



## CHAPTER 6

# Configuring the Cisco IOS XE In Service Software Upgrade Process

---



**Note**

---

ISSU is available in Cisco IOS XE 3.1.0.SG and later releases.

---

Operating on redundant systems, the In Service Software Upgrade (ISSU) process allows Cisco IOS XE software to be updated or otherwise modified while packet forwarding continues. In most networks, planned software upgrades are a significant cause of downtime. ISSU allows Cisco IOS XE software to be upgraded while packet forwarding continues. This increases network availability and reduces downtime caused by planned software upgrades. This document provides information about ISSU concepts and describes the steps taken to perform ISSU in a system.

Topics include:

- [Prerequisites to Performing ISSU, page 6-2](#)
- [About Performing ISSU, page 6-3](#)
- [How to Perform the ISSU Process, page 6-15](#)
- [Cisco High Availability Features in Cisco IOS XE 3.1.0 SG, page 6-35](#)



**Note**

---

For complete syntax and usage information for the switch commands used in this chapter, look at the *Cisco Catalyst 4500 Series Switch Command Reference* and related publications at this location:

<http://www.cisco.com/en/US/products/hw/switches/ps4324/index.html>

If the command is not found in the Catalyst 4500 Command Reference, it is located in the larger Cisco IOS library. Refer to the *Cisco IOS Command Reference* and related publications at this location:

<http://www.cisco.com/en/US/products/ps6350/index.html>

---

## Related Documents

Related Topic	Document Title
Performing ISSU	<i>Cisco IOS Software: Guide to Performing In Service Software Upgrades</i>
Information about Cisco Nonstop Forwarding	<i>Cisco Nonstop Forwarding</i> <a href="http://www.cisco.com/en/US/docs/ios/12_2s/feature/guide/fsnsf20s.html">http://www.cisco.com/en/US/docs/ios/12_2s/feature/guide/fsnsf20s.html</a>
Information about Stateful Switchover	<i>Stateful Switchover</i> <a href="http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/sso120s.html">http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/sso120s.html</a>

## Prerequisites to Performing ISSU

Before performing ISSU, you must meet these prerequisites:

- The type of the existing and target image must match. You cannot upgrade from a Universal Lite image to a Universal image (and vice versa) without experiencing several minutes of traffic loss. The same restriction applies between crypto and non-crypto images.
- The active and the standby supervisor engines must have the same supervisor engine hardware (same model, same memory, and so on).
- The new and old Cisco IOS XE software images must be loaded into the file systems (bootflash, SD card, or USB) of both the active and the standby supervisor engines before you begin the ISSU process.

The old software image should be available either in bootflash, SD card, or USB and the system should have been booted from one of these locations because the boot variable must be changed before the ISSU process starts.



**Note** **auto-boot** must be enabled for ISSU to succeed.

- Stateful Switchover (SSO) must be configured and the standby supervisor engine should be in STANDBY HOT state.  
These commands indicate whether SSO is enabled: **show module**, **show running-config**, **show redundancy state**.  
If you do not have SSO enabled, see the *Stateful Switchover* document for further information on how to enable and configure SSO.
- Nonstop Forwarding (NSF) must be configured and working properly. If you do not have NSF enabled, see the *Cisco Nonstop Forwarding* document for further information on how to enable and configure NSF.
- Before you perform ISSU, ensure that the file system for both the active and the standby supervisor engines contains the new ISSU-compatible IOS XE software. The current Cisco IOS XE version running in the system must also support ISSU.

You can enter various commands on the Catalyst 4500 series switch to determine supervisor engine versioning and Cisco IOS XE software compatibility. Alternatively, you can use the ISSU application on Cisco Feature Navigator to determine this.

- If you enter the **no ip routing** command, ISSU falls back from SSO to RPR mode, resulting in traffic loss.
- Autoboot is turned on and the current booted image matches the one specified in the BOOT environmental variable. For details on how to configure and verify these, please refer to "[Modifying the Boot Field and Using the boot Command](#), page 3-27.
- If you enter the **no ip routing** command, ISSU falls back from SSO to RPR mode, resulting in traffic loss.

## About Performing ISSU



### Note

---

Do not make any hardware changes while performing ISSU.

---

Before you perform ISSU, you should understand the following concepts:

- [Stateful Switchover](#), page 6-3
- [NSF](#), page 6-5
- [ISSU Process](#), page 6-6
- [Performing an ISSU Upgrade: 2 Methods](#), page 6-11
- [Changeversion Process](#), page 6-12
- [Guidelines for Performing ISSU](#), page 6-13
- [Compatibility Matrix](#), page 6-13
- [SNMP Support for ISSU](#), page 6-14
- [Compatibility Verification Using Cisco Feature Navigator](#), page 6-14

## Stateful Switchover

Development of the SSO feature is an incremental step within an overall program to improve the availability of networks constructed with Cisco IOS XE switches.

In specific Cisco networking devices that support dual supervisor engines, SSO takes advantage of supervisor engine redundancy to increase network availability. SSO achieves this by establishing one of the supervisor engines as the active processor while the other supervisor engine is designated as the standby processor. Following an initial synchronization between the two supervisor engines, SSO dynamically synchronizes supervisor engine state information between them in real-time.

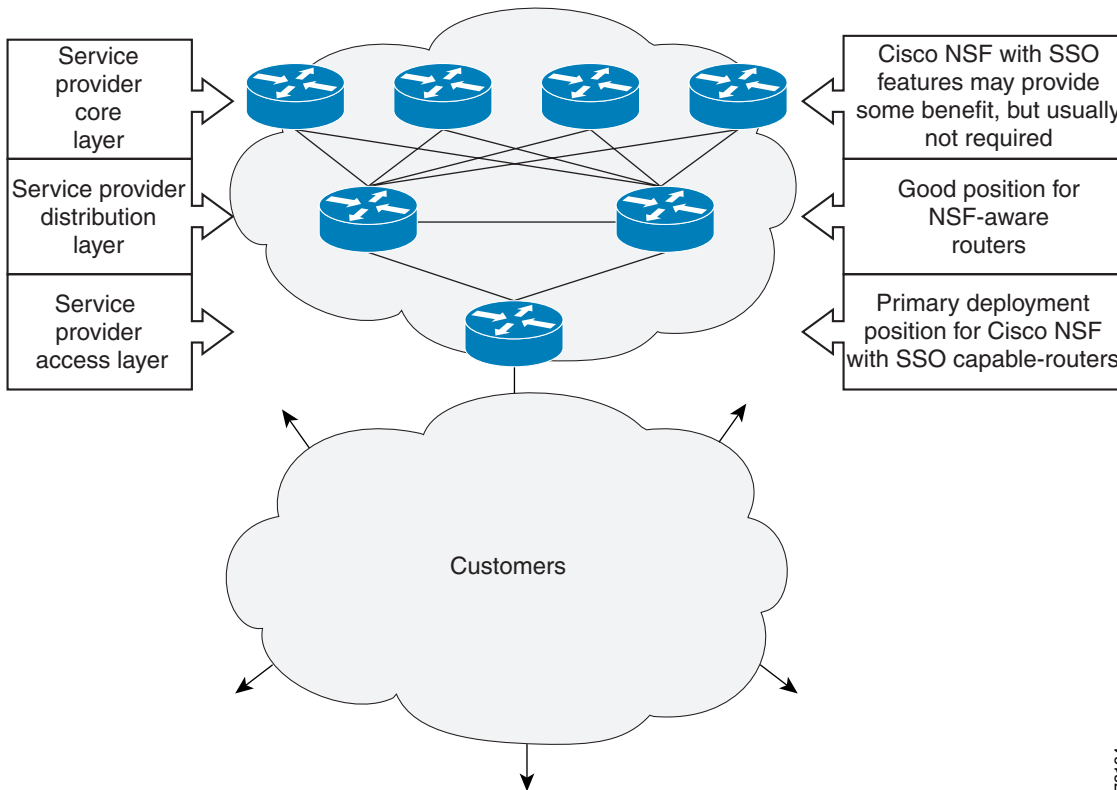
A switchover from the active to the standby supervisor engine occurs when the active supervisor engine fails or is removed from the networking device.

Cisco NSF is used with SSO. Cisco NSF allows the forwarding of data packets to continue along known routes while the routing protocol information is being restored following a switchover. With Cisco NSF, peer networking devices do not experience routing flaps, which reduce loss of service outages for customers.

