CHAPTER 1

Configuring the Cisco IOS In-Service Software Upgrade Process

Note Starting with Cisco IOS 12.2(31)SGA, ISSU is supported on the Catalyst 4500. All line cards are supported.

Operating on redundant systems, the In-Service Software Upgrade (ISSU) process allows Cisco IOS 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 software to be modified 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.

This section includes these topics:

- Prerequisites to Performing ISSU, page 1-2
- About ISSU, page 1-3
- Performing the ISSU Process, page 1-15
- Related Documents, page 1-39

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

Prerequisites to Performing ISSU

Before performing ISSU, you need to meet these prerequisites:

- Image type of the existing and target image must match. For example, you cannot upgrade from an IP Base image to an Enterprise Services image (and vice versa) without experiencing several minutes of traffic loss.

Note A similar limitation 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, NFL daughter card and so on).
- The new and old Cisco IOS software images must be loaded into the file systems (bootflash or compact flash) of both the active and the standby supervisor engines before you begin the ISSU process.

  The old image should be available either in bootflash or compact flash and the system should have been booted from one of these locations because the boot variable should not be changed before the ISSU process unfolds.

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 system is configured for redundancy mode SSO and that the file system for both the active and the standby supervisor engines contains the new ISSU-compatible image. The current Cisco IOS version running in the system must also support ISSU.

  You can enter various commands on the Catalyst 4500 series switch or the ISSU application on Cisco Feature Navigator are to determine supervisor engine versioning and Cisco IOS compatibility.
- 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 1-27.
- If you enter the **no ip routing** command, ISSU falls back from SSO to RPR mode, resulting in traffic loss.

About ISSU

Note Do not make any hardware changes while performing ISSU.

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

Stateful Switchover Overview

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

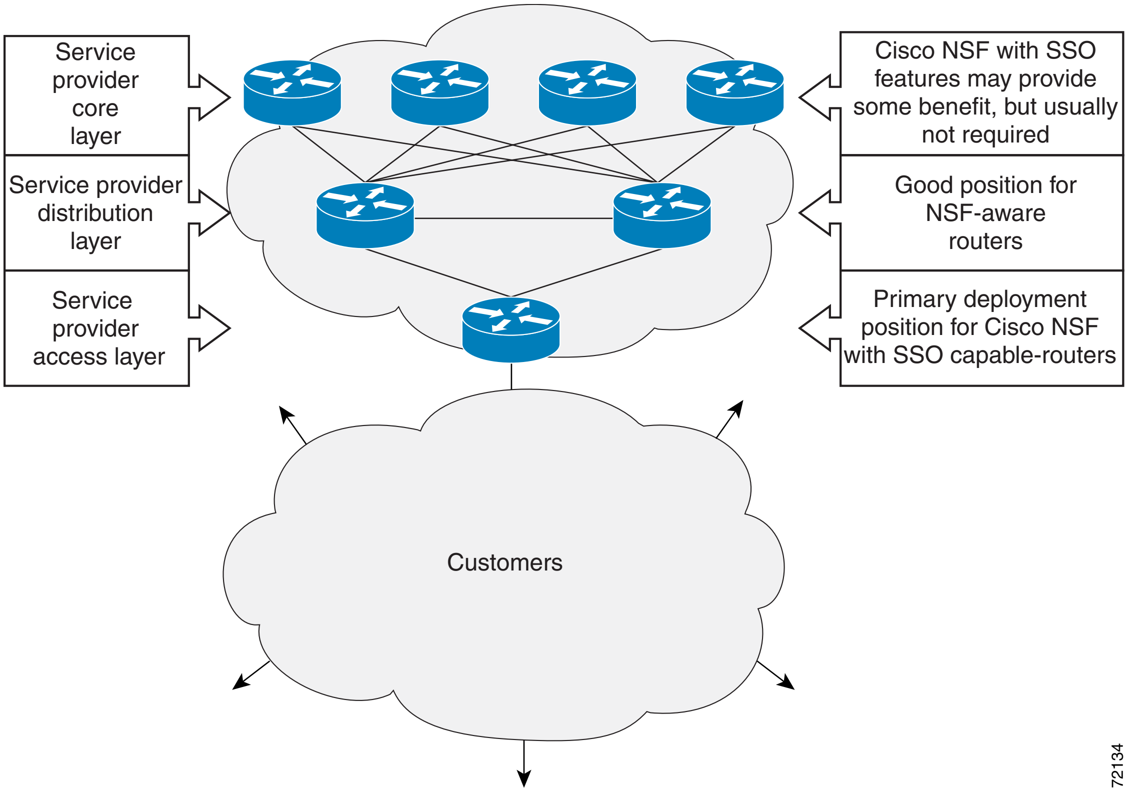
A switchover from the active to the standby processor occurs when the active supervisor engine fails or is removed from the networking device.

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 1-1 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.

