Upgrade Guide for Cisco Catalyst IE9300 Rugged Series Switches

Cisco Catalyst IE9300 Rugged Series
Updated July 27, 2026

Full Cisco Trademarks with Software License

Full Cisco Trademarks with Software License

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 NONINFRINGEMENT 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)

Communications, services, and additional information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions, and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a gateway to the Cisco bug-tracking system, which maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. The BST provides you with detailed defect information about your products and software.

Documentation feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.

Bias Free Language

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

Topics included

Full Cisco Trademarks with Software License....ii
Communications, services, and additional information....iii
Cisco Bug Search Tool....iii
Documentation feedback....iii
Bias Free Language....iv

1 Upgrade Guide for IE9300 Rugged Series Switches....7
Introduction....8
Requirements....8
Best practice for hardware and software selection....8
Background information....8
Software installation mode overview....8
Bundle mode....9
Install mode....10
Upgrade methods....12
Configure autoupgrade on stacked switches....12
Install mode commands for Cisco devices....13
Configure bundle mode software upgrade....17
Resolve a version mismatch in stacked switches....18
Recover a device from ROMmon mode....19

1 Upgrade Guide for IE9300 Rugged Series Switches

Topics:

- Introduction
- Background information
- Software installation mode overview
- Upgrade methods
- Resolve a version mismatch in stacked switches
- Recover a device from ROMmon mode

Introduction

Describes the available upgrade methods for IE 9300 Rugged Series Switches to help administrators successfully update their network infrastructure.

This guide describes supported software upgrade and recovery methods for Cisco Catalyst IE9300 Rugged Series Switches.

Requirements

Describes the absence of specific requirements for this document, indicating that no particular prerequisites, configurations, or conditions need to be met before proceeding with the information presented.

Before performing an upgrade or recovery procedure, ensure that you have console access, a compatible Cisco IOS XE image, sufficient storage space, and an approved maintenance window. For switch stacks, verify that all members are reachable and in the expected state.

Best practice for hardware and software selection

Outlines the software and hardware version considerations and configuration practices for Cisco Catalyst IE9300 Rugged Series Switches.

The examples in this guide use Cisco Catalyst IE9300 Rugged Series Switches running Cisco IOS XE software.

Ensure that you understand the potential impact of any command when your network is live.

The information in this document was generated using devices configured in a specific lab environment. Each device began with its default configuration.

> **Note**
>
> Refer to the configuration guide for instructions on enabling these features on other Cisco platforms.

Background information

Describes upgrade procedures for IE 9300 Rugged Series Switches, covering both new and legacy methods that support bundle and install modes for system software updates.

This guide describes supported upgrade procedures for Cisco Catalyst IE9300 Rugged Series Switches operating in bundle mode or install mode

Software installation mode overview

Describes the different installation modes available for software deployment. This information helps administrators select the most appropriate method for their specific environment.

This section describes the available software installation modes, their characteristics, and their use cases.

Bundle mode

Describes a switch operating mode that runs traditional Cisco IOS XE using a single **.bin** file containing all necessary components. This mode simplifies the boot process by using a single boot statement to load the required software image.

In bundle mode, the switch boots Cisco IOS XE software directly from a single **.bin** image.

Boot system configuration

Use the **show boot** command to display the current boot variables.

```
Switch# show boot
                ---------------------------
                Switch 1
                ---------------------------
                Current Boot Variables:
                BOOT variable = flash:ie9k_iosxe.17.12.02.SPA.bin;
                Boot Variables on next reload:
                BOOT variable does not exist
                Manual Boot = no
                Enable Break = no
                Boot Mode = DEVICE
                iPXE Timeout = 0
```

Use the **boot system** command to set the boot statement. When you reload the switch, the switch boots into version 17.12.03 instead of 17.12.02.

```
Switch# configure terminal
Switch(config)# no boot system
Switch(config)# boot system flash:ie9k_iosxe.17.12.03.SPA.bin
Switch(config)# end
```

Note

Save the configuration so that the new boot variable remains in effect after the switch reloads.

