin**

Example:

The login banner is enabled by default. The following example shows the command output with the login banner enabled:

```
RP/0/RSP0/CPU0:router#admin
```

```
Mon May 22 06:57:29.350 UTC
```

```
root connected from 127.0.0.1 using console on host
```

```
sysadmin-vm:0_RP0# exit
```

```
Mon May 22 06:57:32.360 UTC
```

The following example shows the command output with the login banner disabled:

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

```
Thu Mar 01:07:14.509 UTC
```

```
sysadmin-vm:0_RP0# exit
```

Step 4 (Optional) **exit**

Return to the XR mode from the System Admin mode.

Configure the Management Port

To use the Management port for system management and remote communication, you must configure an IP address and a subnet mask for the management ethernet interface. To communicate with devices on other networks (such as remote management stations or TFTP servers), you need to configure a default (static) route for the router.

Before you begin

- Consult your network administrator or system planner to procure IP addresses and a subnet mask for the management interface.
- Physical port Ethernet 0 and Ethernet 1 on RP are the management ports. Ensure that the port is connected to management network.

SUMMARY STEPS

1. **configure**
2. **interface MgmtEth** *rack/slot/port*
3. **ipv4 address** *ipv4-address subnet-mask*
4. **ipv4 address** *ipv4 virtual address subnet-mask*
5. **no shutdown**
6. **exit**
7. **router static address-family ipv4 unicast** *0.0.0.0/0 default-gateway*
8. Use the **commit** or **end** command.

DETAILED STEPS

Step 1 **configure**

Example:

```
RP/0/RSP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **interface MgmtEth** *rack/slot/port*

Example:

```
RP/0/RSP0/CPU0:router(config)#interface mgmtEth 0/RP0/CPU0/0
```

Enters interface configuration mode for the management interface of the primary RP.

Step 3 **ipv4 address** *ipv4-address subnet-mask*

Example:

```
RP/0/RSP0/CPU0:router(config-if)#ipv4 address 10.1.1.1/8
```

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

Step 4 **ipv4 address** *ipv4 virtual address subnet-mask*

Example:

```
RP/0/RSP0/CPU0:router(config-if)#ipv4 address 1.70.31.160 255.255.0.0
```

Assigns a virtual IP address and a subnet mask to the interface.

Step 5 **no shutdown**

Example:

```
RP/0/RSP0/CPU0:router(config-if)#no shutdown
```

Places the interface in an "up" state.

Step 6 **exit**

Example:

```
RP/0/RSP0/CPU0:router(config-if)#exit
```

Exits the Management interface configuration mode.

Step 7 **router static address-family ipv4 unicast** *0.0.0.0/0 default-gateway*

Example:

```
RP/0/RSP0/CPU0:router(config)#router static address-family ipv4 unicast 0.0.0.0/0 12.25.0.1
```

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

Step 8 Use the **commit** or **end** command.

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

end —Prompts user to take one of these actions:

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

What to do next

Connect to the management port to the ethernet network. With a terminal emulation program, establish a SSH or telnet connection to the management interface port using its IP address. Before establishing a telnet session, use the **telnet ipv4|ipv6 server max-servers** command in the XR Config mode, to set number of allowable telnet sessions to the router.

Perform Clock Synchronization with NTP Server

There are independent system clocks for the XR console and the System Admin console. To ensure that these clocks do not deviate from true time, they need to be synchronized with the clock of a NTP server. In this

task you will configure a NTP server for the XR console. After the XR console clock is synchronized, the System Admin console clock will automatically synchronize with the XR console clock.

Before you begin

Configure and connect to the management port.

SUMMARY STEPS

1. **configure**
2. **ntp server** *server_address*

DETAILED STEPS

Step 1 **configure**

Example:

```
RP/0/RSP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **ntp server** *server_address*

Example:

```
RP/0/RSP0/CPU0:router(config)#ntp server 64.90.182.55
```

The XR console clock is configured to be synchronized with the specified sever.



CHAPTER 4

Perform Preliminary Checks

After successfully logging into the console, you must perform some preliminary checks to verify the default setup. If any setup issue is detected when these checks are performed, take corrective action before making further configurations. These preliminary checks are:

- [Verify Software Version, on page 19](#)
- [Verify Active VMs, on page 20](#)
- [Verify Status of Hardware Modules, on page 22](#)
- [Verify Firmware Version, on page 22](#)
- [Verify SDR Information, on page 24](#)
- [Verify Interface Status, on page 26](#)

Verify Software Version

The router is shipped with the Cisco IOS XR software pre-installed. Verify that the latest version of the software is installed. If a newer version is available, perform a system upgrade. This will install the newer version of the software and provide the latest feature set on the router.

Perform this task to verify the version of Cisco IOS XR software running on the router.

SUMMARY STEPS

1. **show version**

DETAILED STEPS

show version

Example:

```
RP/0/RSP0/CPU0:router# show version
```

Displays the version of the various software components installed on the router. The result includes the version of Cisco IOS XR software and its various components.

Example**What to do next**

Verify the result to ascertain whether a system upgrade or additional package installation is required. If that is required, refer to the tasks in the chapter [Perform System Upgrade and Install Feature Packages](#), on page 43.

Verify Active VMs

On the router both the XR VM and the System Admin VM must be operational. Instances of both VMs should be running on every route processor (RP). Complete this task to verify the VMs are active.

SUMMARY STEPS

1. **show redundancy summary**
2. **admin**
3. **show vm**

DETAILED STEPS**Step 1 show redundancy summary****Example:**

```
RP/0/RP0:hostname#show redundancy summary
Mon Mar 9 16:32:19.276 IST
Active Node Standby Node
-----
0/RP0 0/RP1 (Node Ready, NSR: Not Configured)
0/LC0 0/LC1 (Node Ready, NSR: Not Configured)
RP/0/RP0:hostname#
```

Displays the readiness of the VMs.

Step 2 admin**Example:**

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 3 show vm**Example:**

```
sysadmin-vm:0_RP0#show vm
```

Displays the status of the VMs running on various nodes.

```
sysadmin-vm:0_RP0# sh vm
Mon Mar 9 07:52:06.173 UTC
----- VMs found at location 0/RP0 -----
Id : sysadmin
```

```

Status : running
IP Addr: 192.0.44.1
HB Interval : NA
Last HB Sent: NA
Last HB Rec : NA
-----
Id : default-sdr
Status : running
IP Addr: 192.0.44.4
HB Interval : 0 s 500000000 ns
Last HB Sent: 663743
Last HB Rec : 663743
-----
Id : default-sdr
Status : running
IP Addr: 192.0.44.6
HB Interval : 10 s 0 ns
Last HB Sent: 33183
Last HB Rec : 33183
-----
----- VMs found at location 0/RP1 -----
Id : sysadmin
Status : running
IP Addr: 192.0.88.1
HB Interval : NA
Last HB Sent: NA
Last HB Rec : NA
-----
Id : default-sdr
Status : running
IP Addr: 192.0.88.4
HB Interval : 0 s 500000000 ns
Last HB Sent: 663749
Last HB Rec : 663749
-----
Id : default-sdr
Status : running
IP Addr: 192.0.88.6
HB Interval : 10 s 0 ns
Last HB Sent: 33183
Last HB Rec : 33183
-----
sysadmin-vm:0_RP0#

```

In the above result:

- Id—Name of the VM. "sysadmin" represents System Admin VM; "default-sdr" represents XR VM.
- Status—Status of the VM
- IP Addr—Internal IP address of the VM

If a VM is not running on a node, in the output of the **show vm** command, no output is shown for that node.

What to do next

If the XR VM is not running on a node, try reloading the node. To do so, use the **hw-module location node-id reload** command in the `node` mode. Also, use the **show sdr** command in the `node` mode to verify that the SDR is running on the node.

Verify Status of Hardware Modules

Hardware modules include RPs, fan trays, and so on. On the router, multiple hardware modules are installed. Perform this task to verify that all hardware modules are installed correctly and are operational.

Before you begin

Ensure that all required hardware modules have been installed on the router.

SUMMARY STEPS

1. `show platform`

DETAILED STEPS

show platform

Example:

```
sysadmin-vm:0_RP0# show platform
```

Displays the list of hardware modules detected on the router.

```
Mon Jul 29 10:59:22.163 UTC+00:00
Location  Card Type          HW State    SW State    Config State
-----
0/RP0     N560-RSP4-E        OPERATIONAL OPERATIONAL NSHUT
0/FT0     A907-FAN-E         OPERATIONAL N/A         NSHUT
0/PM2     A900-PWR1200-A     OPERATIONAL N/A         NSHUT
```

Verify Firmware Version

The firmware on various hardware components of the router must be compatible with the Cisco IOS XR image installed. Incompatibility might cause the router to malfunction. Complete this task to verify the firmware version.

SUMMARY STEPS

1. `show hw-module fpd`

DETAILED STEPS

show hw-module fpd

Example:

Displays the list of hardware modules detected on the router.

Note This command can be run from both XR VM and System Admin VM modes.

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

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

What to do next

- Upgrade the required firmware by using the **upgrade hw-module location all fpd** command in the EXEC mode. For the FPD upgrade to take effect, the router needs a power cycle.
- It is recommended to upgrade all FPGAs on a given node using the **upgrade hw-module fpd all location {all | node-id}** command. Do not upgrade the FPGA on a node using the **upgrade hw-module fpd <individual-fpd> location {all | node-id}** as it may cause errors in booting the card.
- If required, turn on the auto fpd upgrade function. To do so, use the **fpd auto-upgrade enable** command in the XR configuration [(config)#] mode. After it is enabled, if there are new FPD binaries present in the image being installed on the router, FPDs are automatically upgraded during the system upgrade operation.

Verify SDR Information

Secure domain routers (SDRs) divide a single physical system into multiple logically-separated routers. SDRs are also known as logical routers (LRs). On the router, only one SDR is supported. This SDR is termed the default-sdr. Every router is shipped with the default-sdr, which owns all RPs installed in the routing system. An instance of this SDR runs on all nodes. Complete this task to verify the details of the SDR instances.