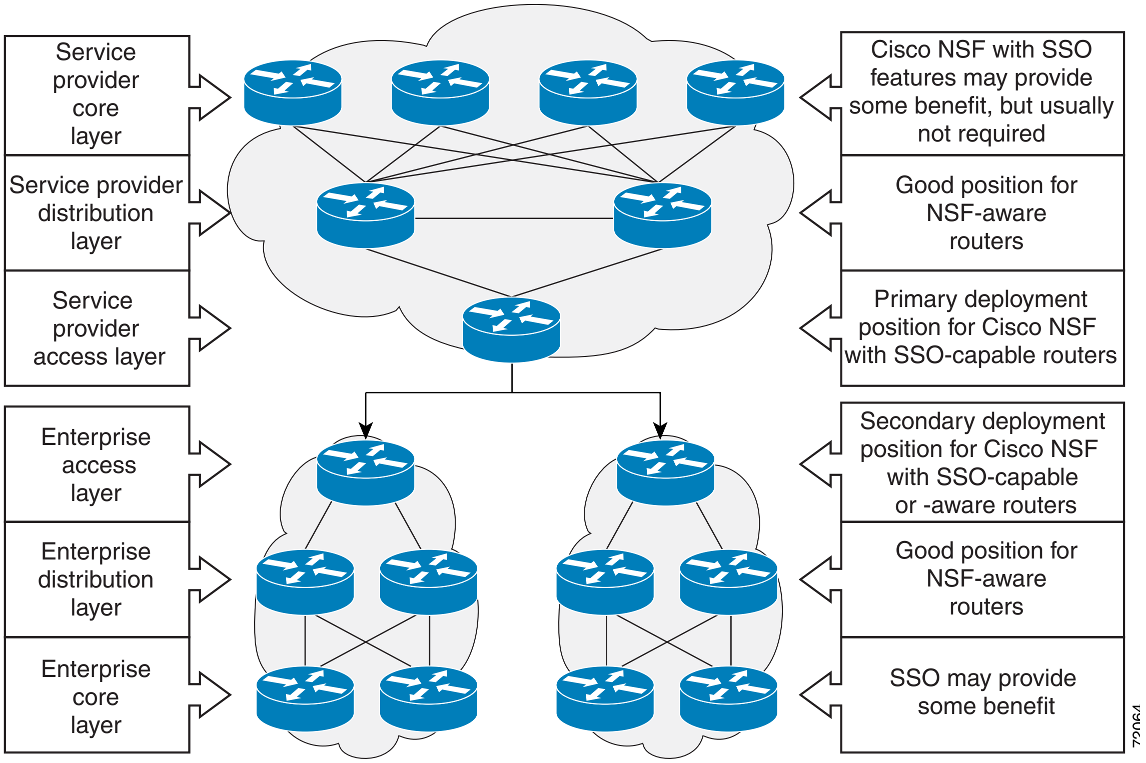
For Cisco NSF protocols that require neighboring devices to participate in Cisco NSF, Cisco NSF-aware software images must be installed on those neighboring distribution layer devices. Depending on your objectives, you may decide to deploy Cisco NSF and SSO features at the core layer of your network.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

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


Service provider core layer
Service provider distribution layer
Service provider access layer
Cisco NSF with SSO features may provide some benefit, but usually not required
Good position for NSF-aware routers
Primary deployment position for Cisco NSF with SSO capable-routers
Customers
72134

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 1-2 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 continue uninterrupted through the network in this example.

Figure 1-2 Cisco NSF with SSO Network Deployment: Enterprise Networks


Service provider core layer
Service provider distribution layer
Service provider access layer
Enterprise access layer
Enterprise distribution layer
Enterprise core layer
Cisco NSF with SSO features may provide some benefit, but usually not required
Good position for NSF-aware routers
Primary deployment position for Cisco NSF with SSO-capable routers
Secondary deployment position for Cisco NSF with SSO-capable or -aware routers
Good position for NSF-aware routers
SSO may provide some benefit
72064

NSF Overview

Cisco NSF works with the SSO feature in Cisco IOS 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.

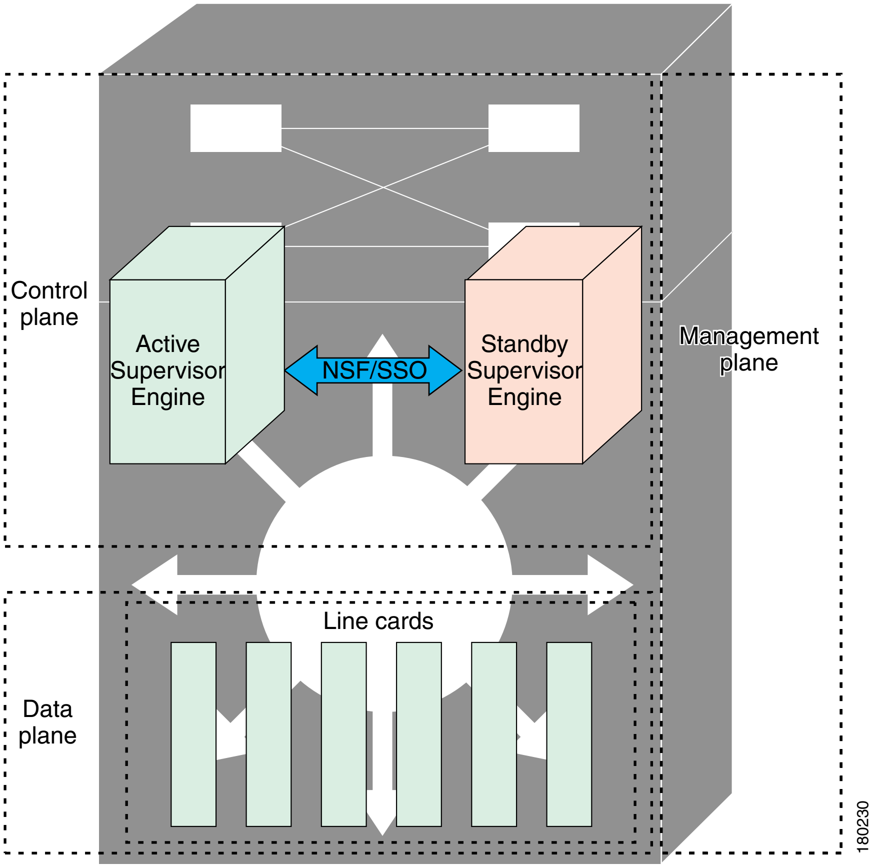
Cisco NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored following a switchover. With Cisco NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded while the standby supervisor engine assumes 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 Overview

The ISSU process allows you to perform a Cisco IOS 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 1-8 on page 1-11.) Cisco IOS ISSU takes advantage of the Cisco IOS high availability infrastructure—Cisco NSF with SSO and hardware redundancy—and eliminates downtime associated with software upgrades or version changes by allowing changes while the system remains in service (see Figure 1-3).

SSO and NSF mode support configuration and runtime state synchronization from the active to the standby supervisor engine. For this process to happen, the images 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 support different sets of features and commands.