Verify

Verify the configured boot variable before reloading the switch.

```
Switch# show running-config | include boot
boot system switch all flash:ie9k_iosxe.17.12.03.SPA.bin
```

```
Switch# show boot
                ---------------------------
                Switch 1
                ---------------------------
                Current Boot Variables:
                BOOT variable = flash:ie9k_iosxe.17.12.03.SPA.bin;
                Boot Variables on next reload:
                BOOT variable does not exist
                Manual Boot = yes
                Enable Break = no
                Boot Mode = DEVICE
```

```
                    iPXE Timeout = 0
```

Note

In a switch stack, configure the boot variable on the active switch. The configuration applies to all stack members. The **show running-config | include boot** output displays **boot system switch all** when the boot variable applies to the entire stack.

Install mode

Explains the newer and recommended mode to run that breaks the .bin file up into smaller .pkg files that must be loaded into memory independently.

Install mode is an operating mode that

- breaks the **.bin** file up into smaller **.pkg** files that must be loaded into memory independently of each other
- allows you to boot faster and utilize memory better, and
- is the newer and recommended mode to run.

Install mode process and components

The **.bin** file you download from software.cisco.com has all the **.pkg** files you need inside.

Note

Think of the **.bin** file as a **.zip** file you extract to obtain the **.pkg** files. A configuration file indicates which **.pkg** files are needed, and this process occurs when you boot in install mode.

The install mode boot process uses this sequence:

1. The boot statement is looked at to determine what configuration file to load (usually packages.conf).
2. The configuration file is opened and used to sort out what **.pkg** files to load.
3. The **.pkg** files are loaded and Cisco IOS completes the boot process.

The **.conf** file is a text file used as a pointer. You can open it and view its contents from the CLI. The **packages.conf** file demonstrates how to load IOS version 17.12.03.

Note

The system creates and updates **packages.conf** automatically. Do not edit this file manually. You can use the **more packages.conf** command to verify which package files it references.

```
Switch# more packages.conf
#! /usr/binos/bin/packages_conf.sh
<..snip..>
boot   rp 0 0   rp_boot ie9k-rpboot.17.12.03.SPA.pkg
iso   rp 0 0   rp_base ie9k-rpbase.17.12.03.SPA.pkg
iso   rp 0 0   rp_daemons ie9k-rpbase.17.12.03.SPA.pkg
iso   rp 0 0   rp_iosd ie9k-rpbase.17.12.03.SPA.pkg
iso   rp 0 0   rp_security ie9k-rpbase.17.12.03.SPA.pkg
iso   rp 0 0   rp_webui ie9k-webui.17.12.03.SPA.pkg
iso   rp 0 0   srdriver ie9k-srdriver.17.12.03.SPA.pkg
iso   rp 0 0   fp ie9k-rpbase.17.12.03.SPA.pkg
iso   rp 0 0   cc ie9k-rpbase.17.12.03.SPA.pkg
iso   rp 0 0   cc_srdriver ie9k-srdriver.17.12.03.SPA.pkg
#
# -start- superpackage .pkginfo
#
# pkginfo: Name: rp_super
# pkginfo: BuildTime: 2023-12-07_18.25
# pkginfo: ReleaseDate: Fri-08-Dec-23-02:07
# pkginfo: .BuildArch: arm64
# pkginfo: BootArchitecture: arm64
# pkginfo: .BootArch: arm64
# pkginfo: RouteProcessor: ie9k
# pkginfo: Platform: IE9K
# pkginfo: User: mcpre
# pkginfo: PackageName: universalk9
# pkginfo: Build: 17.12.03
# pkginfo: .SupportedBoards: ie9k
# pkginfo: .InstallModel:
# pkginfo: .PackageRole: rp_super
# pkginfo: .RestartRole: rp_super
# pkginfo: .UnifiedPlatformList: condor,sonic,clarke_lite,clarke_adv
# pkginfo: CardTypes:
# pkginfo: .CardTypes:
# pkginfo: .BuildPath:
/nobackup/mcpre/s2c-build-ws/binos/linkfarm/ie9k_universalk9-stage/img/rp_super_universalk9
# pkginfo: .Version: 17.12.03.0.1264.1702002330..IOSXE
# pkginfo: .InstallVersion: 1.0.0
# pkginfo: .InstallCapCommitSupport: yes
# pkginfo: .PKGUID: 11f3b996a6be59e1e4c1b5303df410901527196c
#
# -end- superpackage .pkginfo
```