SUMMARY STEPS

1. **admin**
2. **show sdr**

DETAILED STEPS

Step 1 admin

Example:

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 2 show sdr

Example:

```
sysadmin-vm:0_RP0# show sdr
```

Displays the SDR information for every node.

```
sysadmin-vm:0_RP0# show sdr

sdr default-sdr
location 0/0/VM1
sdr-id          2
IP Address of VM 192.0.4.3
MAC address of VM A4:6C:2A:2B:AA:A6
VM State         RUNNING
start-time       2015-12-03T15:38:38.74514+00:00
Last Reload Reason "SMU:Reboot triggered by install"
Reboot Count     2
location 0/1/VM1
sdr-id          2
IP Address of VM 192.0.8.3
MAC address of VM B0:AA:77:E7:5E:DA
VM State         RUNNING
start-time       2015-12-03T15:38:39.730036+00:00
Last Reload Reason "SMU:Reboot triggered by install"
Reboot Count     2
location 0/2/VM1
sdr-id          2
IP Address of VM 192.0.12.3
MAC address of VM B0:AA:77:E7:67:34
VM State         RUNNING
start-time       2015-12-03T15:38:38.886947+00:00
Last Reload Reason "SMU:Reboot triggered by install"
Reboot Count     2
location 0/3/VM1
```



```

sdr-id                2
IP Address of VM      192.0.16.3
MAC address of VM     B0:AA:77:E7:58:86
VM State              RUNNING
start-time            2015-12-03T15:38:40.391205+00:00
Last Reload Reason    "SMU:Reboot triggered by install"
Reboot Count          2
location 0/4/VM1
sdr-id                2
IP Address of VM      192.0.20.3
MAC address of VM     B0:AA:77:E7:46:C2
VM State              RUNNING
start-time            2015-12-03T15:38:39.84469+00:00
Last Reload Reason    "SMU:Reboot triggered by install"
Reboot Count          2
location 0/5/VM1
sdr-id                2
IP Address of VM      192.0.24.3
MAC address of VM     B0:AA:77:E7:84:40
VM State              RUNNING
start-time            2015-12-04T03:48:24.017443+00:00
Last Reload Reason    "VM_REQUESTED_UNGRACEFUL_RELOAD:Headless SDR"
Reboot Count          3
location 0/6/VM1
sdr-id                2
IP Address of VM      192.0.28.3
MAC address of VM     B0:AA:77:E7:55:FE
VM State              RUNNING
start-time            2015-12-03T15:38:38.74753+00:00
Last Reload Reason    "SMU:Reboot triggered by install"
Reboot Count          2
location 0/7/VM1
sdr-id                2
IP Address of VM      192.0.32.3
MAC address of VM     B0:AA:77:E7:60:C6
VM State              RUNNING
start-time            2015-12-03T15:38:38.691481+00:00
Last Reload Reason    "SMU:Reboot triggered by install"
Reboot Count          2
location 0/RP0/VM1
sdr-id                2
IP Address of VM      192.0.108.4
MAC address of VM     10:05:CA:D7:FE:6F
VM State              RUNNING
start-time            2015-12-04T07:03:04.549294+00:00
Last Reload Reason    CARD_SHUTDOWN
Reboot Count          1
location 0/RP1/VM1
sdr-id                2
IP Address of VM      192.0.112.4
MAC address of VM     10:05:CA:D8:3F:43
VM State              RUNNING
start-time            2015-12-04T09:21:42.083046+00:00
Last Reload Reason    CARD_SHUTDOWN
Reboot Count          1

```

For a functional SDR, the VM State is "RUNNING". If the SDR is not running on a node, no output is shown in the result, for that location. At times the node performs a core dump. During such times the VM State is "Paused & Core Dump in Progress".

What to do next

If you find SDR is not running on a node, try reloading the node. To do that, use the **hw-module location node-id reload** command in the `mode`.

Verify Interface Status

After the router has booted, all available interfaces must be discovered by the system. If interfaces are not discovered, it might indicate a malfunction in the unit. Complete this task to view the number of discovered interfaces.

SUMMARY STEPS

1. **show ipv4 interface summary**

DETAILED STEPS

show ipv4 interface summary**Example:**

```
RP/0/RSP0/CPU0:router#show ipv4 interface summary
```

When a router is turned on for the first time, all interfaces are in the 'unassigned' state. Verify that the total number of interfaces displayed in the result matches with the actual number of interfaces present on the router.

In the above result:

- Assigned— An IP address is assigned to the interface.
- Unnumbered— Interface which has borrowed an IP address already configured on one of the other interfaces of the router.
- Unassigned—No IP address is assigned to the interface.

You can also use the **show interfaces brief** and **show interfaces summary** commands in the `mode` to verify the interface status.



CHAPTER 5

Create User Profiles and Assign Privileges

To provide controlled access to the XR and System Admin configurations on the router, user profiles are created with assigned privileges. The privileges are specified using command rules and data rules. The authentication, authorization, and accounting (aaa) commands are used for the creation of users, groups, command rules, and data rules. The `aaa` commands are also used for changing the disaster-recovery password.



Note You cannot configure the external AAA server and services from the System Admin VM. It can be configured only from the XR VM.

Configure AAA authorization to restrict users from uncontrolled access. If AAA authorization is not configured, the command and data rules associated to the groups that are assigned to the user are bypassed. An IOS-XR user can have full read-write access to the IOS-XR configuration through Network Configuration Protocol (NETCONF), google-defined Remote Procedure Calls (gRPC) or any YANG-based agents. In order to avoid granting uncontrolled access, enable AAA authorization before setting up any configuration.



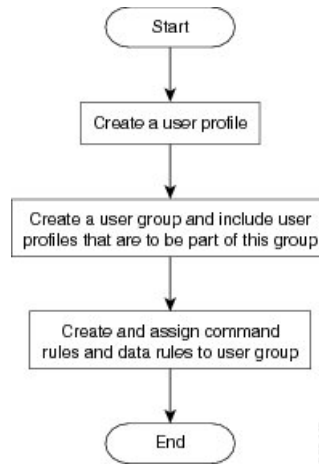
Note If any user on XR is deleted, the local database checks whether there is a first user on System Admin VM.

- If there is a first user, no syncing occurs.
- If there is no first user, then the first user on XR (based on the order of creation) is synced to System Admin VM.
- When a user is added in XR, if there is no user on System Admin mode, then the user is synced to sysadmin-vm. After the synchronization, any changes to the user on XR VM does not synchronize on the System Admin VM.
- A user added on the System Admin VM does not synchronize with XR VM.
- Only the first user or disaster-recovery user created on System Admin VM synchronizes with the host VM.
- Changes to credentials of first user or disaster-recovery user on System Admin VM synchronizes with the host VM.
- The first user or disaster-recovery user deleted on System Admin VM does not synchronize with the host VM. The host VM retains the user.

Users are authenticated using username and password. Authenticated users are entitled to execute commands and access data elements based on the command rules and data rules that are created and applied to user groups. All users who are part of a user group have such access privileges to the system as defined in the command rules and data rules for that user group.

The workflow for creating user profile is represented in this flow chart:

Figure 2: Workflow for Creating User Profiles



Note

The root-lr user, created for the XR VM during initial router start-up, is mapped to the root-system user for the System Admin VM. The root-system user has superuser permissions for the System Admin VM and therefore has no access restrictions.

Use the **show run aaa** command in the Config mode to view existing aaa configurations.

The topics covered in this chapter are:

- [Create User Groups, on page 28](#)
- [Create Users, on page 32](#)
- [Create Command Rules, on page 36](#)
- [Create Data Rules, on page 38](#)
- [Change Disaster-recovery Username and Password, on page 41](#)
- [Recover Password using PXE Boot, on page 42](#)

Create User Groups

Create a new user group to associate command rules and data rules with it. The command rules and data rules are enforced on all users that are part of the user group.

For extensive information about creating user groups, task groups, RADIUS and TACACS configurations, see the *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco ASR 9000 Series Routers*. For detailed information about commands, syntax and their description, see the *Authentication, Authorization, and Accounting Commands* chapter in the *System Security Command Reference for Cisco ASR 9000 Series Routers*.

Configure User Groups in XR VM

User groups are configured with the command parameters for a set of users, such as task groups. Entering the **usergroup** command accesses the user group configuration submode. Users can remove specific user groups by using the **no** form of the **usergroup** command. Deleting a usergroup that is still referenced in the system results in a warning.

Before you begin



Note Only users associated with the WRITE:AAA task ID can configure user groups. User groups cannot inherit properties from predefined groups, such as owner-sdr.

SUMMARY STEPS

1. **configure**
2. **usergroup** *usergroup-name*
3. **description** *string*
4. **inherit usergroup** *usergroup-name*
5. **taskgroup** *taskgroup-name*
6. Repeat Step for each task group to be associated with the user group named in Step 2.
7. Use the **commit** or **end** command.

DETAILED STEPS

Step 1 **configure**

Example:

```
RP/0/RSP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **usergroup** *usergroup-name*

Example:

```
RP/0/RSP0/CPU0:router(config)# usergroup beta
```

Creates a name for a particular user group and enters user group configuration submode.

- Specific user groups can be removed from the system by specifying the **no** form of the **usergroup** command.

Step 3 **description** *string*

Example:

```
RP/0/RSP0/CPU0:router(config-ug)#  
description this is a sample user group description
```

(Optional) Creates a description of the user group named in Step 2.

Step 4 **inherit usergroup** *usergroup-name*

Example:

```
RP/0/RSP0/CPU0:router(config-ug)#
inherit usergroup sales
```

- Explicitly defines permissions for the user group.

Step 5 `taskgroup taskgroup-name`**Example:**

```
RP/0/RSP0/CPU0:router(config-ug)# taskgroup beta
```

Associates the user group named in Step 2 with the task group named in this step.

- The user group takes on the configuration attributes (task ID list and permissions) already defined for the entered task group.

Step 6 Repeat Step for each task group to be associated with the user group named in Step 2.

Step 7 Use the **commit** or **end** command.

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

end —Prompts user to take one of these actions:

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

Create a User Group in System Admin VM

Create a user group for the System Admin VM.

The router supports a maximum of 32 user groups.

Before you begin

Create a user profile. See the *Create User* section.

SUMMARY STEPS

1. **admin**
2. **config**
3. **aaa authentication groups group group_name**
4. **users user_name**
5. **gid group_id_value**
6. Use the **commit** or **end** command.

DETAILED STEPS

Step 1 **admin**

Example:

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 2 **config****Example:**