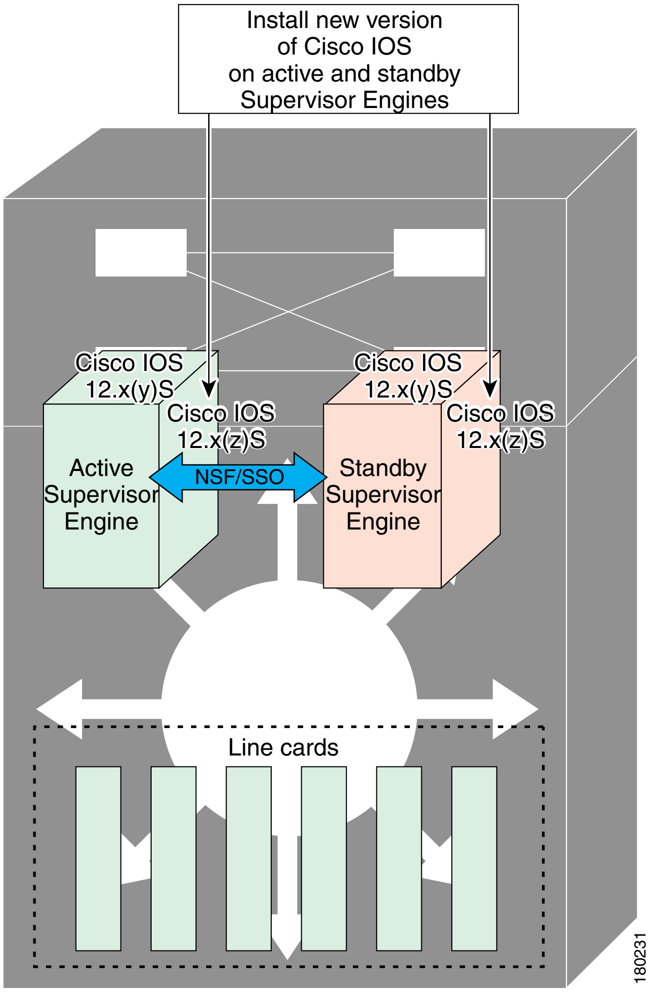
Figure 1-3 High Availability Features and Hardware Redundancy in the ISSU Process



An ISSU-capable switch consists of two supervisor engines (active and standby) and one or more line cards. Before initiating the ISSU process, copy the Cisco IOS software into the file systems of both supervisor engines (see Figure 1-4).

Note In the following figure, Cisco IOS 12.x(y)S represents the *current* version of Cisco IOS.

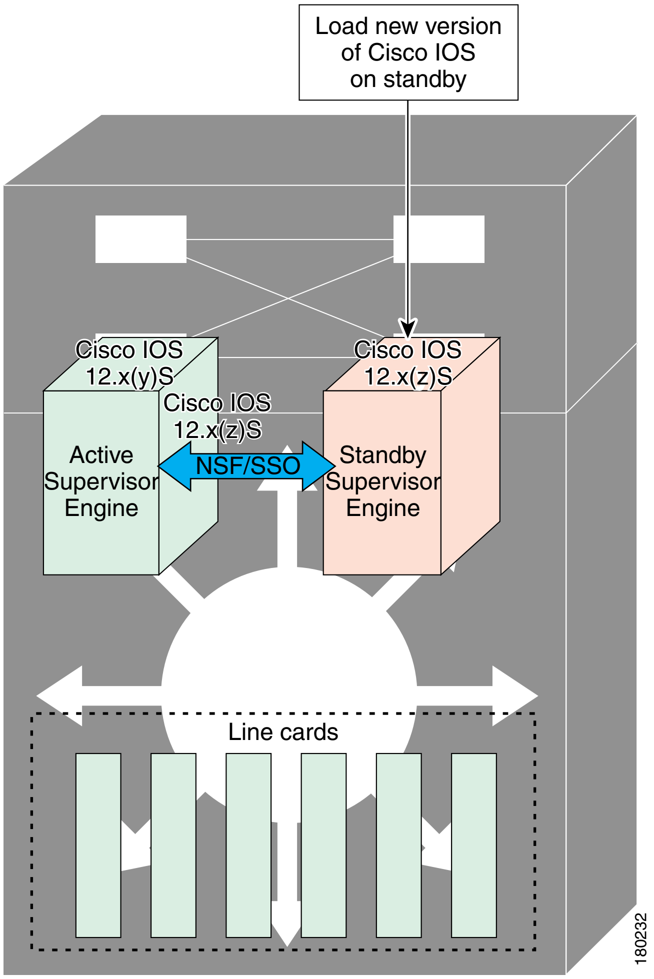
Figure 1-4 Install/Copy New Version of Cisco IOS Software on Both Supervisor Engines



After you have copied the Cisco IOS software to both file systems, load the new version of Cisco IOS software onto the standby supervisor engine (see Figure 1-5).

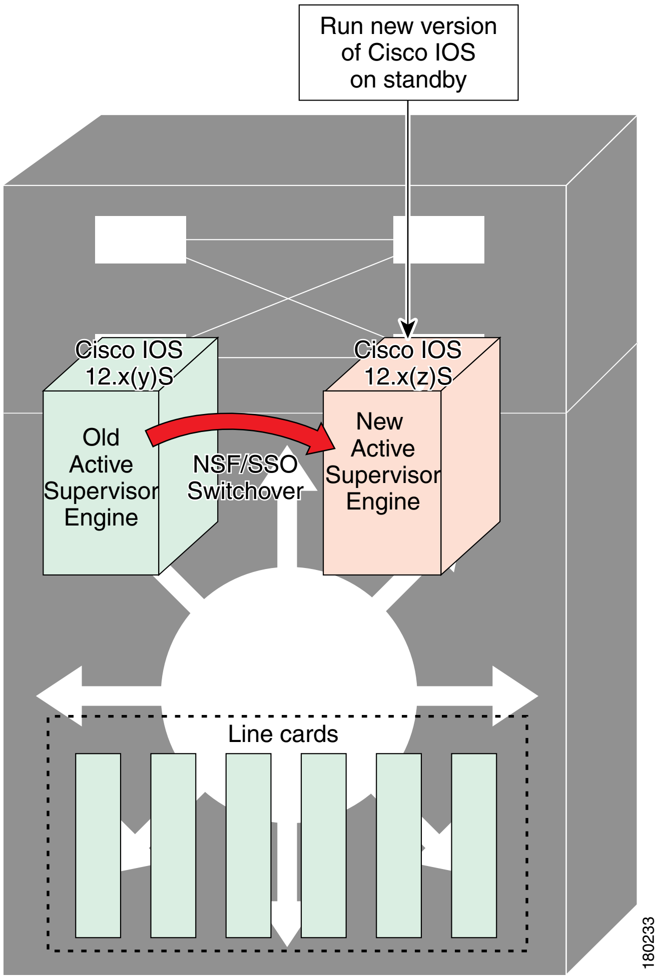
Note Without the ISSU feature, you cannot have SSO or NSF functioning between the active and standby supervisor engines when they are running two different versions of Cisco IOS image.