An install-mode upgrade consists of these operations:

1. Copy the new **.bin** to switch flash.
2. Add the image to the install repository and extract its **.pkg** files from the **.bin** file.
3. The installation process updates **packages.conf** and reloads the switch using the new package files.

Usually, a single command completes all these steps.

Note

The Cisco Catalyst IE9300 Rugged Series Switches ship in install mode by default.

Use the **show version** command to verify the current operating mode of the switch.

```
<../snip>
cisco IE-9320-22S2C4X (ARM64) processor with 587715K/3071K bytes of memory.
Processor board ID FDO2713JK6Z
4 Virtual Ethernet interfaces
48 Gigabit Ethernet interfaces
8 Ten Gigabit Ethernet interfaces
2048K bytes of non-volatile configuration memory.
4013980K bytes of physical memory.
523264K bytes of Crash Files at crashinfo:.
523264K bytes of Crash Files at crashinfo-2:.
2650112K bytes of Flash at flash:.
2650112K bytes of Flash at flash-2:.
3883008K bytes of SD Flash at sdflash:.
3883008K bytes of SD Flash at sdflash-2:.
Base Ethernet MAC Address          : aaa.bbb.ccc
Motherboard Assembly Number        : 00-0000-000
Motherboard Serial Number          : 01234567
Model Revision Number              : 08
Motherboard Revision Number        : 02
Model Number                       : 1111-1111-1111
System Serial Number               : 1234
Switch Ports Model              SW Version        SW Image              Mode

------ ----- -----              ----------        ----------            ----

*    1 29    IE-9320-22S2C4X    17.12.01          IE9K_IOSXE            INSTALL

     2 29    IE-9320-22S2C4X    17.12.01          IE9K_IOSXE            INSTALL
```

Upgrade methods

Describes the various approaches and procedures available for upgrading system software. These methods include in-place upgrades, rolling upgrades, and maintenance window upgrades to ensure minimal service disruption.

This reference provides an overview of the available system software upgrade methods to assist in planning maintenance with minimal service impact.

Configure autoupgrade on stacked switches

Configure autoupgrade to automatically upgrade a standby switch when the active switch detects a version mismatch in a stacked environment.

When the active switch detects a software version mismatch, it upgrades the standby switch automatically. This ensures consistent software versions for all switches in the stack.

This procedure applies only when switches are stacked. The active switch must run in install mode. The standby switch must boot into a valid image for autoupgrade to start.

Before you begin

- This procedure applies only to a switch stack operating in install mode.
- Before upgrading, use the **install remove inactive** command to clear unused files and avoid errors due to limited space. Then, use the **show install log** command to check the upgrade log files.
- If the active switch runs in install mode, use autoupgrade. The active switch upgrades the standby switch regardless of the loaded image format. The standby switch must boot into a valid image for the process to start. Autoupgrade does not recover a switch from ROMmon.

Follow these steps to configure autoupgrade on stacked switches:

Procedure

1. Use the **software auto-upgrade enable** command to enable automatic software upgrades for incompatible stack members.

   Example

   ```
   Switch# configure terminal
   Switch(config)# software auto-upgrade enable
   Switch(config)# end
   ```

2. (Optional) Use the **install autoupgrade** command to run the upgrade manually.