```
sysadmin-vm:0_RP0#config
```

Enters mode.

Step 3 **aaa authentication groups group *group_name*****Example:**

```
sysadmin-vm:0_RP0(config)#aaa authentication groups group gr1
```

Creates a new user group (if it is not already present) and enters the group configuration mode. In this example, the user group "gr1" is created.

Note By default, the user group "root-system" is created by the system at the time of root user creation. The root user is part of this user group. Users added to this group will get root user permissions.

Step 4 **users *user_name*****Example:**

```
sysadmin-vm:0_RP0(config-group-gr1)#users us1
```

Specify the name of the user that should be part of the user group.

You can specify multiple user names enclosed withing double quotes. For example, **users "user1 user2 ..."**.

Step 5 **gid *group_id_value*****Example:**

```
sysadmin-vm:0_RP0(config-group-gr1)#gid 50
```

Specify a numeric value. You can enter any 32 bit integer.

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

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

end —Prompts user to take one of these actions:

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

What to do next

- Create command rules. See [Create Command Rules, on page 36](#).
- Create data rules. See [Create Data Rules, on page 38](#).

Create Users

Create new users for the XR VM and System Admin VM.



Note

Users created in the System Admin VM are different from the ones created in XR VM. As a result, the username and password of a System Admin VM user cannot be used to access the XR VM, and vice versa.

XR VM and System Admin VM User Profile Synchronization

When the user profile is created for the first time in XR VM, the user name and password are synced to the System Admin VM if no user already exists in System Admin VM.

However, the subsequent password change or user deletion in XR VM for the synced user is not synchronized with the System Admin VM.

Therefore, the passwords in XR VM and System Admin VM may not be the same. Also, the user synced with the System Admin VM will not be deleted if the user is deleted in XR VM.

For extensive information about creating user groups, task groups, RADIUS and TACACS configurations, see the *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco ASR 9000 Series Routers*. For detailed information about commands, syntax and their description, see the *Authentication, Authorization, and Accounting Commands* chapter in the *System Security Command Reference for Cisco ASR 9000 Series Routers*.

Create a User Profile in XR VM

Each user is identified by a username that is unique across the administrative domain. Each user must be a member of at least one user group. Deleting a user group may orphan the users associated with that group. The AAA server authenticates orphaned users but most commands are not authorized.

For more information about AAA, and creating users, see the *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco ASR 9000 Series Routers*. For detailed information about related commands, syntax and their description, see the *Authentication, Authorization, and Accounting Commands* chapter in the *System Security Command Reference for Cisco ASR 9000 Series Routers*.

Step 1 **configure**

Example:

```
RP/0/RSP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **username user-name**

Example:

```
RP/0/RSP0/CPU0:router(config)# username user1
```

Creates a name for a new user (or identifies a current user) and enters username configuration submenu.

- The *user-name* argument can be only one word. Spaces and quotation marks are not allowed.

Step 3 Do one of the following:

- **password** {0 | 7} *password*
- **secret** {0 | 5 | 8 | 9 | 10} *secret*

Example:

```
Router(config-un)# password 0 pwd1
```

or

```
Router(config-un)# secret 0 sec1
```

Specifies a password for the user named in Step 2.

- Use the **secret** command to create a secure login password for the user names specified in Step 2.
- Entering **0** following the **password** command specifies that an unencrypted (clear-text) password follows. Entering **7** following the **password** command specifies that an encrypted password follows.
- For the **secret** command, the following values can be entered:

- **0** : specifies that a secure unencrypted (clear-text) password follows
- **5** : specifies that a secure encrypted password follows that uses MD5 hashing algorithm
- **8** : specifies that Type 8 secret that uses SHA256 hashing algorithm follows
- **9** : specifies that Type 9 secret that uses SCrypt hashing algorithm follows

Note The Type 8 and Type 9 secrets are supported on the IOS XR 64-bit operating system starting from Cisco IOS XR Software Release 7.0.1. Prior to this release, it was supported only on the IOS XR 32-bit operating system.

- **10** : specifies Type 10 secret that uses SHA512 hashing algorithm

Note • Type 10 secret is supported only for Cisco IOS XR 64 bit platform.

- Backward compatibility issues such as configuration loss, authentication failure, and so on, are expected when you downgrade to lower versions that still use **MD5** or **SHA256** encryption algorithms. If there are any type 10 secrets, convert the **secrets** to type 5 if you are downgrading the system from versions 7.0.1 and above to versions 6.5.3 and above. If you are downgrading the system from versions 7.0.1 and above to versions below 6.5.3, then un-configure all users from the XR-vm and sysadmin-vm before executing install activate. Backward compatibility issue does not occur in Cisco ASR 9000 Series Routers running Cisco IOS XR 32-Bit software because Type 10 secret is not applicable to such routers.

- In a first user configuration scenario or when you reconfigure a user, the system synchronises only the Type 5 and Type 10 secrets from XR VM to System Admin VM and Host VM. It does not synchronize the Type 8 and Type 9 secrets in such scenarios.

- Type **0** is the default for the **password** and **secret** commands.
- From Cisco IOS XR Software Release 7.0.1 and later, the default hashing type is 10 (SHA512) when clear text secret is configured without choosing the type in the configuration.

Step 4 **group** *group-name*

Example:

```
RP/0/RSP0/CPU0:router(config-un)# group sysadmin
```

Assigns the user named in Step 2 to a user group that has already been defined through the **usergroup** command.

- The user takes on all attributes of the user group, as defined by that user group's association to various task groups.
- Each user must be assigned to at least one user group. A user may belong to multiple user groups.

Step 5 Repeat step 4 for each user group to be associated with the user specified in step 2.

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

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

end —Prompts user to take one of these actions:

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

Create a User Profile in System Admin VM

Create new users for the System Admin VM. Users are included in a user group and assigned certain privileges. The users have restricted access to the commands and configurations in the System Admin VM console, based on assigned privileges.

The router supports a maximum of 1024 user profiles.

The root-lr user of XR VM can access the System Admin VM by entering **Admin** command in the EXEC mode. The router does not prompt you to enter any username and password. The XR VM root-lr user is provided full access to the System Admin VM.

SUMMARY STEPS

1. **admin**
2. **config**
3. **aaa authentication users user** *user_name*
4. **password** *password*
5. **uid** *user_id_value*
6. **gid** *group_id_value*
7. **ssh_keydir** *ssh_keydir*
8. **homedir** *homedir*
9. Use the **commit** or **end** command.

DETAILED STEPS

Step 1 **admin**

Example:

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 2 **config**

Example:

```
sysadmin-vm:0_RP0#config
```

Enters mode.

Step 3 **aaa authentication users user *user_name***

Example:

```
sysadmin-vm:0_RP0(config)#aaa authentication users user us1
```

Creates a new user and enters user configuration mode. In the example, the user "us1" is created.

Step 4 **password *password***

Example:

```
sysadmin-vm:0_RP0(config-user-us1)#password pwd1
```

Enter the password that will be used for user authentication at the time of login into System Admin VM.

Step 5 **uid *user_id_value***

Example:

```
sysadmin-vm:0_RP0(config-user-us1)#uid 100
```

Specify a numeric value. You can enter any 32 bit integer.

Step 6 **gid *group_id_value***

Example:

```
sysadmin-vm:0_RP0(config-user-us1)#gid 50
```

Specify a numeric value. You can enter any 32 bit integer.

Step 7 **ssh_keydir *ssh_keydir***

Example:

```
sysadmin-vm:0_RP0(config-user-us1)#ssh_keydir dir1
```

Specify any alphanumeric value.

Step 8 **homedir *homedir***

Example:

```
sysadmin-vm:0_RP0(config-user-us1)#homedir dir2
```

Specify any alphanumeric value.

Step 9 Use the **commit** or **end** command.

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

end —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.

- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

What to do next

- Create user group that includes the user created in this task. See [Create a User Group in System Admin VM, on page 30](#).
- Create command rules that apply to the user group. See [Create Command Rules, on page 36](#).
- Create data rules that apply to the user group. See [Create Data Rules, on page 38](#).

Create Command Rules

Command rules are rules based on which users of a user group are either permitted or denied the use of certain commands. Command rules are associated to a user group and get applied to all users who are part of the user group.

A command rule is created by specifying whether an operation is permitted, or denied, on a command. This table lists possible operation and permission combinations:

Operation	Accept Permission	Reject Permission
Read (R)	Command is displayed on the CLI when "?" is used.	Command is not displayed on the CLI when "?" is used.
Execute (X)	Command can be executed from the CLI.	Command cannot be executed from the CLI.
Read and execute (RX)	Command is visible on the CLI and can be executed.	Command is neither visible nor executable from the CLI.

By default, all permissions are set to **Reject**.

Each command rule is identified by a number associated with it. When multiple command rules are applied to a user group, the command rule with a lower number takes precedence. For example, cmdrule 5 permits read access, while cmdrule10 rejects read access. When both these command rules are applied to the same user group, the user in this group gets read access because cmdrule 5 takes precedence.

As an example, in this task, the command rule is created to deny read and execute permissions for the "show platform" command.

Before you begin

Create an user group. See [Create a User Group in System Admin VM, on page 30](#).

SUMMARY STEPS

1. **admin**
2. **config**
3. **aaa authorization cmdrules cmdrule *command_rule_number***
4. **command *command_name***

5. **ops** {**r** | **x** | **rx**}
6. **action** {**accept** | **accept_log** | **reject**}
7. **group** *user_group_name*
8. **context** *connection_type*
9. Use the **commit** or **end** command.

DETAILED STEPS

Step 1 **admin**

Example:

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 2 **config**

Example:

```
sysadmin-vm:0_RP0#config
```

Enters `mode`.

Step 3 **aaa authorization cmdrules cmdrule** *command_rule_number*

Example:

```
sysadmin-vm:0_RP0(config)#aaa authorization cmdrules cmdrule 1100
```

Specify a numeric value as the command rule number. You can enter a 32 bit integer.

Important Do not use numbers between 1 to 1000 because they are reserved by Cisco.

This command creates a new command rule (if it is not already present) and enters the command rule configuration mode. In the example, command rule "1100" is created.

Note By default "cmdrule 1" is created by the system when the root-system user is created. This command rule provides "accept" permission to "read" and "execute" operations for all commands. Therefore, the root user has no restrictions imposed on it, unless "cmdrule 1" is modified.