Figure 1-5 Load New Version of Cisco IOS Software on the Standby Supervisor Engine



After a switchover (NSF or SSO, not RPR), the standby supervisor engine takes over as the new active supervisor engine (see Figure 1-6).

Figure 1-6 Switch Over to Standby Supervisor Engine


Run new version of Cisco IOS on standby
Cisco IOS 12.x(y)S
Cisco IOS 12.x(z)S
Old Active Supervisor Engine
NSF/SSO Switchover
New Active Supervisor Engine
Line cards
180233

The former active supervisor engine is loaded with an old Cisco IOS 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 image. Next, the former active supervisor engine is loaded with the new version of Cisco IOS software and becomes the new standby supervisor engine (see Figure 1-7).

Figure 1-7 Load New Standby Supervisor Engine with New Cisco IOS Software

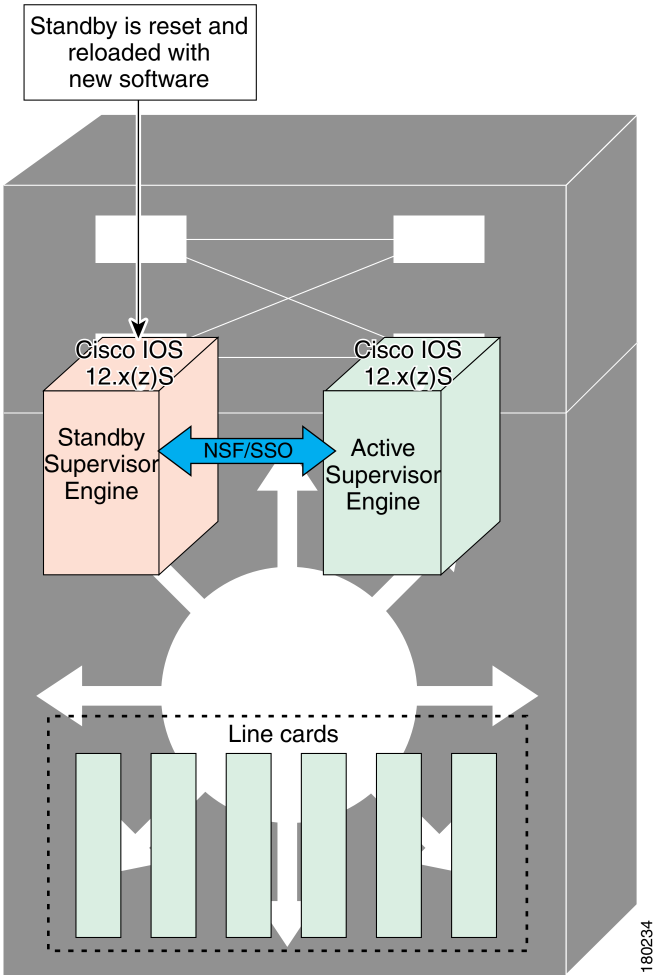
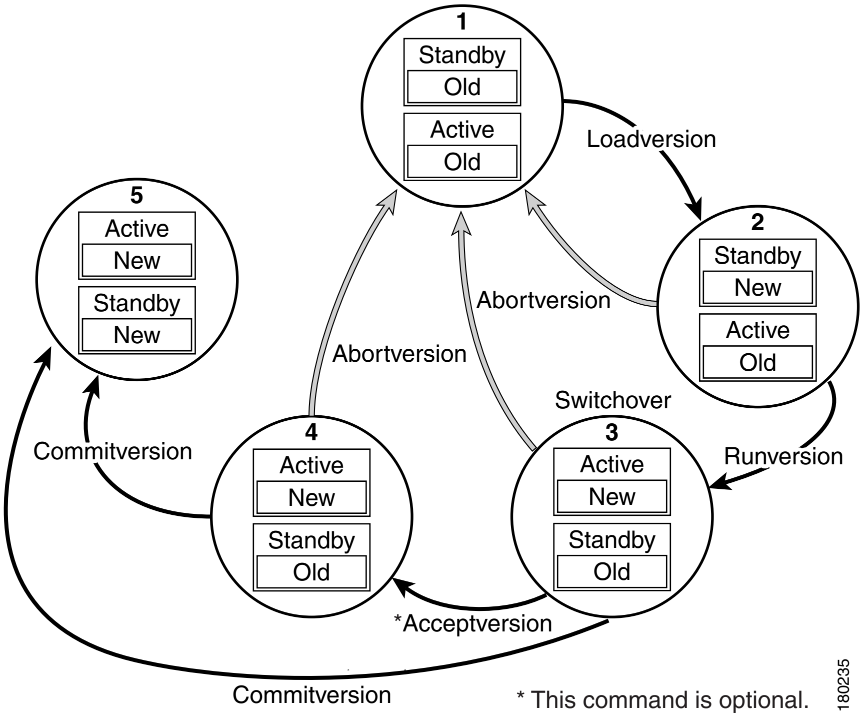


Figure 1-8 shows the steps during the ISSU process.

Figure 1-8 Steps During the ISSU Process



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**, **issue 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.

Note To use the **issu changeversion** command, both old and new IOS versions must support **issu changeversion** functionary.

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 software 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

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 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 software handle, that feature should be disabled prior to initiating the ISSU process.

Versioning Capability in Cisco IOS Software to Support ISSU

Before the introduction of ISSU, the SSO mode of operation required each supervisor engine to be running the same versions of Cisco IOS software.

Note The operating mode of the system in a redundant HA configuration is determined by exchanging version strings when the standby supervisor engine registers with the active supervisor engine.

The system entered SSO mode only if the versions running on the both supervisor engines were the same. If not, the redundancy mode changes to RPR. With ISSU capability, the implementation allows two different but compatible release levels of Cisco IOS images to interoperate in SSO mode and enables software upgrades while packet forwarding continues. Version checking done before ISSU capability was introduced is no longer sufficient to allow the system to determine the operating mode.