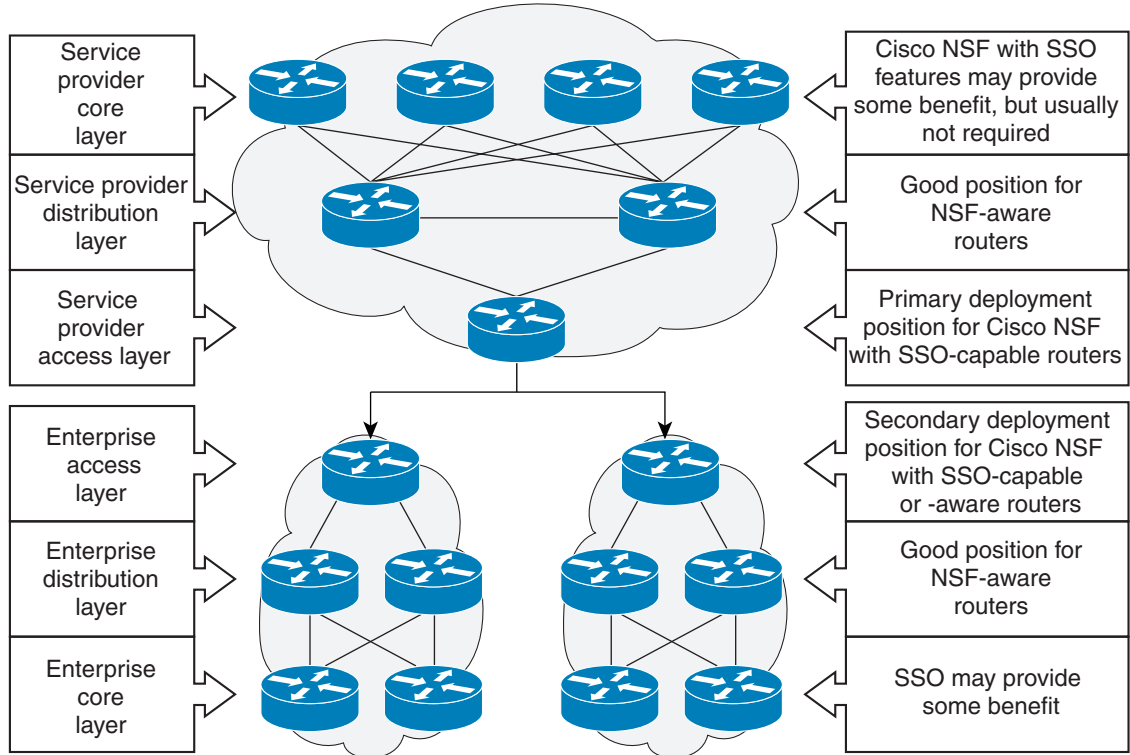
[Figure 6-1](#) illustrates how SSO is typically deployed in service provider networks. In this example, Cisco NSF with SSO is enabled at the access layer (edge) of the service provider network. A fault at this point could result in loss of service for enterprise customers requiring access to the service provider network.

For Cisco NSF protocols that require neighboring devices to participate in Cisco NSF, Cisco NSF-aware software images must be installed on those neighboring distribution layer devices. Depending on your objectives, you may decide to deploy Cisco NSF and SSO features at the core layer of your network. Doing this can help reduce the time required to restore network capacity and service for certain failures, which leads to additional availability.

**Figure 6-1 Cisco NSF with SSO Network Deployment: Service Provider Networks**



Additional levels of availability may be gained by deploying Cisco NSF with SSO at other points in the network where a single point of failure exists. [Figure 6-2](#) illustrates an optional deployment strategy that applies Cisco NSF with SSO at the enterprise network access layer. In this example, each access point in the enterprise network represents another single point of failure in the network design. In the event of a switchover or a planned software upgrade, enterprise customer sessions would continue uninterrupted through the network in this example.

**Figure 6-2 Cisco NSF with SSO Network Deployment: Enterprise Networks**

For further information on SSO, see the *Stateful Switchover* document.

## NSF

Cisco NSF works with the SSO feature in Cisco IOS XE software. SSO is a prerequisite of Cisco NSF. NSF works with SSO to minimize the amount of time a network is unavailable to its users following a switchover. The main objective of Cisco NSF is to continue forwarding IP packets following a supervisor engine switchover.

Usually, when a networking device restarts, all routing peers of that device detect that the device went down and then came back up. This transition results in what is called a routing flap, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. Cisco NSF helps to suppress routing flaps in SSO-enabled devices, thus reducing network instability.

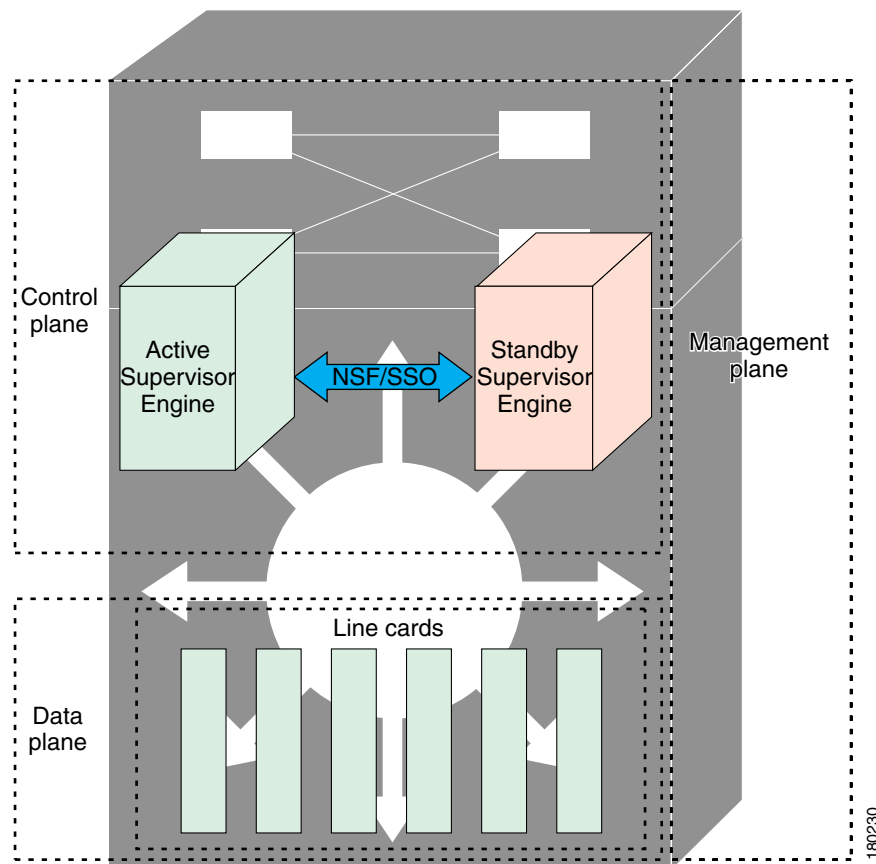
Cisco NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored following a switchover. With Cisco NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded while the standby supervisor engine assumes control from the failed active supervisor engine during a switchover. The ability of physical links to remain up through a switchover and to be kept current with the Forwarding Information Base (FIB) on the active supervisor engine is key to Cisco NSF operation.

## ISSU Process

The ISSU process allows you to perform a Cisco IOS XE software upgrade or downgrade while the system continues to forward packets. (For an illustration of the commands used during the ISSU process, refer to [Figure 6-8](#).) Cisco IOS XE ISSU takes advantage of the Cisco IOS XE high availability infrastructure—Cisco NSF with SSO and hardware redundancy—and eliminates downtime associated with software upgrades by allowing changes while the system remains in service (see [Figure 6-3](#)).

SSO and NSF mode support configuration and runtime state synchronization from the active to the standby supervisor engine. For this process, the IOS XE software image on both the active and the standby supervisor engines must be the same. When images on active and standby supervisor engines are different, ISSU allows the two supervisor engines to be kept in synchronization even when these two versions of Cisco IOS XE support different sets of features and commands.

**Figure 6-3** High Availability Features and Hardware Redundancy in the ISSU Process



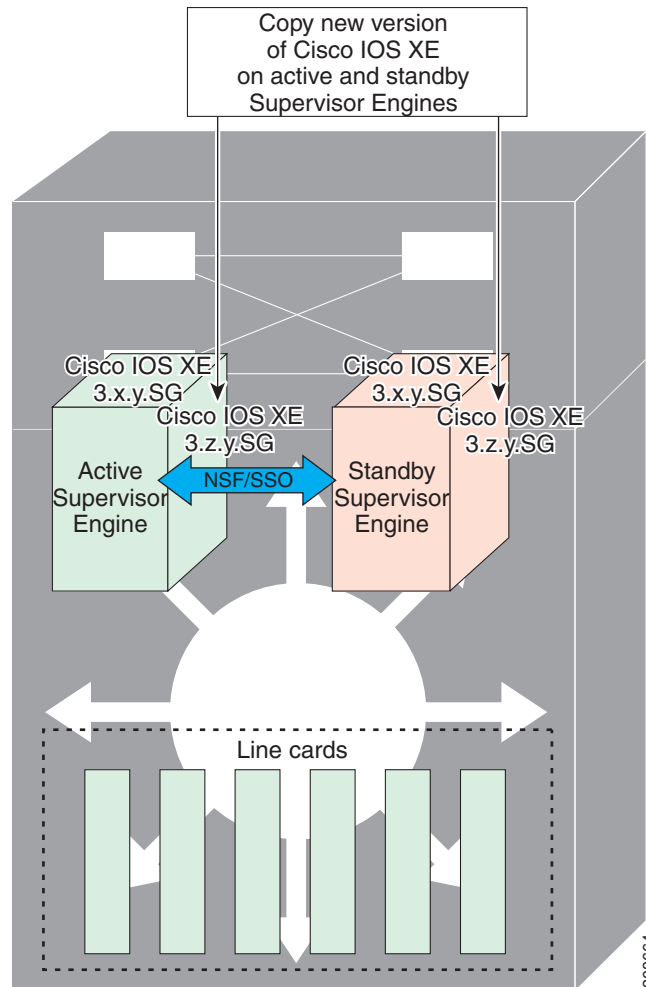
180230

An ISSU-capable switch consists of two supervisor engines (active and standby) and 200 or more linecards. Before initiating the ISSU process, copy the Cisco IOS XE software into the file systems of both supervisor engines (see Figure 6-4).