Step 4 **command** *command_name*

Example:

```
sysadmin-vm:0_RP0(config-cmdrule-1100)#command "show platform"
```

Specify the command for which permission is to be controlled.

If you enter an asterisk '*' for **command**, it indicates that the command rule is applicable to all commands.

Step 5 **ops** {**r** | **x** | **rx**}

Example:

```
sysadmin-vm:0_RP0(config-cmdrule-1100)#ops rx
```

Specify the operation for which permission has to be specified:

- **r** — Read

- **x** — Execute
- **rx** — Read and execute

Step 6 **action** { **accept** | **accept_log** | **reject** }

Example:

```
sysadmin-vm:0_RP0(config-cmdrule-1100)#action reject
```

Specify whether users are permitted or denied the use of the operation.

- **accept** — users are permitted to perform the operation
- **accept_log** — users are permitted to perform the operation and every access attempt is logged.
- **reject** — users are restricted from performing the operation.

Step 7 **group** *user_group_name*

Example:

```
sysadmin-vm:0_RP0(config-cmdrule-1100)#group gr1
```

Specify the user group on which the command rule is applied.

Step 8 **context** *connection_type*

Example:

```
sysadmin-vm:0_RP0(config-cmdrule-1100)#context *
```

Specify the type of connection to which this rule applies. The connection type can be *netconf* (Network Configuration Protocol), *cli* (Command Line Interface), or *xml* (Extensible Markup Language). It is recommended that you enter an asterisk '*'; this indicates that the command rule applies to all connection types.

Step 9 Use the **commit** or **end** command.

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

end — Prompts user to take one of these actions:

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

What to do next

Create data rules. See [Create Data Rules, on page 38](#).

Create Data Rules

Data rules are rules based on which users of the user group are either permitted, or denied, accessing and modifying configuration data elements. The data rules are associated to a user group. The data rules get applied to all users who are part of the user group.

Each data rule is identified by a number associated to it. When multiple data rules are applied to a user group, the data rule with a lower number takes precedence.

Before you begin

Create an user group. See [Create a User Group in System Admin VM, on page 30](#).

SUMMARY STEPS

1. **admin**
2. **config**
3. **aaa authorization datarules datarule** *data_rule_number*
4. **keypath** *keypath*
5. **ops** *operation*
6. **action** { **accept** | **accept_log** | **reject** }
7. **group** *user_group_name*
8. **context** *connection type*
9. **namespace** *namespace*
10. Use the **commit** or **end** command.

DETAILED STEPS

Step 1

admin

Example:

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 2

config

Example:

```
sysadmin-vm:0_RP0#config
```

Enters `mode`.

Step 3

aaa authorization datarules datarule *data_rule_number*

Example:

```
sysadmin-vm:0_RP0(config)#aaa authorization datarules datarule 1100
```

Specify a numeric value as the data rule number. You can enter a 32 bit integer.

Important Do not use numbers between 1 to 1000 because they are reserved by Cisco.

This command creates a new data rule (if it is not already present) and enters the data rule configuration mode. In the example, data rule "1100" is created.

Note By default "datarule 1" is created by the system when the root-system user is created. This data rule provides "accept" permission to "read", "write", and "execute" operations for all configuration data. Therefore, the root user has no restrictions imposed on it, unless "datarule 1" is modified.

Step 4

keypath *keypath*

Example:

```
sysadmin-vm:0_RP0(config-datarule-1100)#keypath /aaa/disaster-recovery
```

Specify the keypath of the data element. The keypath is an expression defining the location of the data element. If you enter an asterisk '*' for **keypath**, it indicates that the command rule is applicable to all configuration data.

Step 5

ops *operation*

Example:

```
sysadmin-vm:0_RP0(config-datarule-1100)#ops rw
```

Specify the operation for which permission has to be specified. Various operations are identified by these letters:

- c—Create
- d—Delete
- u—Update
- w— Write (a combination of create, update, and delete)
- r—Read
- x—Execute

Step 6

action { **accept** | **accept_log** | **reject** }

Example:

```
sysadmin-vm:0_RP0(config-datarule-1100)#action reject
```

Specify whether users are permitted or denied the operation.

- **accept** — users are permitted to perform the operation
- **accept_log**— users are permitted to perform the operation and every access attempt is logged
- **reject**— users are restricted from performing the operation

Step 7

group *user_group_name*

Example:

```
sysadmin-vm:0_RP0(config-datarule-1100)#group gr1
```

Specify the user group on which the data rule is applied. Multiple group names can also be specified.

Step 8

context *connection type*

Example:

```
sysadmin-vm:0_RP0(config-datarule-1100)#context *
```

Specify the type of connection to which this rule applies. The connection type can be *netconf* (Network Configuration Protocol), *cli* (Command Line Interface), or *xml* (Extensible Markup Language). It is recommended that you enter an asterisk '*', which indicates that the command applies to all connection types.

Step 9

namespace *namespace*

Example:

```
sysadmin-vm:0_RP0(config-datarule-1100)#namespace *
```

Enter asterisk '*' to indicate that the data rule is applicable for all namespace values.

Step 10 Use the **commit** or **end** command.

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

end —Prompts user to take one of these actions:

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

Change Disaster-recovery Username and Password

When you define the root-system username and password initially after starting the router, the same username and password gets mapped as the disaster-recovery username and password for the System Admin console. However, it can be changed.

The disaster-recovery username and password is useful in these scenarios:

- Access the system when the AAA database, which is the default source for authentication in System Admin console is corrupted.
- Access the system through the management port, when, for some reason, the System Admin console is not working.
- Create new users by accessing the System Admin console using the disaster-recovery username and password, when the regular username and password is forgotten.



Note On the router, you can configure only one disaster-recovery username and password at a time.

SUMMARY STEPS

1. **admin**
2. **config**
3. **aaa disaster-recovery username** *username* **password** *password*
4. Use the **commit** or **end** command.

DETAILED STEPS

Step 1 **admin**

Example:

```
RP/0/RSP0/CPU0:router# admin
```

Enters administration EXEC mode.

Step 2 **config****Example:**

```
sysadmin-vm:0_RP0#config
```

Enters `mode`.

Step 3 **aaa disaster-recovery username *username* password *password*****Example:**

```
sysadmin-vm:0_RP0(config)#aaa disaster-recovery username us1 password pwd1
```

Specify the disaster-recovery username and the password. You have to select an existing user as the disaster-recovery user. In the example, 'us1' is selected as the disaster-recovery user and assigned the password as 'pwd1'. The password can be entered as a plain text or md5 digest string.

When you need to make use of the disaster recovery username, you need to enter it as *username@localhost*.

Step 4 Use the **commit** or **end** command.

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

end —Prompts user to take one of these actions:

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

Recover Password using PXE Boot

If you are unable to login or lost your XR and System administration passwords, use the following steps to create new password. A lost password cannot be recovered, instead a new username and password must be created with a non-graceful PXE boot.

Step 1 Boot the router using PXE.

Note PXE boot is fully intrusive. The router state, configuration and image is reset.

To PXE boot a router, see [Boot the Router Using iPXE, on page 10](#).

Step 2 Reset the password.



CHAPTER 6

Perform System Upgrade and Install Feature Packages

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

The topics covered in this chapter are:

- [Upgrading the System, on page 43](#)
- [Upgrading Features, on page 44](#)
- [Workflow for Install Process, on page 44](#)
- [Install Packages, on page 44](#)
- [Install Prepared Packages, on page 48](#)
- [Uninstall Packages, on page 51](#)

Upgrading the System



Note If an interface on a router does not have a configuration and is brought up by performing no-shut operation, then upon router reload, the interface state changes to **admin-shutdown** automatically.

System upgrade is done by installing a base package—Cisco IOS XR Unicast Routing Core Bundle. The file name for this bundle is . Install this ISO image using **install** commands. For more information about the install process, see [Workflow for Install Process, on page 44](#).



Caution Do not perform any install operations when the router is reloading.
Do not reload the router during an upgrade operation.

**Note**

CSM Server is a web-based, server-side automation and orchestration framework. It gives service providers the ability to simultaneously schedule and deploy SMUs and perform software upgrades across hundreds of routers in a scheduled manner through a simple point and click Web interface. For more information, see [Cisco Software Manager](#).

For more information on upgrading the system and the RPMs, see *Manage Automatic Dependency* chapter.

Upgrading Features

Upgrading features is the process of deploying new features and software patches on the router. Feature upgrade is done by installing package files, termed simply, packages. Software patch installation is done by installing Software Maintenance Upgrade (SMU) files.

Installing a package on the router installs specific features that are part of that package. Cisco IOS XR software is divided into various software packages; this enables you to select the features to run on your router. Each package contains components that perform a specific set of router functions, such as routing, security, and so on.

For example, the components of the routing package are split into individual RPMs, such as BGP and OSPF. BGP is a mandatory RPM which is a part of the base software version and hence cannot be removed. Optional RPMs such as OSPF can be added and removed as required.

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

Package and SMU installation is performed using **install** commands. For more information about the install process, see [Install Packages, on page 44](#).

There are separate packages and SMUs for the XR VM and the System Admin VM. They can be identified by their filenames.

For more information on upgrading the system and the RPMs, see *Cisco IOS XR Flexible Packaging Configuration Guide*.

Workflow for Install Process

The workflow for installation and uninstallation processes is depicted in this flowchart.

For installing a package, see [Install Packages, on page 44](#). For uninstalling a package, see [Uninstall Packages, on page 51](#).