ISSU requires additional information to determine compatibility between software versions. A compatibility matrix is defined, containing information about other images relative 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 cannot progress to SSO operational mode.

Compatibility Matrix

You can perform the ISSU process when the Cisco IOS software on both the active and the standby supervisor engine is capable of ISSU and the old and new images 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 succeeds 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 succeeds; however, some subsystems cannot always maintain state during the transition from the old to the new version of Cisco IOS. The matrix entry designates the images to be base-level compatible (B).

  However, you should be able to perform an ISSU upgrade without any functionality loss even if the matrix entry is B. The downgrade may experience some functionality loss if the newer image had additional functionality.
- Incompatible—A core set of system infrastructure exists in Cisco IOS 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 software images 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 period when the versions of Cisco IOS at the active and standby supervisor engines are incompatible.

  If you attempt to perform ISSU with a peer that does not support ISSU, the system automatically uses RPR instead.

The compatibility matrix represents the compatibility relationship a Cisco IOS software image has with all of the other Cisco IOS software versions within the designated support window (for example, all of those software versions the image “knows” about) and is populated and released with every 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 software image and on Cisco.com so that users can determine in advance whether an upgrade can be done using the ISSU process.

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 purpose*s 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, assuming that both supervisor engines are running the same version of Cisco IOS software. This assumption is not valid for ISSU.

With ISSU, an SNMP client can handle transformations for the MIBs across two different versions of Cisco IOS, if needed. An SNMP client handles transformation for all MIBs and handles the transmit and receive functionality across the active and standby supervisor engines. During SNMP, a MIB is completely synchronized from the active supervisor engine to the standby supervisor engine only if the versions of the MIB on both Cisco IOS releases are the same.

Compatibility Verification Using Cisco Feature Navigator

The ISSU application on Cisco Feature Navigator allows you to:

- Select an ISSU-capable image
- Identify which images are compatible with that image
- Compare two images and understand the compatibility level of the images (that is, compatible, base-level compatible, and incompatible)
- Compare two images and see the client compatibility for each ISSU client
- Provide links to release notes for the image

Performing 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 a new or modified Cisco IOS software, and have a minimal impact to traffic.

Note For an illustration of the process flow for ISSU, refer to Figure 1-8 on page 1-11.

This section includes the following topics:

- Verifying the ISSU Software Installation, page 1-16
- Loading New Cisco IOS Software on the Standby Supervisor Engine, page 1-18 (required)
- Switching to the Standby Supervisor Engine, page 1-21 (required)
- Stopping the ISSU Rollback Timer (Optional), page 1-23 (optional)
- Loading New Cisco IOS Software on the New Standby Supervisor Engine, page 1-24
- Aborting a Software Upgrade During ISSU, page 1-31
- Configuring the Rollback Timer to Safeguard Against Upgrade Issues, page 1-32
- Displaying ISSU Compatibility Matrix Information, page 1-34

Verifying the ISSU Software Installation

During the ISSU process, five valid states exist: 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 is 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 software.
- Run version (RV) state—The **issu runversion** command forces the switchover of the supervisor engines. The newly active supervisor engine now runs the new Cisco IOS software image.
- System reset (SR) state—This state occurs either when you enter the **issu abortversion** command before the Init state is reached, or if the rollback timer expires before you execute the **issu acceptversion** command.

You can verify the ISSU software installation by entering **show** commands, as follows:

	Command or Action	Purpose
Step 1	`Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch# show issu state [detail]`	Displays the state of the during the ISSU process.
Step 3	`Switch# show redundancy`	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 1 is the active supervisor engine, and that slot 2 is the standby supervisor engine. Both supervisor engines are running the same Cisco IOS software image.

```
Switch# show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
        Unit ID = 1


Redundancy Mode (Operational) = Stateful Switchover
Redundancy Mode (Configured)  = Stateful Switchover
Redundancy State              = Stateful Switchover
```

```
Maintenance Mode = Disabled
    Manual Swact = enabled
  Communications = Up

   client count = 39
 client_notification_TMR = 240000 milliseconds
          keep_alive TMR = 9000 milliseconds
        keep_alive count = 0
    keep_alive threshold = 18
           RF debug mask = 0x0


Switch# show redundancy
Redundant System Information :
------------------------------
       Available system uptime = 1 minute
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 1
        Current Software state = ACTIVE
       Uptime in current state = 0 minutes
                 Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
                          BOOT = bootflash:old_image,1;
        Configuration register = 0x822

Peer Processor Information :
----------------------------
              Standby Location = slot 2
        Current Software state = STANDBY HOT
       Uptime in current state = 1 minute
                 Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
                          BOOT = bootflash:old_image,1;
        Configuration register = 0x822
```

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 the boot variables are set and pointing to valid files.

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

```
Switch# show issu state detail
                          Slot = 1
                      RP State = Active
```

```
                    ISSU State = Init
                 Boot Variable = bootflash:old_image,1;
                Operating Mode = Stateful Switchover
               Primary Version = N/A
             Secondary Version = N/A
               Current Version = bootflash:old_image

                          Slot = 2
                      RP State = Standby
                    ISSU State = Init
                 Boot Variable = bootflash:old_image,1;
                Operating Mode = Stateful Switchover
               Primary Version = N/A
             Secondary Version = N/A
               Current Version = bootflash:old_image
```

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

```
Switch# dir bootflash:
Directory of bootflash:/

    5  -rwx    13636500   Sep 6 2006 09:32:33 +00:00  old_image
    6  -rwx    13636500   Sep 6 2006 09:34:07 +00:00  new_image

61341696 bytes total (1111388 bytes free)

Switch# dir slavebootflash:
Directory of slavebootflash:/

    4  -rwx    13636500   Sep 6 2006 09:40:10 +00:00  old_image
    5  -rwx    13636500   Sep 6 2006 09:42:13 +00:00  new_image

61341696 bytes total (1116224 bytes free)
```

Loading New Cisco IOS Software on the Standby Supervisor Engine

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

Prerequisites

- Ensure that the new version of Cisco IOS 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 standby supervisor engine.

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 rather than SSO mode, you can still upgrade the system using the ISSU CLI commands, but the system experiences 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 image names on the active and standby supervisor engines must match.