**Note**

In the following figure, Cisco IOS XE 3.x.y.SG represents the *current* version of Cisco IOS XE 3.z.y.SG represents the image you are migrating to.

**Figure 6-4 Copy New Version of Cisco IOS XE Software on Both Supervisor Engines**

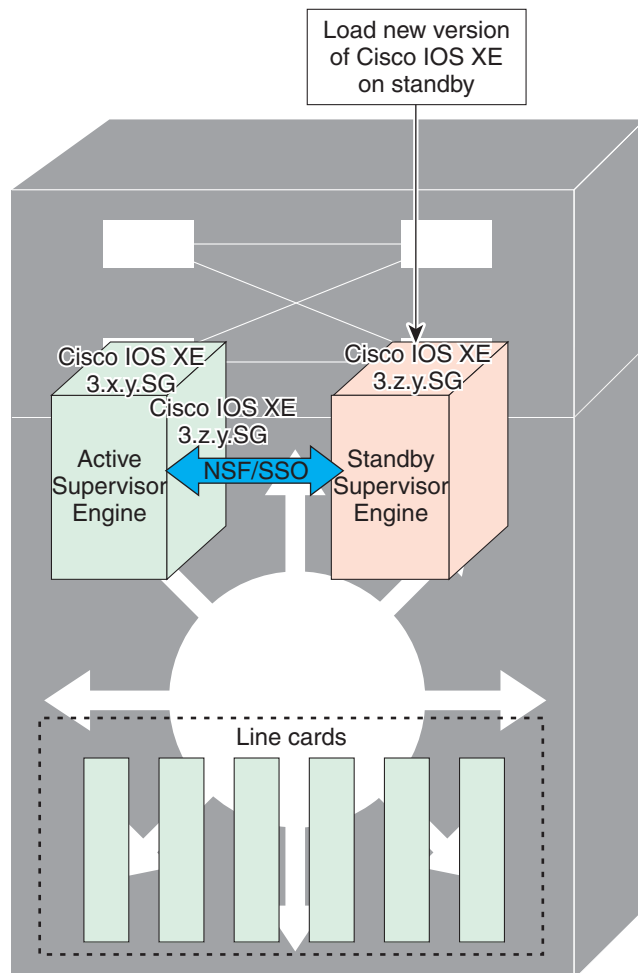


After you have copied the Cisco IOS XE software to both file systems, load the new version of Cisco IOS XE software onto the standby supervisor engine (see [Figure 6-5](#)).

**Note**

Without the ISSU feature, SSO/NSF cannot function between the active and standby supervisor engines when they are running different versions of the Cisco IOS XE image.

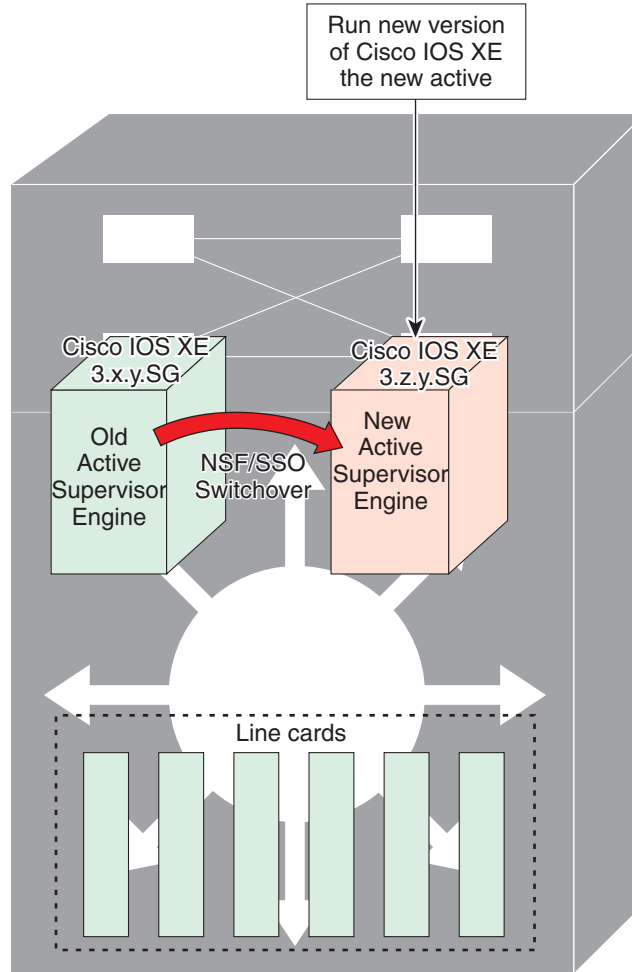
**Figure 6-5** Load New Version of Cisco IOS XE Software on the Standby Supervisor Engine





After a switchover (NSF/SSO, not RPR), the standby supervisor engine takes over as the new active supervisor engine (see [Figure 6-6](#)).

**Figure 6-6** Switch Over to Standby Supervisor Engine



The former active supervisor engine is loaded with an old Cisco IOS XE image so that if the new active supervisor engine experiences problems, you can abort and conduct a switchover to the former active, which is already running the old software image. Next, the former active supervisor engine is loaded with the new version of Cisco IOS XE software and becomes the new standby supervisor engine (see Figure 6-7).

**Figure 6-7** Load New Standby Supervisor Engine with New Cisco IOS XE Software

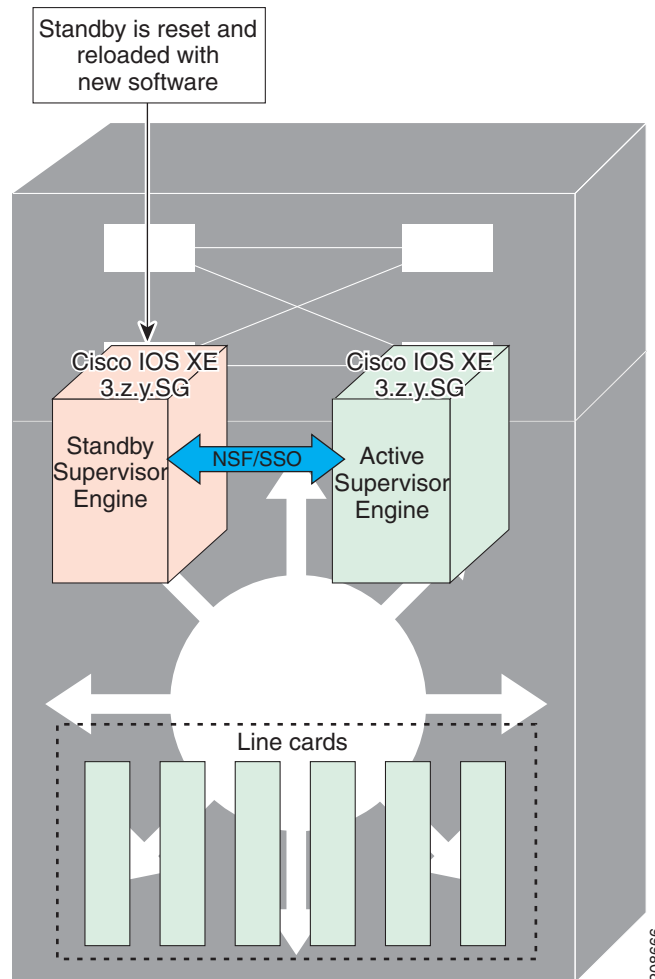
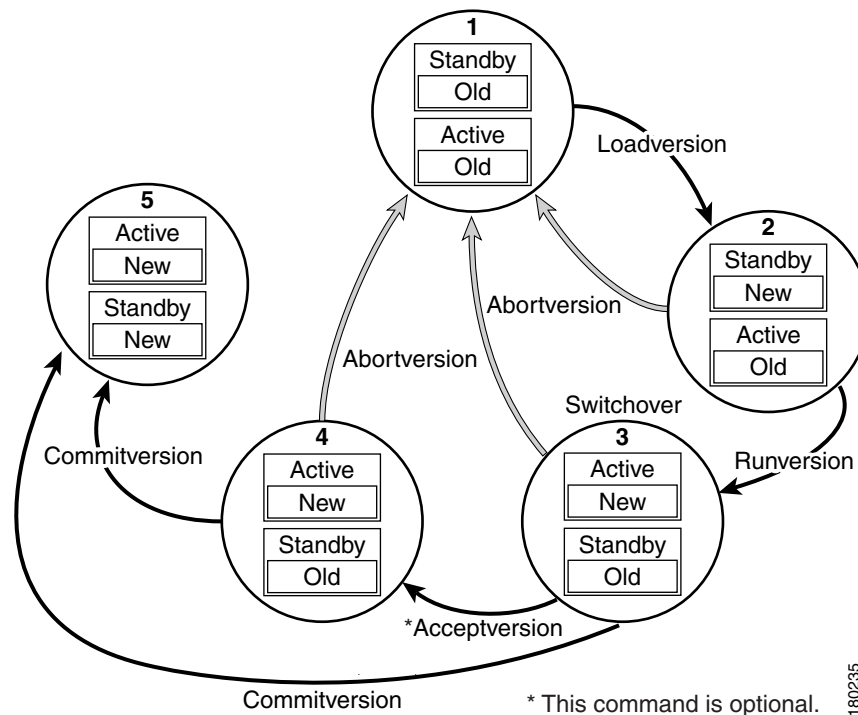


Figure 6-8 shows the steps during the ISSU process.