Install Packages

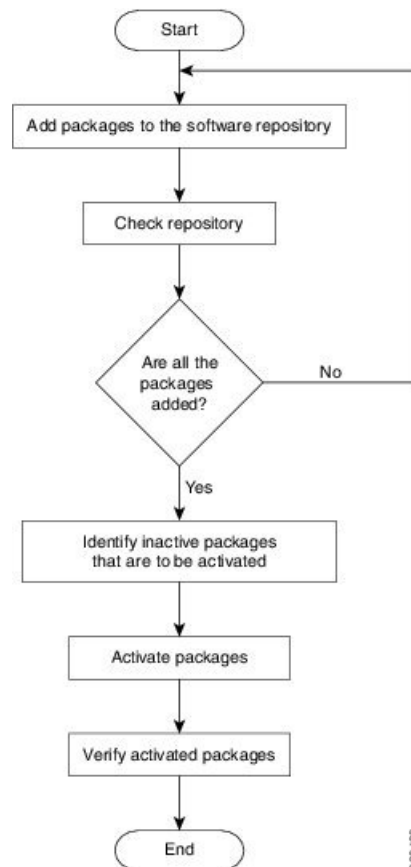
Complete this task to upgrade the system or install a patch. The system upgrade is done using an ISO image file, while the patch installation is done using packages and SMUs. This task is also used to install *.rpm* files. The *.rpm* file contains multiple packages and SMUs that are merged into a single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



Note The System Admin package and XR package can be executed using **install** commands in the `mode` and `mode`. All **install** commands are applicable in both these modes.

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

Figure 3: Installing Packages Workflow



Before you begin

- Configure and connect to the management port. The installable file is accessed through the management port. For details about configuring the management port, see [Configure the Management Port, on page 15](#).
- Copy the package to be installed either on the router's hard disk or on a network server to which the router has access.

SUMMARY STEPS

1. Execute one of these:
 - **install add source** *<tftp transfer protocol>/package_path/ filename1 filename2 ...*

- **install add source** *<ftp or sftp transfer protocol>://user@server:/package_path/ filename1 filename2 ...*
- 2. **show install request**
- 3. **show install repository**
- 4. **show install inactive**
- 5. Execute one of these:
 - **install activate** *package_name*
 - **install activate id** *operation_id*
- 6. **show install active**
- 7. **install commit**

DETAILED STEPS

Step 1 Execute one of these:

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

Example:

or

or

Note A space must be provided between the *package_path* and *filename*.

The software files are unpacked from the package and added to the software repository. This operation might take time depending on the size of the files being added. The operation is performed in asynchronous mode. The **install add** command runs in the background, and the EXEC prompt is returned as soon as possible.

Note The repositories for the XR VM and the System Admin VM are different. The system automatically adds a routing package to the XR VM repository and a system administration package to the System Admin VM repository.

Step 2 **show install request**

Example:

```
RP/0/RSP0/CPU0:router#show install request
```

(Optional) Displays the operation ID of the add operation and its status. The operation ID can be later used to execute the **activate** command.

```
Install operation 8 is still in progress
```

For system administration packages, the remaining steps must be performed from the `admin` mode. Use the **admin** command to enter the `admin` mode.

Step 3 **show install repository**

Example:

```
RP/0/RSP0/CPU0:router#show install repository
```

Displays packages that are added to the repository. Packages are displayed only after the `install add` operation is complete.

Step 4 **show install inactive**

Example:

```
RP/0/RSP0/CPU0:router#show install inactive
```

Displays inactive packages that are present in the repository. Only inactive packages can be activated.

Step 5 Execute one of these:

- **install activate** *package_name*
- **install activate id** *operation_id*

Example:

The *operation_id* is that of the **install add** operation. This command can also be run from System Admin mode. The package configurations are made active on the router. As a result, new features and software fixes take effect. This operation is performed in asynchronous mode. The **install activate** command runs in the background, and the EXEC prompt is returned.

If you use the operation ID, all packages that were added in the specified operation are activated together. For example, if 5 packages are added in operation 8, by executing **install activate id 8**, all 5 packages are activated together. You do not have to activate the packages individually.

Activation does not happen instantaneously, but takes some time. Activation of some SMUs require a manual reloading of the router. When such SMUs are activated, a warning message is displayed to perform reload. The components of the SMU get activated only after the reload is complete. Perform router reload immediately after executing the **install activate** command. If the SMU has dependency on both XR VM and System Admin VM, perform the reload after activating the SMU in both VMs so that they take effect simultaneously. To reload the router, use the **hw-module location all reload** command from the `mode`.

Step 6 **show install active**

Example:

```
RP/0/RSP0/CPU0:router#show install active
```

Displays packages that are active.

From the result, verify that the same image and package versions are active on all RPs and LCs.

Step 7 **install commit**

Example:

```
RP/0/RSP0/CPU0:router#install commit
```

Commits the XR newly active software. To commit both XR and System Admin software, use **install commit system**.

Installing Packages: Related Commands

Related Commands	Purpose
show install log	Displays the log information for the install process; this can be used for troubleshooting in case of install failure.

Related Commands	Purpose
show install package	Displays the details of the packages that have been added to the repository. Use this command to identify individual components of a package.
install prepare	Makes pre-activation checks on an inactive package, to prepare it for activation.
show install prepare	Displays the list of package that have been prepared and are ready for activation.

What to do next

- After performing a system upgrade, upgrade FPD by using the **upgrade hw-module location all fpd all** command from the `mode`. The progress of FPD upgrade process can be monitored using the **show hw-module fpd** command in the `mode`. Reload the router after the FPD upgrade is completed.
- Verify the installation using the **install verify packages** command.
- Uninstall the packages or SMUs if their installation causes any issues on the router. See [Uninstall Packages, on page 51](#).



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

Install Prepared Packages

A system upgrade or feature upgrade is performed by activating the ISO image file, packages, and SMUs. It is possible to prepare these installable files before activation. During the prepare phase, pre-activation checks are made and the components of the installable files are loaded on to the router setup. The prepare process runs in the background and the router is fully usable during this time. When the prepare phase is over, all the prepared files can be activated instantaneously. The advantages of preparing before activation are:

- If the installable file is corrupted, the prepare process fails. This provides an early warning of the problem. If the corrupted file was activated directly, it might cause router malfunction.
- Directly activating an ISO image for system upgrade takes considerable time during which the router is not usable. However, if the image is prepared before activation, not only does the prepare process run asynchronously, but when the prepared image is subsequently activated, the activation process too takes very less time. As a result, the router downtime is considerably reduced.

Complete this task to upgrade the system and install packages by making use of the prepare operation.



Note Depending on whether you are installing a System Admin package or a XR package, execute the **install** commands in the `mode` or `mode` respectively. All **install** commands are applicable in both these modes. System Admin install operations can be done from XR mode.

Before you begin

- If the installable file is corrupted, the prepare process fails. This provides an early warning of the problem. If the corrupted file was activated directly, it might cause router malfunction.
- Directly activating an ISO image for system upgrade takes considerable time during which the router is not usable. However, if the image is prepared before activation, not only does the prepare process run asynchronously, but when the prepared image is subsequently activated, the activation process too takes very less time. As a result, the router downtime is considerably reduced.

SUMMARY STEPS

1. Add the required ISO image and packages to the repository.
2. **show install repository**
3. Execute one of these:
 - **install prepare** *package_name*
 - **install prepare id** *operation_id*
4. **show install prepare**
5. **install activate**
6. **show install active**
7. **install commit**

DETAILED STEPS

Step 1 Add the required ISO image and packages to the repository.
For details, see [Install Packages, on page 44](#).

Step 2 **show install repository**

Example:

```
RP/0/RSP0/CPU0:router#show install repository
```

Perform this step to verify that the required installable files are available in the repository. Packages are displayed only after the "install add" operation is complete.

Step 3 Execute one of these:

- **install prepare** *package_name*
- **install prepare id** *operation_id*

Example:

The prepare process takes place. This operation is performed in asynchronous mode. The **install prepare** command runs in the background, and the EXEC prompt is returned as soon as possible.

If you use the operation ID, all packages that were added in the specified operation are prepared together. For example, if 5 packages are added in operation 8, by executing **install prepare id 8**, all 5 packages are prepared together. You do not have to prepare the packages individually.

Step 4 **show install prepare**

Example:

```
RP/0/RSP0/CPU0:router#show install prepare
```

Displays packages that are prepared. From the result, verify that all the required packages have been prepared.

Step 5 **install activate**

Example:

```
RP/0/RSP0/CPU0:router#install activate
```

All the packages that have been prepared are activated together to make the package configurations active on the router.

Note You should not specify any package name or operation ID in the CLI.

Activation of some SMUs require manual reload of the router. When such SMUs are activated, a warning message is displayed to perform reload. The components of the SMU get activated only after the reload is complete. Perform router reload immediately after the execution of the **install activate** command is completed.

Step 6 **show install active**

Example:

```
RP/0/RSP0/CPU0:router#show install active
```

Displays packages that are active.

From the result, verify that on all RPs and LCs, the same image and package versions are active.

Step 7 **install commit**

Example:

```
RP/0/RSP0/CPU0:router#install commit
```

Installing Packages: Related Commands

Related Commands	Purpose
show install log	Displays the log information for the install process; this can be used for troubleshooting in case of install failure.
show install package	Displays the details of the packages that have been added to the repository. Use this command to identify individual components of a package.
install prepare clean	Clears the prepare operation and removes all the packages from the prepared state.

What to do next

- After performing a system upgrade, upgrade FPD by using the **upgrade hw-module location all fpd all** command from the `mode`. The progress of FPD upgrade process can be monitored using the **show hw-module fpd** command in the `mode`. Reload the router after the FPD upgrade is completed.
- Verify the installation using the **install verify packages** command.
- Uninstall the packages or SMUs if their installation causes any issues on the router. See [Uninstall Packages](#).



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

Uninstall Packages

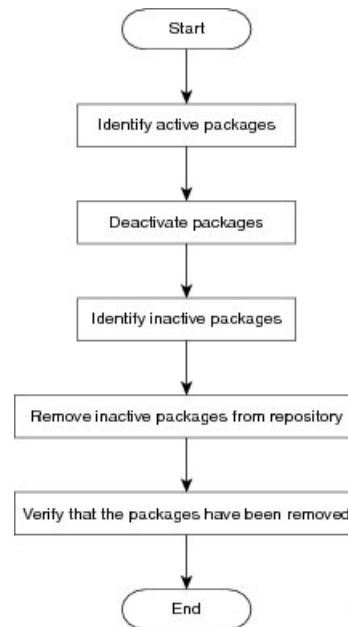
Complete this task to uninstall a package. All router functionalities that are part of the uninstalled package are deactivated. Packages that are added in the XR VM cannot be uninstalled from the System Admin VM, and vice versa.



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

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

Figure 4: Uninstalling Packages Workflow



This task uninstalls XR VM packages. If you need to uninstall System Admin packages, run the same commands from the `mode`.