   Example

   ```
   Switch# install autoupgrade
   ```

Install mode commands for Cisco devices

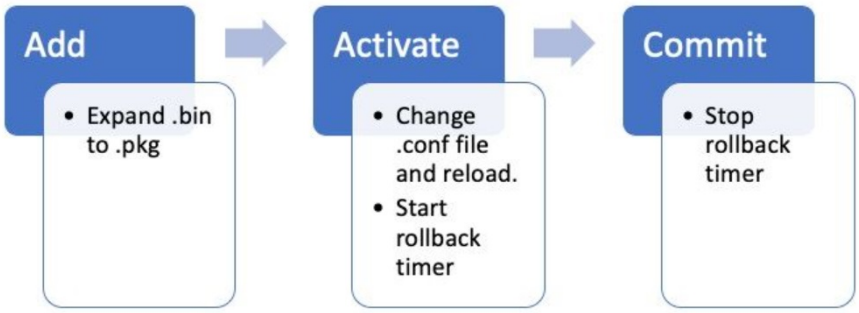
Describes the use of install commands for software installation and management procedures on Cisco devices. This mode provides a streamlined approach to managing software packages and system upgrades.

This section provides information regarding the use of the **install** commands for device management.

Install software packages in install mode

Describes the four-part install mode upgrade process and the roles of each phase.

The install mode upgrade process consists of add, activate, and commit phases. These phases prepare the software package, activate the new image, and confirm the upgrade after the switch reloads.



Install mode commands

Lists the commands used to manage software installation and upgrades in install mode on Cisco switches.

Install mode commands allow you to manage software upgrades on Cisco switches using a package-based installation method. These commands provide control over adding new software packages, activating them, committing changes, and managing the upgrade process with rollback protection.

Restrictions

Your boot statement must always be packages.conf in install mode. If you attempt to change the boot statement to an alternative **.conf** file during an upgrade, the upgrade might fail. After you run the **add** command, either activate the added package or remove it to begin again.

Summary

To move your switch from bundle mode to install mode, first change your boot statement to **packages.conf** if it is not already set. Then, add the **.bin** file onto flash by using USB, or an SD card. If there is not enough space to copy the file onto the switch, remove unnecessary files.

- Use these commands to complete the upgrade in one step, or follow each step separately if you prefer.

```
configure terminal
no boot system
boot system flash:packages.conf
end
write
```

install add file flash:<new .bin file> activate commit

Note

For the initial conversion from bundle mode to install mode, use the one-step **install add file image-filename activate commit** command. The following error occurs if you choose to perform the installation in separate steps.

```
Switch# install add file flash:image-filename.bin activate commit
```

Note

When the switches are stacked, the boot system command set on the active switch applies to all members of the stack.

Install add

- When you start the upgrade, only the current version is available.

```
Switch# show install summary
[ Switch 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   C    17.12.01.0.286
--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

- In order to get started, you must add the new version to the list of versions that you want to work with.

```
Switch# install add file flash:ie9k_iosxe.17.12.03.SPA.bin
```

- Once it is added, you can see the **.pkg** file in flash and it lists 17.12.03 as an inactive version.

```
Switch# dir | include .pkg

55328   -rw-          63996816  Apr 24 2024 05:15:28 +00:00
ie9k-rpboot.17.12.03.SPA.pkg
55331   -rw-         513676280  May 28 2024 06:06:17 +00:00
ie9k-rpbase.17.12.03.SPA.pkg
55333   -rw-          14345208  May 28 2024 06:04:52 +00:00
ie9k-webui.17.12.03.SPA.pkg
55332   -rw-           4637692  May 28 2024 06:04:49 +00:00
ie9k-srdriver.17.12.03.SPA.pkg

Switch# show install inactive
[ R0 R1 ] Inactive Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
```

```
--------------------------------------------------------------------------------
IMG   I    17.12.03.0.1264 <-- Installed but still Inactive (I)
```

Install activate

- Reload the switch to activate the new version. This updates the **.conf** file so it references the new **.pkg** files.

  install activate

- The system prompts you to confirm the reload before proceeding.