Figure 6-8 Steps During the ISSU Process



**Note** **Accept version** stops the rollback timer.

## Performing an ISSU Upgrade: 2 Methods

There are two ways to perform an ISSU upgrade: manually, with four commands; or automatically, with one command.

The normal ISSU upgrade process involves issuing four separate ISSU exec commands (**issu loadversion**, **issu runversion**, **issu acceptversion**, **issu commitversion**) along with additional show command invocations to evaluate the success of each command before proceeding. Although the ISSU process is complicated, you should not expect disruption of service. The use of multiple ISSU commands dictates an additional level of care to ensure no service disruption. However, in some scenarios, this upgrade procedure might be cumbersome and of minimal value. A typical example is during a network upgrade that involves performing an ISSU upgrade on a large number of Catalyst 4500 switches. In these cases, we recommend that you first perform the normal (four command) ISSU upgrade procedure on one switch (possibly in a lab environment) to verify successful upgrade. Then, use a single **issu changeversion** command to perform an automatic ISSU on the rest of the Catalyst 4500 switches in the network.

## Changeversion Process

The **issu changeversion** command launches a single-step complete ISSU upgrade cycle. It performs the logic for all four of the standard commands (**issu loadversion**, **issu runversion**, **issu acceptversion**, and **issu commitversion**) without user intervention, streamlining the upgrade through a single CLI step.

Additionally, **issu changeversion** allows the upgrade process to be scheduled for a future time. This enables you to stage a number of systems to perform upgrades sequentially when a potential disruption would be least harmful.

After the standby supervisor engine initializes and the system reaches a terminal state (RPR/SSO), the upgrade process is complete and the BOOT variable is permanently written with the new IOS XE software image. Hence, a reset on any RP will keep the system booting the new software image. Console and syslog messages will be generated to notify anyone monitoring the upgrade that the state transition has occurred.

Similar to the normal ISSU upgrade procedure, the in-progress upgrade procedure initiated by the **issu changeversion** command can be aborted with the **issu abortversion** command. If the system detects any problems or detects an unhealthy system during an upgrade, the upgrade might be automatically aborted.

When the **issu runversion** command is entered during the four step manual upgrade process, if any incompatible ISSU clients exist, the upgrade process reports them and their side effects, and allows the user to abort the upgrade. While performing a single-step upgrade process, when the process reaches the runversion state, it will either automatically continue with the upgrade provided the base clients are compatible, or automatically abort because of client incompatibility. If the user wants to continue the upgrade procedure in RPR mode, the user must use the normal ISSU command set and specify the **force** option when entering the **issu loadversion** command.

### Changeversion: Quick Option (LV to INIT)

The **issu changeversion** command provides an optional quick command option that can reduce the time required to perform the automatic ISSU upgrade. When the **quick** command option is applied, the ISSU upgrade state transition differs from that described previously. With this option, the software logic up the loadversion stage remains the same as previously described, and the logic that performs runversion and commitversion is combined. This logic skips the step in the upgrade procedure that loads the old software version on the new standby (old active) supervisor, reducing the time required for the automatic ISSU upgrade by about a third.

### Scheduled Changeversion: "in" and "at" Options

**issu changeversion** provides **in** and **at** command options that enable you to schedule a future automatic ISSU upgrade.

The **at** command option schedules an automatic ISSU upgrade to begin at a specific time. This option specifies an exact time (*hh:mm*, 24 hour format) in the next 24 hours at which the upgrade will occur.

The **in** command option schedules an automatic ISSU upgrade to begin after a certain amount of time has elapsed. This option specifies the number of hours and minutes (*hh:mm* format) that must elapse before an upgrade will occur, with a maximum value of 99:59.

## Changeversion Deployment Scenario

The typical **issu changeversion** command usage scenario is for experienced users with a large installed base. These users typically validate a new image using a topology and configuration similar to their production network. The validation process should be done using both the existing multi-command process and the new **issu changeversion** command process. Once users certify an IOS XE software image and want to roll it out broadly, they can use the single command process to perform an efficient upgrade of their network.

## Aborting an In-Progress Changeversion Procedure

The **issu changeversion** command functionality is designed to perform an ISSU software upgrade without user intervention. However, status messages are displayed to the console as the upgrade transitions through the various states. If any anomalies are noticed during the automatic upgrade, perhaps with peers or other parts of the network, you can use the **issu abortversion** command to manually abort the upgrade at any point in the process prior to the commitversion operation.

## Guidelines for Performing ISSU

Be aware of the following guidelines while performing the ISSU process:

- Even with ISSU, it is recommended that upgrades be performed during a maintenance window.
- The new features should not be enabled (if they require change of configuration) during the ISSU process.



---

**Note** Enabling them will cause the system to enter RPR mode because commands are only supported on the new version.

---

- In a downgrade scenario, if any feature is not available in the downgrade revision of the Cisco IOS XE software handle, that feature should be disabled prior to initiating the ISSU process.

## Compatibility Matrix

ISSU requires additional information to determine compatibility between software versions. Therefore, a compatibility matrix is defined that contains information about other IOS XE software image with respect to the one in question.

This compatibility matrix represents the compatibility of two software versions, one running on the active and the other on the standby supervisor engine, and to allow the system to determine the highest operating mode it can achieve. Incompatible versions will not be able to progress to SSO operational mode.

The compatibility matrix represents the compatibility relationship a Cisco IOS XE software image has with all of the other Cisco IOS XE software versions within the designated support window (for example, all of those software versions the IOS XE software image “knows” about) and is populated and released with every IOS XE software image. The matrix stores compatibility information between its own release and prior releases. It is always the newest release that contains the latest information about compatibility with existing releases in the field. The compatibility matrix is available within the Cisco IOS XE software image and on Cisco.com so that users can determine in advance whether an upgrade can be done using the ISSU process.

You can perform the ISSU process when the old and new Cisco IOS XE software are compatible. The compatibility matrix information stores the compatibility among releases as follows:

- **Compatible**—The base-level system infrastructure and all optional HA-aware subsystems are compatible. An in-service upgrade or downgrade between these versions will succeed with minimal service impact. The matrix entry designates the images to be compatible (C).
- **Base-level compatible**—One or more of the optional HA-aware subsystems is not compatible. An in-service upgrade or downgrade between these versions will succeed; however, some subsystems will not be able to maintain state always during the transition from the old to the new version of Cisco IOS XE. The matrix entry designates the images to be base-level compatible (B).
- **Incompatible**—A core set of system infrastructure exists in Cisco IOS XE that must be able to interoperate in a stateful manner for SSO to function correctly. If any of these required features or subsystems is not interoperable, then the two versions of the Cisco IOS XE software image are declared to be incompatible. An in-service upgrade or downgrade between these versions is not possible. The matrix entry designates the images to be incompatible (I). The system operates in RPR mode during the upgrade process when the versions of Cisco IOS XE at the active and standby supervisor engines are incompatible.
- Cisco IOS XE determines the compatibility between the active and the standby IOS XE software dynamically during STANDBY boot up. The matrix is represented by “x”.

To display the compatibility matrix data between two software versions on a given system, enter the **show issu comp-matrix stored** command.


**Note**

This command is useful *only for verification purposes* because it is available *only after* the ISSU process has started. You might want to check the compatibility matrix prior to starting ISSU. Use the Feature Navigator to obtain the needed information

<http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp>

## SNMP Support for ISSU

SNMP for SSO provides a mechanism for synchronizing the SNMP configurations and the MIBs that support SSO from the active supervisor engine to the standby supervisor engine.

## Compatibility Verification Using Cisco Feature Navigator

The ISSU application on Cisco Feature Navigator allows you to:

- Select a specific software bundle.
- Identify which software images are compatible with the selected software image.

- Compare two IOS XE software images and understand the compatibility level of the software images (that is, compatible, base-level compatible, and incompatible), or dynamically determined.
- Compare two software images and see the client compatibility for each ISSU client.
- Provide links to release notes for the software image.

## How to Perform the ISSU Process

Unlike SSO, which is a mode of operation for the device and a prerequisite for performing ISSU, the ISSU process is a series of steps performed while the switch is in operation. The steps result in an upgrade to new or modified Cisco IOS XE software, and have a minimal impact to traffic.

**Note**

For an illustration of the process flow for ISSU, refer to [Figure 6-8 on page 6-11](#).