SUMMARY STEPS

1. **show install active**
2. Execute one of these:
 - **install deactivate** *package_name*

- **install deactivate id** *operation_id*
3. **show install inactive**
 4. **install remove** *package_name*
 5. **show install repository**

DETAILED STEPS

Step 1 **show install active**

Example:

```
RP/0/RSP0/CPU0:router#show install active
```

Displays active packages. Only active packages can be deactivated.

Step 2 Execute one of these:

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

Example:

The *operation_id* is the ID from **install add** operation. All features and software patches associated with the package are deactivated. You can specify multiple package names and deactivate them simultaneously.

If you use the operation ID, all packages that were added in the specified operation are deactivated together. You do not have to deactivate the packages individually. If System admin packages were added as a part of the **install add** operation (of the ID used in deactivate) then those packages will also be deactivated.

Step 3 **show install inactive**

Example:

```
RP/0/RSP0/CPU0:router#show install inactive
```

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

Step 4 **install remove** *package_name*

Example:

The inactive packages are removed from the repository.

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

Step 5 **show install repository**

Example:

```
RP/0/RSP0/CPU0:router#show install repository
```

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

What to do next

Install required packages. .

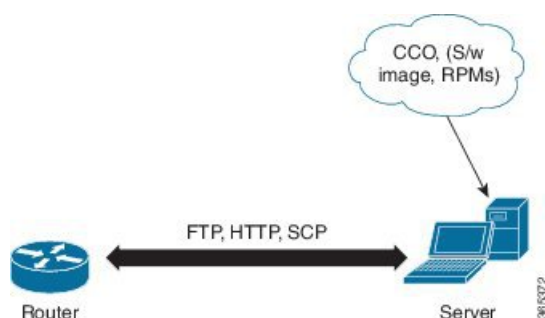


CHAPTER 7

Manage Automatic Dependency

Flexible packaging supports automatic dependency management. While you update an RPM, the system automatically identifies all relevant dependent packages and updates them.

Figure 5: Flow for Installation (base software, RPMs and SMUs)



Until this release, you downloaded the software image and required RPMs from CCO on a network server (the repository), and used the **install add** and the **install activate** commands to add and activate the downloaded files on the router. Then, you manually identify relevant dependent RPMs, to add and activate them.

With automatic dependency management, you need not identify dependent RPMs to individually add and activate them. You can execute new install command to identify and install dependent RPMs automatically.

The command **install source** adds and activates packages. The command **install replace** adds and activates packages in a given golden ISO (GISO).



Note

1. Cisco IOS XR Version 6.1.1 does not provide third party SMUs as part of automatic dependency management (**install source** command). The third party SMUs must be installed separately, and in isolation from other installation procedures (installation of SMUs and RPMs in IOS XR or admin containers).

The rest of this chapter contains these sections:

- [Update RPMs and SMUs, on page 54](#)
- [Upgrade Base Software Version, on page 54](#)
- [Downgrade an RPM, on page 55](#)

Update RPMs and SMUs

An RPM may contain a fix for a specific defect, and you may need to update the system with that fix. To update RPMs and SMUs to a newer version, use the **install source** command. When this command is issued for a particular RPM, the router communicates with the repository, and downloads and activates that RPM. If the repository contains a dependent RPM, the router identifies that dependent RPM and installs that too.

The syntax of the **install source** command is:

```
install source repository [rpm]
```

Four scenarios in which you can use the **install source** command are:

- **When a package name is not specified**

When no package is specified, the command updates the latest SMUs of all installed packages.

```
install source [repository]
```

- **When a package name is specified**

If the package name is specified, the command installs that package, updates the latest SMUs of that package, along with its dependencies. If the package is already installed, only the SMUs of that package are installed. (SMUs that are already installed are skipped.)

```
install source[repository] asr9k-mpls.rpm
```

- **When a package name and version number are specified**

If a particular version of package needs to be installed, the complete package name must be specified; that package is installed along with the latest SMUs of that package present in the repository.

```
install source[repository] asr9k-mpls-1.0.2.0-r611.x86_64.rpm
```

- **When an SMU is specified**

If an SMU is specified, that SMU is downloaded and installed, along with its dependent SMUs.

```
install source[repository] asr9k-mpls-1.0.2.1-r611.CSCub12345.x86_64.rpm
```

Upgrade Base Software Version

You can upgrade to a newer version of the base software when it becomes available. To upgrade to the latest base software version, use the **install source** command. With the upgrade of the base version, RPMs that are currently available on the router are also upgraded.



Note SMUs are not upgraded as part of this process.

The syntax of the **install source** command is:

```
install source repository version version-number [rpm]
```



Note VRF and TPA on dataport is not supported. If the server is reachable only through non-default VRF interface, the file must already be retrieved using ftp, sftp, scp, http or https protocols.



Note Default routes (0.0.0.0/0) cannot be copied onto Linux due to TPA implementation.

You can use the **install source** command when:

- **The version number is specified**

The base software (.mini) is upgraded to the specified version; all installed RPMs are upgraded to the same release version.

```
install source [repository] version <version> asr9k-mini-x64-<version>.iso
```

For example,

```
install source repository version 7.0.1 asr9k-mini-x64-7.0.1.iso
```

You can also automatically fetch the .mini file and RPMs of the required release and proceed with the upgrade.

```
install source repository asr9k-mini-x64-7.0.1.iso
```

- **The version number for an RPM is specified**

When performing a system upgrade, the user can choose to have an optional RPM to be of a different release (from that of the base software version); that RPM can be specified.

```
install source repository version 7.0.1
asr9k-mp1s-1.0.2.0-r701.x86_64.rpm
```

Downgrade an RPM

An RPM can be downgraded after it is activated. RPMs are of the following types:

- **Hostos RPM:** The RPM contains `hostos` in the name.

For example:

- <platform>-sysadmin-hostos-6.5.1-r651.CSChu77777.host.arm
- <platform>-sysadmin-hostos-6.5.1-r651.CSChu77777.admin.arm
- <platform>-sysadmin-hostos-6.5.1-r651.CSChu77777.host.x86_64
- <platform>-sysadmin-hostos-6.5.1-r651.CSChu77777.admin.x86_64

- **Non-hostos RPM:** The RPM does not contain `hostos` in the name.

For example:

- <platform>-sysadmin-system-6.5.1-r651.CSCvc12346

To deactivate the RPMs, perform the following steps:

- **Downgrade Hostos RPM**

- Scenario 1: To downgrade to version 06 from the active version 09:

1. Download the version 06 hostos RPMs, and add the RPMs.

```
install add source [repository]
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.host.arm
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.admin.arm
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.host.x86_64
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.admin.x86_64
```

2. Activate the downloaded RPMs.

```
install activate [repository]
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.host.arm
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.admin.arm
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.host.x86_64
<platform>-sysadmin-hostos-6.5.1.06-r65108I.CSChu44444.admin.x86_64
```

3. Commit the configuration.

```
install commit
```

- Scenario 2: Deactivate hostos RPM by activating base RPM, consider version 09 is active:

1. Activate the base RPM.

```
install activate <platform>-sysadmin-hostos-6.5.1.08I-r65108I.admin.arm
<platform>-sysadmin-hostos-6.5.1.08I-r65108I.host.arm
<platform>-sysadmin-hostos-6.5.1.08I-r65108I.admin.x86_64
<platform>-sysadmin-hostos-6.5.1.08I-r65108I.host.x86_64
```

For example, if RPM is the RPM installed, then is its base RPM.

2. Commit the configuration.

```
install commit
```

The downgrade for third-party RPMs is similar to the hostos RPMs. To downgrade a SMU, activate the lower version of the SMU. If only one version of SMU is present, the base RPM of the SMU must be activated.



Note Hostos and third-party RPMs cannot be deactivated. Only activation of different versions is supported.

- **Downgrade Non-Hostos RPM**

1. Deactivate the RPM to downgrade to earlier version of RPM.

```
install deactivate <platform>-<rpm-name>
```

2. Check the active version of the RPM.

```
show install active
```

3. Commit the configuration.


```
install commit
```




CHAPTER 8

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 github location [Github](#) location.

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 64](#).

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, on page 59](#)
- [Golden ISO Workflow, on page 60](#)
- [Build Golden ISO, on page 61](#)
- [Install Golden ISO, on page 64](#)
- [Install Replace with Golden ISO, on page 67](#)

Limitations

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

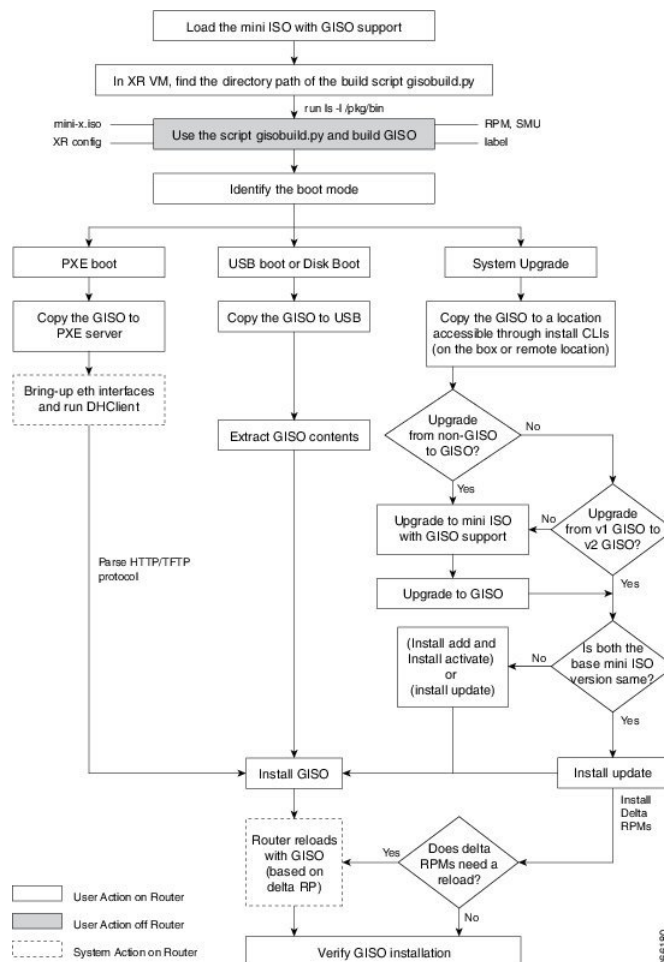
- GISO image size more than 1.8 GB is not supported. The maximum image size for RSP880-LT-SE/TR is 1.599 GB.