Perform this task at the active supervisor engine:

	Command or Action	Purpose
Step 1	`Switch> `**`enable`**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch# `**`issu loadversion`** *`active-slot active-image-new standby-slot standby-image-new`* `[`**`forced`**`]`	Starts the ISSU process and (optionally) overrides the automatic rollback when the new Cisco IOS software version is detected to be incompatible. It may take several seconds after the **issu loadversion** command is entered for Cisco IOS 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 image. If you use the **forced** option, the standby supervisor engine is booted with the new image. After the image is loaded on the standby supervisor engine, if the image is incompatible, the system is forced to the RPR mode. Otherwise the system continues in the SSO mode.
Step 3	`Switch# `**`show issu state`** `[`**`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 and is in SSO mode. It may take several seconds after entering the **issu loadversion** command for Cisco IOS software to load onto the standby supervisor engine and the standby supervisor engine to transition to SSO mode. If you enter the **show issu state** command too quickly, you may not see the information you need.
Step 4	`Switch# `**`show redundancy`** `[`**`states`**`]`	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 (2) with the new image:

```
Switch> enable
Switch# issu loadversion 1 bootflash:new_image 2 slavebootflash:new_image
Switch# show issu state detail
                          Slot = 1
                      RP State = Active
                    ISSU State = Load Version
                 Boot Variable = bootflash:old_image,12
                Operating Mode = Stateful Switchover
               Primary Version = bootflash:old_image
             Secondary Version = bootflash:new_image
               Current Version = bootflash:old_image
```

```
                          Slot = 2
                      RP State = Standby
                    ISSU State = Load Version
                 Boot Variable = bootflash:new_image,12;bootflash:old_image,12
                Operating Mode = Stateful Switchover
               Primary Version = bootflash:old_image
             Secondary Version = bootflash:new_image
               Current Version = bootflash:new_image



Switch# show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
        Unit ID = 1


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

   client count = 39
 client_notification_TMR = 240000 milliseconds
          keep_alive TMR = 9000 milliseconds
        keep_alive count = 1
    keep_alive threshold = 18
           RF debug mask = 0x0
```

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

```
Switch> enable
Switch# issu loadversion 1 bootflash:new_image 2 slavebootflash:new_image forced
Switch# show issu state detail
                          Slot = 1
                      RP State = Active
                    ISSU State = Load Version
                 Boot Variable = bootflash:old_image,12
                Operating Mode = RPR
               Primary Version = bootflash:old_image
             Secondary Version = bootflash:new_image
               Current Version = bootflash:old_image

                          Slot = 2
                      RP State = Standby
                    ISSU State = Load Version
                 Boot Variable = bootflash:new_image,12;bootflash:old_image,12
                Operating Mode = RPR
               Primary Version = bootflash:old_image
             Secondary Version = bootflash:new_image
               Current Version = bootflash:new_image
```

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 = 1


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

   client count = 39
 client_notification_TMR = 240000 milliseconds
          keep_alive TMR = 9000 milliseconds
        keep_alive count = 1
    keep_alive threshold = 18
           RF debug mask = 0x0
```

Switching to the Standby Supervisor Engine

This task describes how to switchover to the standby supervisor engine, which is running the new Cisco IOS software image.

Perform this task at the active supervisor engine:

	Command or Action	Purpose
Step 1	`Switch>` **`enable`**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch#` **`issu runversion`** *`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 image. When you enter the **issu runversion** command, an SSO switchover is performed, and NSF procedures are invoked if configured.
Step 3	`Switch#` **`show issu state`** `[`**`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 2.
Step 4	`Switch#` **`show redundancy`** `[`**`states`**`]`	Displays redundancy facility state information.

This example shows how to cause a switchover to the former standby supervisor engine (slot 2), reset the former active supervisor engine and reload it with the old image so it becomes the standby supervisor engine:

```
Switch> enable
Switch# issu runversion 2 slavebootflash:new_image
This command will reload the Active unit.  Proceed ? [confirm]
```

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

```
Switch# show issu state detail
                          Slot = 2
                      RP State = Active
                    ISSU State = Run Version
                 Boot Variable = bootflash:new_image,12;bootflash:old_image,12
                Operating Mode = Stateful Switchover
               Primary Version = bootflash:new_image
             Secondary Version = bootflash:old_image
               Current Version = bootflash:new_image

                          Slot = 1
                      RP State = Standby
                    ISSU State = Run Version
                 Boot Variable = bootflash:old_image,12
                Operating Mode = Stateful Switchover
               Primary Version = bootflash:new_image
             Secondary Version = bootflash:old_image
               Current Version = bootflash:old_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 = Secondary
        Unit ID = 2


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

   client count = 39
 client_notification_TMR = 240000 milliseconds
          keep_alive TMR = 9000 milliseconds
        keep_alive count = 1
    keep_alive threshold = 18
           RF debug mask = 0x0
```

Once **runversion** command completes, the new active supervisor engine is running the new version of software and the previously active supervisor engine now becomes the standby supervisor engine. The standby is reset and reloaded, but remains 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 = 23 minutes
Switchovers system experienced = 1
              Standby failures = 0
        Last switchover reason = user forced
```

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

Current Processor Information :
-------------------------------
               Active Location = slot 2
        Current Software state = ACTIVE
       Uptime in current state = 11 minutes
                 Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
                          BOOT = bootflash:new_image,12;bootflash:old_image,12
        Configuration register = 0x822

Peer Processor Information :
----------------------------
              Standby Location = slot 1
        Current Software state = STANDBY HOT
       Uptime in current state = 4 minutes
                 Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
                          BOOT = bootflash:old_image,12
        Configuration register = 0x822
```

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 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 this state for an extended period, you need to stop the rollback timer (then validate and run the **acceptversion** command directly).
- If you want to proceed to the following step (running “commitversion”) within the rollback timer window of 45 minutes, you do not need to stop the rollback timer.

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

	Command or Action	Purpose
Step 1	`Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch# issu acceptversion active-slot [active-image-new]`	Halts the rollback timer and ensures the new Cisco IOS ISSU process is not automatically aborted during the ISSU process. Enter the **issu acceptversion** 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, and the system reverts to the previous version of Cisco IOS software by switching to the standby supervisor engine.
Step 3	`Switch# show issu rollback-timer`	Displays the amount of time left before an automatic rollback occurs.

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 occurs.