This section includes the following topics:

- [Verifying the ISSU Software Installation, page 6-15](#)
- [Loading New Cisco IOS XE Software on the Standby Supervisor Engine, page 6-18](#) (required)
- [Switching to the Standby Supervisor Engine, page 6-21](#) (required)
- [Stopping the ISSU Rollback Timer \(Optional\), page 6-23](#) (optional)
- [Loading New Cisco IOS XE Software on the New Standby Supervisor Engine, page 6-24](#) (required)
- [Using changeversion to Automate an ISSU Upgrade, page 6-25](#)
- [Aborting a Software Upgrade During ISSU, page 6-31](#)
- [Configuring the Rollback Timer to Safeguard Against Upgrade Issues, page 6-32](#)
- [Displaying ISSU Compatibility Matrix Information, page 6-34](#)

## Verifying the ISSU Software Installation

During the ISSU process, there are five valid states: disabled, init, load version, run version, and system reset. Use the **show issu state** command to obtain the current ISSU state:

- Disabled state—The state for the standby supervisor engine while this engine is resetting.
- Init state—The initial state for two supervisor engines, one active and one standby, before the ISSU process is started. It is also the final state after the ISSU process completes.
- Load version (LV) state—The standby supervisor engine is loaded with the new version of Cisco IOS XE software.
- Run version (RV) state—The **issu runversion** command forces the switchover of the supervisor engines. The newly active supervisor engine runs the new Cisco IOS XE software image.
- While running ISSU, if both supervisor engines are reset due to power outage, for example, the ISSU context is lost and returns to the init state. Both supervisor engines return to the old software.

You can verify the ISSU software upgrade by entering **show** commands to provide information on the state of the during the ISSU process:

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>show issu state [detail]</b>	Displays the state of the during the ISSU process.
Step 3	Switch# <b>show redundancy</b>	Displays current or historical status, mode, and related redundancy information about the device.

This example shows how to display the state and the current status of the supervisor engine during the ISSU process:

```
Switch> enable
Switch# show issu state
Switch# show redundancy
```

## Verifying Redundancy Mode Before Beginning the ISSU Process

Before you begin the ISSU process, verify the redundancy mode for the system and be sure to configure NSF and SSO.

The following example displays verification that the system is in SSO mode, that slot 3 is the active supervisor engine, and that slot 4 is the standby supervisor engine. Both supervisor engines are running the same Cisco IOS XE software release.

```
Switch# show redundancy states
  my state = 13 -ACTIVE
  peer state = 8 -STANDBY HOT
  Mode = Duplex
  Unit = Primary
  Unit ID = 5

Redundancy Mode (Operational) = Stateful Switchover
Redundancy Mode (Configured) = Stateful Switchover
  Redundancy State = Stateful Switchover
  Manual Swact = enabled

Communications = Up

  client count = 64
  client_notification_TMR = 240000 milliseconds
  keep_alive TMR = 9000 milliseconds
  keep_alive count = 1
  keep_alive threshold = 18
  RF debug mask = 0

Switch# show redundancy
Redundant System Information :

-----
  Available system uptime = 12 minutes
Switchovers system experienced = 0
  Standby failures = 0
  Last switchover reason = none
```



```

                Hardware Mode = Duplex
Configured Redundancy Mode = Stateful Switchover
                Operating Redundancy Mode = Stateful Switchover
                Maintenance Mode = Disabled
                Communications = Up

Current Processor Information :
-----
                Active Location = slot 5
                Current Software state = ACTIVE
                Uptime in current state = 9 minutes
                Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
                Configuration register = 0x2920

Peer Processor Information :
-----
                Standby Location = slot 6
                Current Software state = STANDBY HOT
                Uptime in current state = 2 minutes
                Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
                Configuration register = 0x2920

```

## Verifying the ISSU State Before Beginning the ISSU Process

Ensure that the active and standby supervisor engines are up and in ISSU Init state and that both supervisor engines are running the same current image.

The following example displays the ISSU state before the process begins:

```

Switch# show issu state detail
                Slot = 5
                RP State = Active
                ISSU State = Init
                Operating Mode = Stateful Switchover
                Current Image = bootflash:old_image
                Pre-ISSU (Original) Image = N/A
                Post-ISSU (Targeted) Image = N/A

                Slot = 6
                RP State = Standby
                ISSU State = Init
                Operating Mode = Stateful Switchover
                Current Image = bootflash:old_image
                Pre-ISSU (Original) Image = N/A
                Post-ISSU (Targeted) Image = N/A

```

The new version of the Cisco IOS XE software must be present on both of the supervisor engines. The directory information displayed for each of the supervisor engines shows that the new version is present.

```

Switch# dir bootflash:
Directory of bootflash:/

 5  -rw-   129923402   Aug 6 2010 09:32:33 +00:00  old_image
 6  -rw-   127724184   Aug 6 2010 09:34:07 +00:00  new_image

```

```

61341696 bytes total (1111388 bytes free)

Switch# dir slavebootflash:
Directory of slavebootflash:/

   4  -rw-   129923402   Aug 6 2006 09:40:10 +00:00  old_image
   5  -rw-   127724184   Aug 6 2006 09:42:13 +00:00  new_image

61341696 bytes total (1116224 bytes free)

```

## Loading New Cisco IOS XE Software on the Standby Supervisor Engine

This task describes how to use ISSU to load a new version of Cisco IOS XE software to the standby supervisor engine.

### Prerequisites

- Ensure that the new version of Cisco IOS XE software image is already present in the file system of both the active and standby supervisor engines. Also ensure that appropriate boot parameters (BOOT string and config-register) are set for the active and standby supervisor engines.




---

**Note** The switch must boot with the BOOT string setting before the ISSU procedure is attempted.

---




---

**Note** **auto-boot** must be enabled for ISSU to succeed.

---

- Optionally, perform additional tests and commands to determine the current state of peers and interfaces for later comparison.
- Ensure the system (both active and standby supervisor engines) is in SSO redundancy mode. If the system is in RPR mode, you can still upgrade the system using the ISSU CLI commands, but the system will experience extended packet loss during the upgrade.'

Refer to the *Stateful Switchover* document for more details on how to configure SSO mode on supervisor engines.

- For ISSU to function, the IOS XE file names on the active and standby supervisor engines must match.

Perform the following steps at the active supervisor engine:

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>issu loadversion</b> [ <i>active-slot</i> ] <i>active-image-new</i> [ <i>standby-slot</i> ] <i>standby-image-new</i> [ <b>forced</b> ]	Starts the ISSU process and (optionally) overrides the automatic rollback when the new Cisco IOS XE software version is detected to be incompatible.  It may take several minutes after entering the <b>issu loadversion</b> command for Cisco IOS XE software to load onto the standby supervisor engine and for the standby supervisor engine to transition to SSO mode. This causes the standby supervisor engine to reload with the new software image.  If you use the <b>forced</b> option, the standby supervisor engine is booted with the new software image. After the software image is loaded on the standby supervisor engine, if the software image is incompatible, the system is forced to the RPR mode. Otherwise the system will continue in the SSO mode.
Step 3	Switch# <b>show issu state</b> [ <i>detail</i> ]	Displays the state of ISSU during the ISSU process. At this point in the ISSU process, use this command to check that the standby supervisor engine is loaded and is in SSO mode.  It may take several minutes after entering the <b>issu loadversion</b> command for Cisco IOS XE software to load onto the standby supervisor engine and the standby supervisor engine to transition to SSO mode. If you enter the <b>show issu state</b> command too quickly, you may not see the information you need.
Step 4	Switch# <b>show redundancy</b> [ <i>states</i> ]	Displays redundancy facility state information.

This example shows how to start the ISSU process, boot the standby supervisor engine in the Standby Hot state, and load the standby supervisor engine slot (6) with the new IOS XE software image:

```
Switch> enable
Switch# issu loadversion 5 bootflash:new_image 6 slavebootflash:new_image
%issu loadversion executed successfully, Standby is being reloaded
Switch# show issu state detail
          Slot = 5
          RP State = Active
          ISSU State = Load Version
          Operating Mode = Stateful Switchover
          Current Image = bootflash:old_image
          Pre-ISSU (Original) Image = bootflash:old_image
          Post-ISSU (Targeted) Image = bootflash:new_image

          Slot = 6
          RP State = Standby
          ISSU State = Load Version
          Operating Mode = Stateful Switchover
```

```

Current Image = bootflash:new_image
Pre-ISSU (Original) Image = bootflash:old_image
Post-ISSU (Targeted) Image = bootflash:new_image

Switch# show redundancy states
my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 5

Redundancy Mode (Operational) = Stateful Switchover
Redundancy Mode (Configured) = Stateful Switchover
Redundancy State = Stateful Switchover
Manual Swact = enabled

Communications = Up

client count = 64
client_notification_TMR = 240000 milliseconds
keep_alive TMR = 9000 milliseconds
keep_alive count = 1
keep_alive threshold = 18
RF debug mask = 0

```

The following example shows how the forced option places the system in RPR mode:

```

Switch> enable
Switch# issu loadversion 5 bootflash:new_image 6 slavebootflash:new_image forced
%issu loadversion executed successfully, Standby is being reloaded

Switch# show issu state detail
Slot = 5
RP State = Active
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:old_image
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

Slot = 6
RP State = Standby
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:old_image
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

```

The following example shows the redundancy mode as RPR:

```

Switch# show redundancy states
my state = 13 -ACTIVE
peer state = 4 -STANDBY COLD
Mode = Duplex
Unit = Primary
Unit ID = 3

Redundancy Mode (Operational) = RPR
Redundancy Mode (Configured) = Stateful Switchover
Redundancy State = RPR
Manual Swact = enabled
Communications = Up

```

```

client count = 64
client_notification_TMR = 240000 milliseconds
  keep_alive TMR = 9000 milliseconds
  keep_alive count = 1
keep_alive threshold = 18
  RF debug mask = 0

```

## Switching to the Standby Supervisor Engine

This task describes how to switchover to the standby supervisor engine, which is running the new Cisco IOS XE software image. Perform the following steps at the active supervisor engine.