- Building and booting GISO for asynchronous package (a package of different release than the ISO) is not supported.
- Verifying the XR configuration is not supported in the GISO build script `gisobuild.py`.
- Renaming a GISO build and then installing from the renamed GISO build is not supported.
- Migrating from IOS XR 32-bit to 64-bit OS using GISO involves the following restrictions:
 - The IOS XR 32-bit to 64-bit conversion script does not support file names exceeding 48 characters.
 - The IOS XR 32-bit OS has a maximum file size limit of 2 GB. Ensure that GISO does not exceed that limit.

For more information about migration methods and system requirements, see the [Migration Guide for Cisco ASR 9000 Series Routers](#).

Golden ISO Workflow

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



Build Golden ISO

The customized ISO is built using Cisco Golden ISO (GISO) build script `gisobuild.py` available on the [Github](#) location.

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 `asr9000-mpls-te-rsvp` is dependent on the parent RPM `asr9000-mpls`. If only the child RPM is included, the Golden ISO build fails.

Build Golden ISO Using Script

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 (mandatory)
- one or more third-party SMUs for host, XR and System admin (mandatory)
- Label for golden ISO (optional)
- Optional RPMs

Use the following naming convention when building GISO:

GISO Build	Format	Example
GISO without k9sec RPM	<code><platform>-golden-x.iso-<version>.<label></code>	<code><platform>-golden-x64.iso-<version>.v1</code>
	<code><platform>-golden-x-<version>.iso.<label></code>	<code><platform>-golden-x64-<version>.iso.v1</code>
GISO with k9sec RPM	<code><platform>-goldenk9-x.iso-<version>.<label></code>	<code><platform>-goldenk9-x64.iso-<version>.v1</code>
	<code><platform>-goldenk9-x-<version>.iso.<label></code>	<code><platform>-goldenk9-x64-<version>.iso.v1</code>



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 ASR 9000 series routers, upgrade to release 6.1.3 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.py` script is executed must have root credentials.

-
- Step 1** Copy the script `gisobuild.py` from the [Github](#) location 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.
- Step 2** Run the script `gisobuild.py` and provide parameters to build the golden ISO off the router. Ensure that all RPMs and SMUs are present in the same directory or on a repository. The number of RPMs and SMUs that can be used to build the Golden ISO is 128.

Example:

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

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

The corresponding GISO and build logs are available under the specified `out_directory` path. The default directory is `/output_gisobuild`.

```
[directory-path]$ gisobuild.py [-h] [-i asr9k-mini-x.iso] [-r <rpm repository>]
[-c <config-file>] [-l <giso label>] [-m] [-v]
Golden ISO build process starting...
```

```
System requirements check [PASS]
```

```
Platform: asr9k Version: <version>
```

```
Info: Migration option is provided so migration tar will be generated
```

```
Scanning repository [repo]...
```

```
Building RPM Database...
```

```
Total 56 RPM(s) present in the repository path provided in CLI
```

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

```

(+) asr9k-diags-x64-1.0.0.0-r<version>.x86_64.rpm
(+) asr9k-parser-x64-2.0.0.0-r<version>.x86_64.rpm
(+) asr9k-mgbl-x64-3.0.0.0-r<version>.x86_64.rpm
(+) asr9k-k9sec-x64-2.2.0.0-r<version>.x86_64.rpm
(+) asr9k-os-supp-64-4.0.0.1-r<version>.CSCu77777.x86_64.rpm
(+) asr9k-mpis-x64-2.1.0.0-r<version>.x86_64.rpm
(+) asr9k-k9sec-x64-2.2.0.1-r<version>.CSCxr33333.x86_64.rpm

-----Truncated

Skipping following rpms from repository since they are already present in base ISO:

(-) asr9k-parser-x64-2.0.0.0-r<version>.x86_64.rpm
(-) asr9k-bgp-x64-1.0.0.0-r<version>.x86_64.rpm
(-) asr9k-diags-x64-1.0.0.0-r<version>.x86_64.rpm

...RPM compatibility check [PASS]

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

(+) asr9k-sysadmin-system-<version>-r<version>.CSCcv11111.x86_64.rpm
(+) asr9k-sysadmin-shared-<version>-r<version>.CSCcv22222.x86_64.rpm
(+) asr9k-sysadmin-system-<version>-r<version>.CSCcv44444.x86_64.rpm

-----Truncated

...RPM compatibility check [PASS]

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

(+) asr9k-sysadmin-hostos-<version>-r<version>.CSCu77777.host.x86_64.rpm
(+) cisco-klm-mifpga-0.1.p1-r0.0.CSCtp12345.host.x86_64.rpm
(+) kernel-modules-3.14-r0.1.host.x86_64.rpm

...RPM compatibility check [PASS]

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

XR rpms:
asr9k-k9sec-x64-2.2.0.0-r<version>.x86_64.rpm
iosxr-infra-asr9k-64-4.0.0.2-r<version>.CSCxr11111.x86_64.rpm
asr9k-mpis-te-rsvp-x64-1.2.0.0-r<version>.x86_64.rpm

HOST rpms:
asr9k-sysadmin-hostos-<version>-r<version>.CSCu77777.host.x86_64.rpm
cisco-klm-mifpga-0.1.p1-r0.0.CSCtp12345.host.x86_64.rpm
kernel-modules-3.14-r0.1.host.x86_64.rpm

...Golden ISO creation SUCCESS.

Golden ISO Image Location: <repo>/asr9k-goldenk9-x64.iso-<version>.v1

Building Migration tar...

Migration tar creation SUCCESS.

Migration tar Location: <repo>/asr9k-goldenk9-x64-migrate_to_eXR.tar-<version>.v1

Detail logs: <repo>/Giso_build.log-2016-12-18:03:53:54.837680
[03:54:53]-[router:<repo>]$ ls -ltr asr9k-goldenk9-x64*
-rw-r--r-- 1 root root 1274669056 Dec 18 03:54 asr9k-goldenk9-x64.iso-<version>.v1
-rw-r--r-- 1 root root 1425008640 Dec 18 03:54 asr9k-goldenk9-x64-migrate_to_x64.tar-<version>.v1

```

-----Truncated

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

Note It is recommended to build GISOs with a label name.

The corresponding GISO and build logs are available under the specified directory in `out_directory`. If a directory is not specified, the files are placed in `/output_gisobuild` directory.



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

What to do next

Install the GISO image 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 DHCPClient is run on each interface. DHCPClient script parses HTTP or TFTP protocol, and GISO is downloaded to the box.

When you bring up a router using the PXE boot mode, existing configurations are removed. To recover smart licensing configurations like Permanent License Reservation (PLR), enable these configurations after the router comes up.

```
Router# configure
Router(config)# license smart reservation
Router(config)# commit
```

The following example shows the logs from installation of GISO using PXE boot:

```
...
Fri Dec 2 19:18:03 UTC 2016: ---Starting to prepare host logical volume---
...
Fri Dec 02 19:18:14 UTC 2016: Skipping tp base rpm(openssh-scp-6.6p1-r0.0.host.x86_64.rpm) from
installation
Fri Dec 02 19:18:14 UTC 2016: Skipping tp base rpm(kernel-modules-3.14-r0.1.host.x86_64.rpm) from
installation
Fri Dec 02 19:18:15 UTC 2016: Installing asr9k-sysadmin-hostos-6.1.3-r613.CSChu77777.host.x86_64
[SUCCESS]
...

Fri Dec 2 19:18:23 UTC 2016: ---Starting to prepare calvados logical volume---
...

Fri Dec 02 19:18:48 UTC 2016: Skipping tp base rpm(openssh-scp-6.6p1-r0.0.admin.x86_64.rpm) from
installation
Fri Dec 02 19:18:48 UTC 2016: Skipping tp base rpm(kernel-modules-3.14-r0.1.admin.x86_64.rpm)
from installation
Fri Dec 02 19:18:49 UTC 2016: Installing asr9k-sysadmin-system-6.1.3-r613.CSCcv44444.x86_64
[SUCCESS]
Fri Dec 02 19:18:50 UTC 2016: Installing asr9k-sysadmin-shared-6.1.3-r613.CSCcv33333.x86_64
[SUCCESS]
Fri Dec 02 19:18:51 UTC 2016: Installing asr9k-sysadmin-hostos-6.1.3-r613.CSChu77777.admin.x86_64
[SUCCESS]
...

Fri Dec 2 19:19:07 UTC 2016: ---Starting to prepare repository---
Fri Dec 2 19:19:11 UTC 2016: File system creation on /test took 3 seconds
Fri Dec 2 19:19:11 UTC 2016: Copying /iso/host.iso to repository /iso directory
Fri Dec 2 19:19:11 UTC 2016: Copy Host rpms to repository
Fri Dec 2 19:19:13 UTC 2016: Copying /iso/asr9k-sysadmin.iso to repository /iso directory
Fri Dec 2 19:19:13 UTC 2016: Copy Sysadmin rpms to repository
Fri Dec 2 19:19:16 UTC 2016: Copy HostOs rpms to repository
Fri Dec 2 19:19:16 UTC 2016: Copy XR rpms to repository
Fri Dec 2 19:19:16 UTC 2016: Copy giso_info.txt to repository
Fri Dec 2 19:19:17 UTC 2016: Copying /iso/asr9k-xr.iso to repository /iso directory
Fri Dec 2 19:19:21 UTC 2016: Copying all ISOs to repository took 10 seconds
...
```

- **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 replace** commands.

Important To replace the current version and packages on the router with the version from GISO, note the change in command and format.

- In versions prior to Cisco IOS XR Release 6.3.3, 6.4.x and 6.5.1, use the **install update** command:

```
install update source <source path> <Golden-ISO-name> replace
```

- In Cisco IOS XR Release 6.5.2 and later, use the **install replace** command.

```
install replace <absoulte-path-of-Golden-ISO>
```

The options to upgrade the system are as follows:

- **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.
- **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 replace** command to perform the system upgrade. The router reloads after the upgrade with the version2 GISO image.