```
Switch# show install summary
[ R0 R1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   U    17.12.03.0.1264

--------------------------------------------------------------------------------
Auto abort timer: active on install_activate, time before rollback - 01:52:08-
  when this hits zero,
the device reloads back to original version.
--------------------------------------------------------------------------------
Switch# show install uncommitted
[ R0 R1 ] Uncommitted Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   U    17.12.03.0.1264
```

The table shows that version 17.12.03 is activated but not yet committed.

> **Note**
>
> Activating an image starts the auto-abort timer. The default timer is 120 minutes. Commit the installation before the timer expires, or the switch rolls back to the previously committed image and reloads.

Install commit

- To stop the rollback timer and finish the upgrade process, you need to commit the new version.

  Use the **install commit** command to stop the auto-abort timer and make the active image persistent across reloads.

```
Switch# install commit
```

- This is the final step in the upgrade process. Verify that the new image is active and committed.

```
Switch# show install summary
[ R0 R1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
```

```
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
IMG   C    17.12.03.0.1264 - Now Active and Committed

--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

Install abort

If you detect a problem after activating an image but before committing it, use the **install abort** command to cancel the installation. The switch reloads and restores the previously committed image.

```
Switch# install abort
```

Prompt level

Add **prompt-level none** to a supported one-step install command to suppress confirmation prompts. When the switch reaches the activation stage, it reloads without requesting confirmation.

```
Switch# install add file flash:ie9k_iosxe.17.12.03.SPA.bin activate commit
prompt-level none
```

Auto abort timer

- If an upgrade takes too long, the switch automatically aborts the upgrade to prevent issues. The default timer is 120 minutes. Stop the timer manually if needed.

  Use the **install auto-abort-timer stop** command to stop the auto-abort timer and prevent the switch from automatically rolling back to the previously committed software image. You can set the rollback timer when you do the activate step.

```
Switch# install activate auto-abort-timer ?
  <30-1200>  Number of minutes the abort timer is set for
```

If you run **install activate** separately, verify the new image and run **install commit** before the auto-abort timer expires. If you use a one-step command that includes **activate commit**, no additional commit operation is required after the command completes successfully.

Configure bundle mode software upgrade

Configure the classic method of software upgrade with the use of a boot statement that points to the .bin file (versus a .pkg file).

This task enables you to perform a software upgrade using the classic bundle mode method, which uses a boot statement pointing to a **.bin** file rather than a **.pkg** file.

Use bundle mode, the traditional approach to software upgrades on Cisco switches. Change the boot configuration as required to perform the upgrade.

Before you begin

Complete these tasks before you start this procedure.

- Load the Cisco IOS file onto the switch.
- If your switches use a stack configuration, copy the **.bin** file to each switch in the stack.

Perform these steps to configure bundle mode software upgrade:

Procedure

Change the boot statement and reload the switch.

Example

```
configure terminal
no boot system
boot system flash:<new filename>
end
write
reload
```

Resolve a version mismatch in stacked switches

Configure a resolution for V-mismatch error that occurs when software versions differ between switches in a stack.

Resolve V-mismatch errors that occur when the software running on a switch is different from the active switch in a stack configuration.

A V-mismatch occurs in a stack configuration when a member switch runs a software version different from the active switch. Auto-upgrade may resolve the issue in install mode, but manual steps are required in bundle mode. This task covers both modes and applies only to stacked switches.

Before you begin

- Ensure you have access to both the stack and individual switch consoles if required.
- Identify whether the stack is running in install or bundle mode.