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>issu runversion</b> [standby-slot] [standby-image-new]	Forces a switchover from the active to the standby supervisor engine and reloads the former active (current standby) supervisor engines with the old IOS XE image. When you enter the <b>issu runversion</b> command, an SSO switchover will be performed, and NSF procedures will be invoked if so configured.
Step 3	Switch# <b>show issu state</b> [detail]	Displays the state of the during the ISSU process. At this point in the ISSU process, use this command to check that a switchover occurs to slot 6.
Step 4	Switch# <b>show redundancy</b> [states]	Displays redundancy facility state information.

This example shows how to cause a switchover to the former standby supervisor engine (slot 6), reset the former active supervisor engine and reload it with the old IOS XE software image so it becomes the standby supervisor engine:

```

Switch> enable
Switch# issu runversion 6 slavebootflash:new_image
%issu runversion initiated successfully

```

A switchover happens at this point. At the new active supervisor engine, do the following after old active supervisor engine comes up as standby.

```

Switch# show issu state detail
      Slot = 6
      RP State = Active
      ISSU State = Run Version
      Operating Mode = Stateful Switchover
      Current Image = bootflash:new_image
      Pre-ISSU (Original) Image = bootflash:old_image
      Post-ISSU (Targeted) Image = bootflash:new_image

      Slot = 5
      RP State = Standby
      ISSU State = Run Version
      Operating Mode = Stateful Switchover
      Current Image = bootflash:old_image
      Pre-ISSU (Original) Image = bootflash:old_image
      Post-ISSU (Targeted) Image = bootflash:new_image

```

**Note**

The new active supervisor engine is now running the new version of software, and the standby supervisor engine is running the old version of software and is in the standby hot state.

```
Switch# show redundancy states
  my state = 13 -ACTIVE
  peer state = 8 -STANDBY HOT
    Mode = Duplex
    Unit = Primary
    Unit ID = 6

Redundancy Mode (Operational) = Stateful Switchover
Redundancy Mode (Configured) = Stateful Switchover
  Redundancy State = Stateful Switchover
  Manual Swact = enabled

Communications = Up

  client count = 64
  client_notification_TMR = 240000 milliseconds
  keep_alive TMR = 9000 milliseconds
  keep_alive count = 0
  keep_alive threshold = 18
  RF debug mask = 0
```

Once Runversion has completed, the new active supervisor engine will be running the new version of software and the previously active supervisor engine will now become the standby supervisor engine. The standby will be reset and reloaded, but it will remain on the previous version of software and come back online in standbyhot status. The following example shows how to verify these conditions:

```
Switch# show redundancy
Redundant System Information :
-----
  Available system uptime = 12 minutes
Switchovers system experienced = 0
  Standby failures = 0
  Last switchover reason = none

  Hardware Mode = Duplex
  Configured Redundancy Mode = Stateful Switchover
  Operating Redundancy Mode = Stateful Switchover
  Maintenance Mode = Disabled
  Communications = Up

Current Processor Information :
-----
  Active Location = slot 6
  Current Software state = ACTIVE
  Uptime in current state = 9 minutes
  Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
  Configuration register = 0x2920

Peer Processor Information :
-----
  Standby Location = slot 5
  Current Software state = STANDBY HOT
  Uptime in current state = 2 minutes
```

```

Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

```

## Stopping the ISSU Rollback Timer (Optional)

This optional task describes how to stop the rollback timer.

If you do not run the following procedure before the rollback timer “timeout,” the system automatically aborts the ISSU process and reverts to the original Cisco IOS XE software version. By default the rollback timer is 45 minutes.

Use the following information to decide what action you should take:

- If you want to retain your switch in the runversion state for an extended period, you need to stop the rollback timer by entering the **acceptversion** command. Then validate the new software and enter the **commitversion** command (as described in the following section).
- If you want to proceed with the commitversion operation within the rollback timer window of 45 minutes (or the configured value), you do not need to use the **acceptversion** command to stop the roll-back timer.



### Note

The **issu acceptversion** command may be optionally executed after the **issu runversion** command.

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>issu acceptversion</b> [ <i>active-slot</i> ] [ <i>active-image-new</i> ]	Halts the rollback timer and ensures the new Cisco IOS XE ISSU process is not automatically aborted during the ISSU process.  Enter the <b>issu acceptversion</b> command within the time period specified by the rollback timer to acknowledge that the supervisor engine has achieved connectivity to the outside world; otherwise, the ISSU process is terminated, when the rollback timer expires, and the system reverts to the previous version of Cisco IOS XE software by switching to the standby supervisor engine.
Step 3	Switch# <b>show issu rollback-timer</b>	Displays the amount of time left before an automatic rollback will occur.

This example displays the Timer before you stop it. In the following example, the “Automatic Rollback Time” information indicates the amount of time remaining before an automatic rollback will occur.

```

Switch> enable
Switch# show issu rollback-timer
    Rollback Process State = 00:31:09 remaining
    Configured Rollback Time = 00:45:00

Switch# issu acceptversion 6 bootflash:new_image

```

```
% Rollback timer stopped. Please issue the commitversion command.
Switch# show issu rollback-timer
    Rollback Process State = Not in progress
    Configured Rollback Time = 00:45:00
```

## Loading New Cisco IOS XE Software on the New Standby Supervisor Engine

This task explains how to load new version of Cisco IOS XE software to the new standby supervisor engine.

Perform the following steps at the active supervisor engine:

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>issu commitversion</b> [standby-slot] [standby-image-new]	Allows the new Cisco IOS XE software image to be loaded into the standby supervisor engine.
Step 3	Switch# <b>show redundancy</b> [states]	Displays redundancy facility state information.
Step 4	Switch# <b>show issu state</b> [detail]	Displays the state of the during the ISSU process. At this point in the ISSU process, use this command to check that the standby supervisor engine is loaded with the new image.

This example shows how to reset and reload the current standby supervisor engine (slot 1) with the new Cisco IOS XE software version. After you enter the **commitversion** command, the standby supervisor engine boots in the Standby Hot state.

```
Switch> enable
Switch# issu commitversion 5 slavebootflash:new_image
%issu commitversion executed successfully
```

Use the following to verify that the standby supervisor engine is reloaded with the new software.

```
Switch# show redundancy states
    my state = 13 -ACTIVE
    peer state = 8 -STANDBY HOT
        Mode = Duplex
        Unit = Primary
        Unit ID = 6

    Redundancy Mode (Operational) = Stateful Switchover
    Redundancy Mode (Configured) = Stateful Switchover
    Redundancy State = Stateful Switchover
    Manual Swact = enabled

    Communications = Up

    client count = 64
    client_notification_TMR = 240000 milliseconds
    keep_alive TMR = 9000 milliseconds
    keep_alive count = 1
    keep_alive threshold = 18
    RF debug mask = 0

Switch# show redundancy
Redundant System Information :
```



```

-----
    Available system uptime = 12 minutes
Switchovers system experienced = 0
    Standby failures = 0
    Last switchover reason = none

    Hardware Mode = Duplex
Configured Redundancy Mode = Stateful Switchover
    Operating Redundancy Mode = Stateful Switchover
    Maintenance Mode = Disabled
    Communications = Up

Current Processor Information :
-----
    Active Location = slot 4
    Current Software state = ACTIVE
    Uptime in current state = 9 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

Peer Processor Information :
-----
    Standby Location = slot 3
    Current Software state = STANDBY HOT
    Uptime in current state = 2 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

Switch# show issu state detail
    Slot = 6
    RP State = Active
    ISSU State = Init
    Operating Mode = Stateful Switchover
    Current Image = bootflash:new_image
    Pre-ISSU (Original) Image = N/A
    Post-ISSU (Targeted) Image = N/A

    Slot = 5
    RP State = Standby
    ISSU State = Init
    Operating Mode = Stateful Switchover
    Current Image = bootflash:new_image
    Pre-ISSU (Original) Image = N/A
    Post-ISSU (Targeted) Image = N/A

```

The ISSU process has completed. At this stage, any further Cisco IOS XE software version upgrade or downgrade will require that a new ISSU process be invoked.

## Using changeversion to Automate an ISSU Upgrade

This task describes how to use the **issu changeversion** command to perform a one step ISSU upgrade.

## Prerequisites

- Ensure that the new version of Cisco IOS XE software image is already present in the file system of both the active and standby supervisor engines. Also ensure that appropriate boot parameters (BOOT string and config-register) are set for the active and standby supervisor engines
- Optionally, perform additional tests and commands to determine the current state of peers and interfaces for later comparison.
- Ensure the system (both active and standby supervisor engines) is in SSO redundancy mode. If the system is in RPR mode, you can still upgrade the system using the ISSU CLI commands, but the system will experience extended packet loss during the upgrade.'