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
-----
asr9k-sysadmin-6.1.1

asr9k-sysadmin-hostos-6.1.1-r611.CSCcv10001.admin.x86_64
asr9k-sysadmin-system-6.1.1-r611.CSCcv10005.x86_64
....

XR repository
-----
asr9k-iosxr-mgbl-3.0.0.0-r611.x86_64
asr9k-xr-6.1.1
....

Host repository
-----
host-6.1.1
```

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

Note To list RPMs in the GISO, the GISO must be present in the install repository.

```
Router#show install package asr9k-goldenk9-x64-6.1.3
```

```
Sun Dec  4 13:52:48.279 UTC
This may take a while ...
ISO Name: asr9k-goldenk9-x64-6.1.3
ISO Type: bundle
ISO Bundled: asr9k-mini-x64-6.1.3
```

```

Golden ISO Label: temp
ISO Contents:
  ISO Name: asr9k-xr-6.1.3
  ISO Type: xr
  rpms in xr ISO:
    iosxr-os-asr9k-64-5.0.0.0-r613
    iosxr-ce-asr9k-64-3.0.0.0-r613
    iosxr-infra-asr9k-64-4.0.0.0-r613
    iosxr-fwding-asr9k-64-4.0.0.0-r613
    iosxr-routing-asr9k-64-3.1.0.0-r613
    ...

  ISO Name: asr9k-sysadmin-6.1.3
  ISO Type: sysadmin
  rpms in sysadmin ISO:
    asr9k-sysadmin-topo-6.1.3-r613
    asr9k-sysadmin-shared-6.1.3-r613
    asr9k-sysadmin-system-6.1.3-r613
    asr9k-sysadmin-hostos-6.1.3-r613.admin
    ...

  ISO Name: host-6.1.3
  ISO Type: host
  rpms in host ISO:
    asr9k-sysadmin-hostos-6.1.3-r613.host

Golden ISO Rpms:
  xr rpms in golden ISO:
    asr9k-k9sec-x64-2.2.0.1-r613.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
    asr9k-mpls-x64-2.1.0.0-r613.x86_64.rpm
    asr9k-k9sec-x64-2.2.0.0-r613.x86_64.rpm

  sysadmin rpms in golden ISO:
    asr9k-sysadmin-system-6.1.3-r613.CSCcv11111.x86_64.rpm
    openssh-scp-6.6p1-r0.0.admin.x86_64.rpm
    openssh-scp-6.6p1.p1-r0.0.CSCtp12345.admin.x86_64.rpm

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

```

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

Install Replace with Golden ISO

Golden ISO (GISO) upgrades the router to a version that has a predefined list of software maintenance update (SMUs) with a single operation. However, to update to the same version with a different set of SMUs requires a two-step process.

To avoid this two-step process, use the **install replace** command to replace the currently active version with the full package including the image and SMUs from the newly added GISO.

The process involves upgrading the GISO to add the delta SMUs, and manually deactivating the SMUs that are not in use. In addition, this is the only method to upgrade to GISO containing different optional RPMs, which is a subset of the running set of optional RPMs. For example, consider V1 of GISO is the running

version with V1 mini and optional RPMs V1 mpls, V1 mpls-te, V1 mgbl, and V1 k9sec. If V2 of GISO does not contain V2 k9sec, then use **install replace** to upgrade to the optional RPMs in V2.

**Important**

To replace the current version and packages on the router with the version from GISO, note the change in command and format.

- In versions prior to Cisco IOS XR Release 6.3.3, 6.4.x and 6.5.1, use the **install update** command:

```
install update source <source path> <Golden-ISO-name> replace
```

- In Cisco IOS XR Release 6.5.2 and later, use the **install replace** command.

```
install replace <absoulte-path-of-Golden-ISO>
```

**Note**

The **install replace** command is supported only with GISO, but not with .mini and .rpm packages directly.

**Note**

A service pack (SP) is a tar file consisting of bugfix SMUs used for only SMU upgrades. Whereas, a GISO image is used for upgrading the iso images, feature packages, and SMUs on the router. It is recommended to install GISO image because GISO provides customization options during the build process and an enhanced install upgrade experience. GISO provides benefits such as label-based install, automatic management of dependencies, filtering out fully superseded SMUs, handling optional RPMs during GISO build and decreased install upgrade schedule.

A standard image consisting of the latest SMU set, optional RPMs, and base iso packages are combined to build a GISO with a frequency matching the regular service pack release schedule for 32-bit platforms. For 64-bit platforms, download the standard GISO from [Software Download Center](#) and perform the following steps. Alternatively, to build a GISO consisting of SMU RPMs using customized GISO.

Step 1 **install replace** <GISO-location> [commit|noprompt]

Example:

```
Router#install replace harddisk:/<giso-image>.iso
*****
Install operation 11 started by root:
exec-timeout is suspended.
No install operation in progress at this moment
Label = More_Pkgs
ISO <giso-iso-image>.iso in input package list. Going to upgrade the system to

version <new-giso-image>.
System is in committed state
Current full-label: <giso-image>_R_Commit
Current only-label: R_Commit
Current label: R_Commit
Updating contents of golden ISO
Scheme : localdisk
Hostname : localhost
Username : None
SourceDir : /ws
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
Fetching .... <giso-image>.iso
Label within GISO: More_Pkgs
Skipping <platform>-mgbl-3.0.0.0-<release>.x86_64.rpm from GISO as it's active
Adding packages
      <platform>-golden-x-<release>-<Label>.iso
RP/0/RP0/CPU0:Jun 20 14:43:59.349 UTC: sdr_instmgr[1164]: %INSTALL-INSTMGR-2-OPERATION_SUCCESS :

Install operation 12 finished successfully
Install add operation successful
Activating <platform>-golden-x-<release>-<Label>
Jun 20 14:44:05 Install operation 13 started by root:
      install activate pkg <platform>-golden-x-<release>-<Label> replace noprompt
Jun 20 14:44:05 Package list:
Jun 20 14:44:05      <platform>-golden-x-<release>-<Label>.iso
Jun 20 14:44:29 Install operation will continue in the background
exec-timeout is resumed.
Router# Install operation 13 finished successfully
Router: sdr_instmgr[1164]: %INSTALL-INSTMGR-2-OPERATION_SUCCESS :

Install operation 13 finished successfully

Router#install replace <path-to-image> <platform-name-golden-x-<version>-<label>.iso
Tue Mar 17 08:07:15.176 UTC
+++++
Mar 17 08:07:24 Install operation 46 started by root:
Mar 17 08:07:24   install replace source <path-to-image> <platform-name-golden-x-<version>-<label>.iso
Mar 17 08:07:24 No install operation in progress at this moment
Mar 17 08:07:24 Checking system is ready for install operation
Mar 17 08:07:24 'install replace' in progress
Mar 17 08:07:24 Label = GISO_IMAGE_XRV9K_<version>
Mar 17 08:07:24 ISO xrv9k-goldenk9-x-<version>-<label>.iso in input package list. Going to upgrade
the system to version <new-version>
Mar 17 08:07:25 Scheme : http
Mar 17 08:07:25 Hostname : 10.x.x.x
Mar 17 08:07:25 Collecting software state..
Mar 17 08:07:25 Getting platform
Mar 17 08:07:25 Getting supported architecture
Mar 17 08:07:25 Getting active packages from XR
Mar 17 08:07:25 Getting inactive packages from XR
Mar 17 08:07:28 Getting list of RPMs in local repo
Mar 17 08:07:28 Getting list of provides of all active packages
Mar 17 08:07:28 Getting provides of each rpm in repo
Mar 17 08:07:28 Getting requires of each rpm in repo
Mar 17 08:07:36 Fetching .... xrv9k-goldenk9-x-<version>-<label>.iso
Mar 17 08:08:02 Adding packages
      xrv9k-goldenk9-x-<version>-<label>.iso
Router:Mar 17 08:09:03.487 UTC: sdr_instmgr[1281]: %INSTALL-INSTMGR-2-OPERATION_SUCCESS : Install
operation 47 finished successfully
Mar 17 08:09:03 Install add operation successful
Mar 17 08:09:08 Activating xrv9k-goldenk9-x-<version>-<label>
Mar 17 08:09:10 Install operation 46 started by root:
      install activate pkg xrv9k-goldenk9-x-<version>-<label> replace
Mar 17 08:09:10 Package list:
Mar 17 08:09:10      xrv9k-goldenk9-x-<version>-<label>
This install operation will reload the system, continue?
[yes/no]:[yes] yes

```

```
Mar 17 08:10:30 Install operation will continue in the background
Mar 17 08:10:30 Activate operation ID is: 46 for 'install source' ID:46
```

```
Router# Install operation 46 finished successfully
%INSTALL-INSTMGR-2-OPERATION_SUCCESS : Install operation 46 finished successfully
sdr_instmgr[1150]: %INSTALL-INSTMGR-2-SYSTEM_RELOAD_INFO : The whole system will be reloaded to
complete install operation 46
```

For Cisco IOS XR Release 7.1.1, use the command `install replace harddisk:/<dir>/<giso-image>.iso`.

Important For versions earlier than Cisco IOS XR Release 6.5.2, use the following command:

For example,

```
Router#install update source harddisk:/ <giso-image>.iso replace
```

The version and label of the newly added GISO is compared with the version and label of the currently active version. If a mismatch is identified, a new partition is created and the full package is installed. After installation, the system reloads with the image and packages from the newly added GISO.

Using the `commit` keyword in the command automatically starts the **install commit** operation after the router reloads during activation of the image.

Note Activating or deactivating on a system that has a valid label invalidates the label. This action is irreversible. For example, running **show version** command on the system displays the label `6.3.3_633rev1005`. If any SMU is activated or deactivated on the system, the label `633rev1005` is invalidated, and the `show version` command displays only `6.3.3` as the label.

Step 2 **show version**

Example:

```
Router#show version
Wed Jun 20 15:06:37.915 UTC
Cisco IOS XR Software, Version <new-giso-image>
Copyright (c) 2013-2018 by Cisco Systems, Inc.
```

```
Build Information:
Built By      : <user>
Built On     : <date>
Build Host   : <host-name>
Workspace    : <workspace-name>
Version      : <version>
Location     : <path>
Label       : <label-name>
```

```
cisco <platform> () processor
System uptime is 3 hours 51 minutes
```

The system loads with the image and packages from the newly added GISO.