```
Switch> enable
Switch# show issu rollback-timer
        Rollback Process State = In progress
      Configured Rollback Time = 45:00
       Automatic Rollback Time = 38:30

Switch# issu acceptversion 2 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 = 45:00
```

Loading New Cisco IOS Software on the New Standby Supervisor Engine

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

Perform this task at the active supervisor engine:

	Command or Action	Purpose
Step 1	`Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch# issu commitversion standby-slot-number [standby-image-new]`	Allows the new Cisco IOS software image to be loaded into the standby supervisor engine.
Step 3	`Switch# show redundancy [states]`	Displays redundancy facility state information.
Step 4	`Switch# show issu state [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 2.

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

```
Switch> enable
Switch# issu commitversion 1 slavebootflash:new_image

Wait till standby supervisor is reloaded with the new image. Then apply the following:

Switch# show redundancy states
00:17:12: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Secondary
        Unit ID = 2


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

   client count = 39
 client_notification_TMR = 240000 milliseconds
          keep_alive TMR = 9000 milliseconds
        keep_alive count = 0
    keep_alive threshold = 18
           RF debug mask = 0x0

Switch# show redundancy
Redundant System Information :
------------------------------
       Available system uptime = 41 minutes
Switchovers system experienced = 1
              Standby failures = 1
        Last switchover reason = user forced

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

Current Processor Information :
-------------------------------
               Active Location = slot 2
        Current Software state = ACTIVE
       Uptime in current state = 29 minutes
                 Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
                          BOOT = bootflash:new_image,12;bootflash:old_image,1;
        Configuration register = 0x822

Peer Processor Information :
----------------------------
              Standby Location = slot 1
        Current Software state = STANDBY HOT
       Uptime in current state = 12 minutes
```

```
                 Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
                          BOOT = bootflash:new_image,12;bootflash:old_image,1;
        Configuration register = 0x822

Switch# show issu state detail
                          Slot = 2
                      RP State = Active
                    ISSU State = Init
                 Boot Variable = bootflash:new_image,12;bootflash:old_image,1;
                Operating Mode = Stateful Switchover
               Primary Version = N/A
             Secondary Version = N/A
               Current Version = bootflash:new_image

                          Slot = 1
                      RP State = Standby
                    ISSU State = Init
                 Boot Variable = bootflash:new_image,12;bootflash:old_image,1;
                Operating Mode = Stateful Switchover
               Primary Version = N/A
             Secondary Version = N/A
               Current Version = bootflash:new_image
```

The ISSU process has been completed. At this stage, any further Cisco IOS software version upgrade or downgrade requires 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 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 1, "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> **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	Switch# **issu changeversion** [*active-slot active-image-new*]] [*standby-slot* [*standby-image-new*]] [**at** *hh:mm* | **in** *hh:mm*] [**quick**]	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. *active-slot*—Defines the active slot number. *new-image*—Specifies IOS XE image URL to be upgraded to. *standby-slot*—Defines the standby slot number. *standby-image—Specifies the* standby IOS XE image URL. **at** *hh:mm*—Schedules an ISSU upgrade to begin in the future. Provides an exact time (*hh*:*mm*, 24 hour format) in the next 24 hours when the upgrade will occur. **in** *hh:mm*—Schedules an ISSU upgrade to begin in the future. Provides the number of hours and minutes (*hh*:*mm* format) that will elapse before an upgrade will occur (99:59 max). **quick—**Upon switchover, boots the standby supervisor engine with the new, rather than old, image for faster upgrade.
Step 3	Switch# **show issu state** [**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 and is in SSO mode.
Step 4	Switch# **show redundancy** [**states**]	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 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 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 enter 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 engine is fully booted *before* entering 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>` **`enable`**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch#` **`issu abortversion`** *`active slot [active-image-new]`*	Cancels the ISSU upgrade or downgrade process in progress and restores the router to its state before the process had started.

This example shows how to abort the ISSU process on slot number 2, the slot for the current active supervisor engine:

```
Switch> enable
Switch# issu abortversion 2
```

Configuring the Rollback Timer to Safeguard Against Upgrade Issues

Cisco IOS 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 software before committing the new 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 ISSU process has been successful and you want to remain in the current state, you must indicate acceptance by entering the **issu acceptversion** command, which stops the rollback timer. Entering the **issu acceptversion** command is extremely important in advancing the ISSU process.

Entering 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 now and are satisfied with the new software version.

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

Perform this task to configure the rollback timer:

	Command or Action	Purpose
Step 1	`Switch>` **`enable`**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch#` **`configure terminal`**	Enters global configuration mode.
Step 3	`Switch(config)#` **`issu set rollback-timer`** *`hh::mm::ss`*	Configures the rollback timer value.
Step 4	`Switch(config)#` **`exit`**	Returns the user to privileged EXEC mode.
Step 5	`Switch#` **`show issu rollback-timer`**	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 LV state, as the following example illustrates:

```
Switch# show issu state detail
                          Slot = 1
                      RP State = Active
                    ISSU State = Load Version
                 Boot Variable = bootflash:old_image,12
                Operating Mode = RPR
               Primary Version = bootflash:old_image
             Secondary Version = bootflash:new_image
               Current Version = bootflash:old_image

                          Slot = 2
                      RP State = Standby
                    ISSU State = Load Version
                 Boot Variable = bootflash:new_image,12;bootflash:old_image,12
                Operating Mode = RPR
               Primary Version = bootflash:old_image
             Secondary Version = bootflash:new_image
               Current Version = 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 software images about 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.

Perform this task to display information about the ISSU compatibility matrix:

	Command or Action	Purpose
Step 1	`Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	`Switch# show issu comp-matrix {negotiated | stored | xml}`	Displays information regarding the ISSU compatibility matrix. • **negotiated**—Displays negotiated compatibility matrix information. • **stored**—Displays negotiated compatibility matrix information. • **xml**—Displays negotiated compatibility matrix information in XML format.