Refer to the Stateful Switchover document for more details on how to configure SSO mode on supervisor engines (refer to [Chapter 5, “Configuring Supervisor Engine Redundancy Using RPR and SSO”](#)).

- For ISSU to function, the IOS XE software image file names on the active and standby supervisor engines must match.

Perform the following steps at the active supervisor engine:

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>issu changeversion</b> [ <i>active-slot active-image-new</i> ] [ <i>standby-slot standby-image-new</i> ] [ <b>at</b> <i>hh:mm</i>   <b>in</b> <i>hh:mm</i> ] [ <b>quick</b> ]	Initiates a single-step complete upgrade process cycle. Performs the logic of the four standard commands (issu loadversion, issu runversion, issu acceptversion, and issu commitversion) without user intervention.  <i>active-slot</i> —Defines the active slot number.  <i>new-image</i> —Specifies IOS XE image URL to be upgraded to.  <i>standby-slot</i> —Defines the standby slot number.  <i>standby-image</i> —Specifies the standby IOS XE image URL.  <b>at</b> <i>hh:mm</i> —Schedules an ISSU upgrade to begin in the future. Provides an exact time ( <i>hh:mm</i> , 24 hour format) in the next 24 hours when the upgrade will occur.  <b>in</b> <i>hh:mm</i> —Schedules an ISSU upgrade to begin in the future. Provides the number of hours and minutes ( <i>hh:mm</i> format) that will elapse before an upgrade will occur (99:59 max).  <b>quick</b> —Upon switchover, boots the standby supervisor engine with the new, rather than old, image for faster upgrade.
Step 3	Switch# <b>show issu state</b> [ <b>detail</b> ]	Displays the state of the during the ISSU process. At this point in the ISSU process, use this command to check that the standby supervisor engine is loaded and is in SSO mode.
Step 4	Switch# <b>show redundancy</b> [ <b>states</b> ]	Displays redundancy facility state information.

This example shows how to initiate an ISSU upgrade process using the `issu changeversion` command on slot number 5, the slot for the current active supervisor engine. The `show issu state detail` and `show redundancy` command output is included to show the supervisor state before and after the upgrade procedure.

**Note**

The success messages included in the output below is displayed after some delay because the ISSU upgrade procedure progresses through the ISSU states.

```
Switch> enable
Switch# show issu state detail
                Slot = 5
                RP State = Active
                ISSU State = Init
                Operating Mode = Stateful Switchover
                Current Image = bootflash:x.bin
                Pre-ISSU (Original) Image = N/A
                Post-ISSU (Targeted) Image = N/A

                Slot = 6
                RP State = Standby
                ISSU State = Init
                Operating Mode = Stateful Switchover
                Current Image = bootflash:x.bin
                Pre-ISSU (Original) Image = N/A
                Post-ISSU (Targeted) Image = N/A

Switch# show redundancy
Redundant System Information :
-----
    Available system uptime = 12 minutes
    Switchovers system experienced = 0
    Standby failures = 0
    Last switchover reason = none

    Hardware Mode = Duplex
    Configured Redundancy Mode = Stateful Switchover
    Operating Redundancy Mode = Stateful Switchover
    Maintenance Mode = Disabled
    Communications = Up

Current Processor Information :
-----
    Active Location = slot 5
    Current Software state = ACTIVE
    Uptime in current state = 9 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

Peer Processor Information :
-----
    Standby Location = slot 6
    Current Software state = STANDBY HOT
    Uptime in current state = 2 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
```

```
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920
```

```
Switch# issu changeversion bootflash:y.bin
% 'issu changeversion' is now executing 'issu loadversion'
% issu loadversion executed successfully, Standby is being reloaded

% changeversion finished executing loadversion, waiting for standby to reload and reach
SSO ...
```




---

**Note** Standby reloads with target image.

---

```
.....
.....
```

```
*Feb 25 20:41:00.479: %INSTALLER-7-ISSU_OP_SUCC:  issu changeversion is now executing
'issu runversion'
*Feb 25 20:41:03.639: %INSTALLER-7-ISSU_OP_SUCC:  issu changeversion successfully executed
'issu runversion'
```




---

**Note** Switchover occurs.

---

```
.....
.....
```

Look at the console of new active supervisor engine.

```
*Feb 25 20:47:39.859: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
*Feb 25 20:47:39.971: %INSTALLER-7-ISSU_OP_SUCC:  issu changeversion is now executing
'issu commitversion'
```

```
....
....
```




---

**Note** The new standby supervisor engine reloads with target image; changeversion is successful upon SSO terminal state is reached.

---

```
*Feb 25 20:54:16.092: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Feb 25 20:54:16.094: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
Switch#
```

```
Switch# show issu state detail
```

```

                Slot = 6
                RP State = Active
                ISSU State = Init
                Operating Mode = Stateful Switchover
                Current Image = bootflash:y.bin
                Pre-ISSU (Original) Image = N/A
                Post-ISSU (Targeted) Image = N/A
```

```

                Slot = 5
                RP State = Standby
                ISSU State = Init
                Operating Mode = Stateful Switchover
                Current Image = bootflash:y.bin
                Pre-ISSU (Original) Image = N/A
                Post-ISSU (Targeted) Image = N/A
```

```

Switch# show redundancy
Redundant System Information :
-----
    Available system uptime = 12 minutes
Switchovers system experienced = 0
    Standby failures = 0
    Last switchover reason = none

    Hardware Mode = Duplex
    Configured Redundancy Mode = Stateful Switchover
    Operating Redundancy Mode = Stateful Switchover
    Maintenance Mode = Disabled
    Communications = Up

Current Processor Information :
-----
    Active Location = slot 6
    Current Software state = ACTIVE
    Uptime in current state = 9 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

Peer Processor Information :
-----
    Standby Location = slot 5
    Current Software state = STANDBY HOT
    Uptime in current state = 2 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

```

This example shows how to use `issu changeversion` with the `at` command option to schedule an ISSU upgrade procedure to automatically start at the specified time. This example specifies that the ISSU upgrade should be started at 16:30 (24 hour format). The `show issu state detail` and `show redundancy` command output is included to show the supervisor state before and after the `issu changeversion` command was entered.

```

Switch> enable
Switch# show issu state detail

    Slot = 5
    RP State = Active
    ISSU State = Init
    Operating Mode = Stateful Switchover
    Current Image = bootflash:x.bin
    Pre-ISSU (Original) Image = N/A
    Post-ISSU (Targeted) Image = N/A

    Slot = 6
    RP State = Standby
    ISSU State = Init
    Operating Mode = Stateful Switchover
    Current Image = bootflash:x.bin
    Pre-ISSU (Original) Image = N/A
    Post-ISSU (Targeted) Image = N/A

```

```

Switch# show redundancy
Redundant System Information :
-----
    Available system uptime = 12 minutes
Switchovers system experienced = 0
    Standby failures = 0
    Last switchover reason = none

    Hardware Mode = Duplex
    Configured Redundancy Mode = Stateful Switchover
    Operating Redundancy Mode = Stateful Switchover
    Maintenance Mode = Disabled
    Communications = Up

Current Processor Information :
-----
    Active Location = slot 5
    Current Software state = ACTIVE
    Uptime in current state = 9 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

Peer Processor Information :
-----
    Standby Location = slot 6
    Current Software state = STANDBY HOT
    Uptime in current state = 2 minutes
    Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
    Configuration register = 0x2920

Switch# issu changeversion 5 bootflash:y.bin 6 slavebootflash:y at 16:30
% 'issu changeversion' was executed at [ Apr 12 16:27:43 ].
% The planned ISSU changeversion is to occur in (hh:mm:ss) [ 00:03:00 ] at [ Apr 12
16:30:43 ].
% Current system time: [ Apr 12 16:27:43 ]
% Planned upgrade image: bootflash:y.bin
% To cancel the planned upgrade, please execute 'issu abortversion'

Switch# show issu state detail
    Slot = 5
    RP State = Active
    ISSU State = Init
    Changeversion = TRUE
    Operating Mode = Stateful Switchover
    Current Image = bootflash:x.bin
    Pre-ISSU (Original) Image = N/A
    Post-ISSU (Targeted) Image = N/A

    Slot = 6
    RP State = Standby
    ISSU State = Init
    Changeversion = TRUE
    Operating Mode = Stateful Switchover
    Current Image = bootflash:x.bin

```

Pre-ISSU (Original) Image = N/A  
 Post-ISSU (Targeted) Image = N/A

## Aborting a Software Upgrade During ISSU

You can abort the ISSU process at any stage manually (prior to entering the **issu commitversion** command) by entering the **issu abortversion** command. The **issu abortversion** command may also be issued after entering the **issu changeversion** command while the automatic ISSU upgrade is still in progress. The ISSU process also aborts on its own if the software detects a failure.



### Note

If you enter the **issu abortversion** command before the standby supervisor engine becomes hot, the traffic might be disrupted.

If you abort the process after you issue the **issu loadversion** command, the standby supervisor engine is reset and reloaded with the original software.