Procedure

1. When a switch in the stack enters the V-Mismatch state, choose one of the following recovery options.
   - **Option 1**: Remove the mismatched switch from the stack, upgrade it to the correct software version, and then add it back to the stack.
   - **Option 2**: You cannot change the boot statement on a mismatched switch. However, if you have console access and can reach ROMmon, you can manually boot the switch into the correct software version.

     > ⚠ **Caution**
     >
     > Ensure that you have console access to the switch before using this option so that you can recover the switch if it does not boot.

   - **Option 3**: The first two options usually recover the mismatched switch without bringing down the stack. However, if this is a new deployment, or if you are okay with reloading the whole stack, copy the image file from the mismatched switch to the other switches and synchronize them to the new switch instead. This option can help when you are deploying a new stack and do not have physical access, because you can complete all steps using Telnet. After the switches are synchronized, upgrade all switches to the required software version.

2. Use this procedure if you choose Option 2.
   a. Stack the switch and allow it to boot into the **V-Mismatch** state.
   b. Use the **dir ?** command to identify the available flash file systems.

```
Switch# dir ?
  /all              List all files
  /recursive        List files recursively
  all-filesystems   List files on all filesystems
  cns:              Directory or file name
  crashinfo-1:      Directory or file name
  crashinfo-2:      Directory or file name
  crashinfo:        Directory or file name
  flash-1:          Directory or file name
  flash-2:          Directory or file name
  flash:            Directory or file name
```

   c. Remove only image files that are no longer required. Do not remove the currently booted image or the recovery image until you verify that the replacement image is available.
   d. Copy the new **.bin** file you want to boot to the switch flash.
   e. Power cycle the switch. After the existing image is deleted, the switch does not boot normally and enters ROMmon.
   f. From the console, use the **switch: boot flash:image-name.bin** command in ROMmon to manually boot the new image that you copied to flash.

```
switch: boot flash:ie9k_iosxe.17.12.03.SPA.bin
```

   g. After the switch boots with the correct image, allow the configuration to synchronize from the active switch.
   h. Use the **show switch** command to verify that the member reaches the `Ready` state.

Example

```
Switch# show switch
Switch/Stack Mac Address : f8b7.e206.4900 - Local Mac Address
Mac persistency wait time: Indefinite
                                             H/W   Current
Switch#   Role    Mac Address     Priority Version  State
-------------------------------------------------------------------------------

 1       Member  aaaa.bbbb.cccc     15     V01     V-Mismatch  <---Indicates
 a version mismatch
*2       Active  dddd.eeee.ffff    14     V01     Ready
```

Recover a device from ROMmon mode

Configure device recovery from ROMmon mode when no valid image is available using SD-FLASH, USB, or FLASH storage.

Restore device operation by loading a valid operating-system image from an alternate storage location when the device is in ROMmon mode.

If the device enters ROMmon mode because no valid boot image is configured, you can boot a valid Cisco IOS XE image from an SD card or USB drive.

Before you begin

Ensure that the Cisco IOS XE image is available in one of these locations:

- SD card
- USB drive
- Flash storage

Procedure

1. From the ROMmon prompt, identify the storage location that contains the Cisco IOS XE image.
2. Choose one of the following boot options based on where the software image is available.
   - You can boot directly from an SD card using the command:

     ```
     switch: boot sdflash:ie9k_iosxe.17.12.03.SPA.bin
     ```

   - You can boot directly from a USB drive using the command:

     ```
     switch: boot usbflash1:ie9k_iosxe.17.12.03.SPA.bin
     ```

   - You can boot directly from flash storage using the command:

     ```
     switch: boot flash:ie9k_iosxe.17.12.03.SPA.bin
     ```

3. Boot the device using the command for the selected storage location.

   Example

   ```
   switch: boot sdflash:ie9k_iosxe.17.12.03.SPA.bin
   ```

   Use the image filename that matches the software version that you want to boot.

   Note

   For more information about ROMmon recovery, see Recover from Corrupt or Missing File Image or in ROMmon Mode