If the process is aborted after you enter either the **issu runversion** or **issu acceptversion** command, then a second switchover is performed to the new standby supervisor engine that is still running the original software version. The supervisor engine that had been running the new software is reset and reloaded with the original software version.



### Note

Ensure that the standby supervisor is fully booted *before* issuing the **abortversion** command on an active supervisor engine.

The following task describes how to abort the ISSU process before you complete the ISSU process with the **issu commitversion** command.

Perform the following task on the active supervisor engine.

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>issu abortversion</b> [ <i>active slot</i> ] [ <i>active-image-new</i> ]	Cancels the ISSU upgrade or downgrade process in progress and restores the switch to its state before the process had started.

This example shows how to abort the ISSU process on slot number 6, the slot for the current active supervisor engine. In this example, the ISSU upgrade process is in the Runversion state when the **issu abortversion** command is entered:

```
Switch> enable
Switch# show issu state detail
                Slot = 6
                RP State = Active
                ISSU State = Run Version
                Operating Mode = Stateful Switchover
                Current Image = bootflash:x.bin
                Pre-ISSU (Original) Image = bootflash:y.bin
                Post-ISSU (Targeted) Image = bootflash:x.bin

                Slot = 5
```

```

RP State = Standby
ISSU State = Run Version
Operating Mode = Stateful Switchover
Current Image = bootflash:y.bin
Pre-ISSU (Original) Image = bootflash:y.bin
Post-ISSU (Targeted) Image = bootflash:x.bin

Switch# issu abortversion 6
% issu abortversion initiated successfully
Switch# show issu state detail

Slot = 5
RP State = Active
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:y.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

Slot = 6
RP State = Standby
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:y.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

Switch#

```

## Configuring the Rollback Timer to Safeguard Against Upgrade Issues

Cisco IOS XE software maintains an ISSU rollback timer, to safeguard against an upgrade that may leave the new active supervisor engine in a state in which communication with the standby supervisor engine is severed.

You may want to configure the rollback timer to fewer than 45 minutes (the default) so that the user need not wait in case the new software is not committed or the connection to the switch was lost while it was in runversion mode. A user may want to configure the rollback timer to more than 45 minutes in order to have enough time to verify the operation of the new Cisco IOS XE software before committing the new software image.



### Note

The valid timer value range is from 0 to 7200 seconds (two hours). A value of 0 seconds disables the rollback timer.

Once you are satisfied that the new image at the active supervisor engine has been successful and you want to remain in the current state, you may indicate acceptance by issuing the **issu acceptversion** command, which stops the rollback timer.

Issuing the **issu commitversion** command at this stage is equal to entering both the **issu acceptversion** and the **issu commitversion** commands. Use the **issu commitversion** command if you do not intend to run in the current state for a period of time and are satisfied with the new software version.



### Note

The rollback timer can be configured only in the ISSU Init state.

This task explains how to configure the rollback timer:



	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>configure terminal</b>	Enters global configuration mode.
Step 3	Switch(config)# <b>issu set rollback-timer value</b>	Configures the rollback timer value, which can range from 0 to 7200.
Step 4	Switch(config)# <b>exit</b>	Returns the user to privileged EXEC mode.
Step 5	Switch# <b>show issu rollback-timer</b>	Displays the current setting of the ISSU rollback timer.

This example shows how to set the rollback timer to 3600 seconds:

```
Switch> enable
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# issu set rollback-timer 3600
% Rollback timer value set to [ 3600 ] seconds

Switch(config)# exit

Switch# show issu rollback-timer
Rollback Process State = Not in progress
Configured Rollback Time = 60:00
```

The Rollback Timer cannot be set in loadversion or runversion state, as the following example illustrates:

```
Switch# show issu state detail
Slot = 5
RP State = Active
ISSU State = Load Version
Operating Mode = Stateful Switchover
Current Image = bootflash:old_image
Pre-ISSU (Original) Image = bootflash:old_image
Post-ISSU (Targeted) Image = bootflash:new_image

Slot = 6
RP State = Standby
ISSU State = Load Version
Operating Mode = Stateful Switchover
Current Image = bootflash:new_image
Pre-ISSU (Original) Image = bootflash:old_image
Post-ISSU (Targeted) Image = bootflash:new_image

Switch# show issu rollback-timer
Rollback Process State = Not in progress
Configured Rollback Time = 60:00

Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# issu set rollback-timer 20
% ISSU state should be [ init ] to set the rollback timer
```

## Displaying ISSU Compatibility Matrix Information

The ISSU compatibility matrix contains information about other IOS XE software releases and the version in question. This compatibility matrix represents the compatibility of the two software versions, one running on the active and the other on the standby supervisor engine, and the matrix allows the system to determine the highest operating mode it can achieve. This information helps the user identify whether to use ISSU.

This task shows how to display information about the ISSU compatibility matrix.

	Command or Action	Purpose
Step 1	Switch> <b>enable</b>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# <b>show issu comp-matrix</b> { <b>negotiated</b>   <b>stored</b>   <b>xml</b> }	Displays information regarding the ISSU compatibility matrix. <ul style="list-style-type: none"> <li>• <b>negotiated</b>—Displays negotiated compatibility matrix information.</li> <li>• <b>stored</b>—Displays negotiated compatibility matrix information.</li> <li>• <b>xml</b>—Displays negotiated compatibility matrix information in XML format.</li> </ul> <b>Note</b> These commands display only the data within IOSd process. Use the show package compatibility to display the information for the whole system.
Step 3	Switch# <b>show package compatibility</b>	Displays information regarding all client compatibility in the system.

This example shows how to display negotiated information regarding the compatibility matrix:

```
Switch> enable
Switch# show issu comp-matrix negotiated

CardType: WS-C4507R-E(182), Uid: 4, Image Ver: 03.00.00.1.68
Image Name: cat4500e-UNIVERSALK9-M

Cid      Eid      Sid      pSid    pUid     Compatibility
=====
2        1        131078  3       3        COMPATIBLE
3        1        131100  5       3        COMPATIBLE
4        1        131123  9       3        COMPATIBLE
.....
.....

Message group summary:
Cid      Eid      GrpId    Sid      pSid    pUid     Nego Result
=====
2        1        1        131078  3       3        Y
3        1        1        131100  5       3        Y
4        1        1        131123  9       3        Y
.....
.....

List of Clients:
```

```

Cid      Client Name                Base/Non-Base
=====
2       ISSU Proto client          Base
3       ISSU RF                    Base
4       ISSU CF client             Base
.....
.....

```

This example shows how to display stored information regarding the compatibility matrix:

```
Switch# show issu comp-matrix stored
```

```
Number of Matrices in Table = 1
```

```

(1) Matrix for cat4500e-ENTSERVICESK9-M(182) - cat4500ex-ENTSERVICESK9-M(182)
=====
Start Flag (0xDEADBABE)

      My Image ver: 03.01.00.SG
      Peer Version  Compatibility
      -----
      03.01.00.SG          Comp(3)

```

```
Switch#
```

Dynamic Image Version Compatibility (DIVC) feature is supported in IOS XE releases. With DIVC, we store Dynamic(0) rather than Incomp(1), Base(2), or Comp(3), and determine compatibility during run-time when two different DIVC-capable IOS XE software images are running in the active and standby supervisor engines during ISSU.

For Catalyst 4500 switches, a value of Dynamic(0) in the stored compatibility-matrix normally results in Base(2) or Comp(3) upon run-time negotiation between the two IOS XE software images; as of today, you never observe Incomp(1) as long as the other IOS XE name is present in the stored compatibility-matrix.

This example shows how to display negotiated information regarding non-IOSd clients:

```
Switch# show package compatibility
```

```

PackageName      PeerPackageName      ModuleName      Compatibility
-----
rp_base          rp_base              aaa              COMPATIBLE
rp_base          rp_base              aaacommon       COMPATIBLE
rp_base          rp_base              access_policy   COMPATIBLE
rp_base          rp_base              app_sess        COMPATIBLE
rp_base          rp_base              app_sess_ios    COMPATIBLE
rp_base          rp_base              auth_mgr        COMPATIBLE
.....
.....

```

## Cisco High Availability Features in Cisco IOS XE 3.1.0 SG

This section provides a list of High Availability software features that are supported in Cisco IOS XE 3.1.0SG. Links to the feature documentation are included.

Feature guides may contain information about more than one feature. To find information about a specific feature within a feature guide, see the Feature Information table at the end of the guide.

Feature guides document features that are supported on many different software releases and platforms. Your Cisco software release or platform may not support all the features documented in a feature guide. See the Feature Information table at the end of the feature guide for information about which features in

that guide are supported in your software release. Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

### **ISSU (IOS In-Service Software Upgrade)**

[http://www.cisco.com/en/US/products/ps7149/products\\_ios\\_protocol\\_group\\_home.html](http://www.cisco.com/en/US/products/ps7149/products_ios_protocol_group_home.html)

### **Enhanced High System Availability**

[http://www.cisco.com/en/US/solutions/collateral/ns340/ns517/ns431/ns17/net\\_brochure0900aecd8044ddd.pdf](http://www.cisco.com/en/US/solutions/collateral/ns340/ns517/ns431/ns17/net_brochure0900aecd8044ddd.pdf)