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

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

CardType: WS-C4507R(112), Uid: 2,  Image Ver: 12.2(31)SGA
Image Name: cat4500-ENTSERVICES-M

Cid     Eid     Sid     pSid    pUid      Compatibility
=======================================================
2       1       262151  3       1         COMPATIBLE
3       1       262160  5       1         COMPATIBLE
4       1       262163  9       1         COMPATIBLE
5       1       262186  25      1         COMPATIBLE
7       1       262156  10      1         COMPATIBLE
8       1       262148  7       1         COMPATIBLE
9       1       262155  1       1         COMPATIBLE
10      1       262158  2       1         COMPATIBLE
11      1       262172  6       1         COMPATIBLE
100     1       262166  13      1         COMPATIBLE
110     113     262159  14      1         COMPATIBLE
200     1       262167  24      1         COMPATIBLE
2002    1       -       -       -         UNAVAILABLE
2003    1       262185  23      1         COMPATIBLE
2004    1       262175  16      1         COMPATIBLE
2008    1       262147  26      1         COMPATIBLE
2008    1       262168  27      1         COMPATIBLE
```

```
2010    1       262171  32      1         COMPATIBLE
2012    1       262180  31      1         COMPATIBLE
2021    1       262170  41      1         COMPATIBLE
2022    1       262152  42      1         COMPATIBLE
2023    1       -       -       -         UNAVAILABLE
2024    1       -       -       -         UNAVAILABLE
2025    1       -       -       -         UNAVAILABLE
2026    1       -       -       -         UNAVAILABLE
2027    1       -       -       -         UNAVAILABLE
2028    1       -       -       -         UNAVAILABLE
2054    1       262169  8       1         COMPATIBLE
2058    1       262154  29      1         COMPATIBLE
2059    1       262179  30      1         COMPATIBLE
2067    1       262153  12      1         COMPATIBLE
2068    1       196638  40      1         COMPATIBLE
2070    1       262145  21      1         COMPATIBLE
2071    1       262178  11      1         COMPATIBLE
2072    1       262162  28      1         COMPATIBLE
2073    1       262177  33      1         COMPATIBLE
2077    1       262165  35      1         COMPATIBLE
2078    1       196637  34      1         COMPATIBLE
2079    1       262176  36      1         COMPATIBLE
2081    1       262150  37      1         COMPATIBLE
2082    1       262161  39      1         COMPATIBLE
2083    1       262184  20      1         COMPATIBLE
2084    1       262183  38      1         COMPATIBLE
4001    101     262181  17      1         COMPATIBLE
4002    201     262164  18      1         COMPATIBLE
4003    301     262182  19      1         COMPATIBLE
4004    401     262146  22      1         COMPATIBLE
4005    1       262149  4       1         COMPATIBLE

Message group summary:
Cid     Eid     GrpId     Sid     pSid    pUid    Nego Result
=============================================================
2       1       1         262151  3       1       Y
3       1       1         262160  5       1       Y
4       1       1         262163  9       1       Y
5       1       1         262186  25      1       Y
7       1       1         262156  10      1       Y
8       1       1         262148  7       1       Y
9       1       1         262155  1       1       Y
10      1       1         262158  2       1       Y
11      1       1         262172  6       1       Y
100     1       1         262166  13      1       Y
110     113     115       262159  14      1       Y
200     1       1         262167  24      1       Y
2002    1       2         -       -       -       N - did not negotiate
2003    1       1         262185  23      1       Y
2004    1       1         262175  16      1       Y
2008    1       1         262147  26      1       Y
2008    1       2         262168  27      1       Y
2010    1       1         262171  32      1       Y
2012    1       1         262180  31      1       Y
2021    1       1         262170  41      1       Y
2022    1       1         262152  42      1       Y
2023    1       1         -       -       -       N - did not negotiate
2024    1       1         -       -       -       N - did not negotiate
2025    1       1         -       -       -       N - did not negotiate
2026    1       1         -       -       -       N - did not negotiate
2027    1       1         -       -       -       N - did not negotiate
2028    1       1         -       -       -       N - did not negotiate
2054    1       1         262169  8       1       Y
2058    1       1         262154  29      1       Y
```

```
2059    1       1         262179  30      1       Y
2067    1       1         262153  12      1       Y
2068    1       1         196638  40      1       Y
2070    1       1         262145  21      1       Y
2071    1       1         262178  11      1       Y
2072    1       1         262162  28      1       Y
2073    1       1         262177  33      1       Y
2077    1       1         262165  35      1       Y
2078    1       1         196637  34      1       Y
2079    1       1         262176  36      1       Y
2081    1       1         262150  37      1       Y
2082    1       1         262161  39      1       Y
2083    1       1         262184  20      1       Y
2084    1       1         262183  38      1       Y
4001    101     1         262181  17      1       Y
4002    201     1         262164  18      1       Y
4003    301     1         262182  19      1       Y
4004    401     1         262146  22      1       Y
4005    1       1         262149  4       1       Y

List of Clients:
Cid       Client Name              Base/Non-Base
================================================
2         ISSU Proto client        Base
3         ISSU RF                  Base
4         ISSU CF client           Base
5         ISSU Network RF client   Base
7         ISSU CONFIG SYNC         Base
8         ISSU ifIndex sync        Base
9         ISSU IPC client          Base
10        ISSU IPC Server client   Base
11        ISSU Red Mode Client     Base
100       ISSU rfs client          Base
110       ISSU ifs client          Base
200       ISSU Event Manager clientBase
2002      CEF Push ISSU client     Base
2003      ISSU XDR client          Base
2004      ISSU SNMP client         Non-Base
2008      ISSU Tableid Client      Base
2010      ARP HA                   Base
2012      ISSU HSRP Client         Non-Base
2021      XDR Int Priority ISSU cliBase
2022      XDR Proc Priority ISSU clBase
2023      FIB HWIDB ISSU client    Base
2024      FIB IDB ISSU client      Base
2025      FIB HW subblock ISSU clieBase
2026      FIB SW subblock ISSU clieBase
2027      Adjacency ISSU client    Base
2028      FIB IPV4 ISSU client     Base
2054      ISSU process client      Base
2058      ISIS ISSU RTR client     Non-Base
2059      ISIS ISSU UPD client     Non-Base
2067      ISSU PM Client           Base
2068      ISSU PAGP_SWITCH Client  Non-Base
2070      ISSU Port Security clientNon-Base
2071      ISSU Switch VLAN client  Non-Base
2072      ISSU dot1x client        Non-Base
2073      ISSU STP                 Non-Base
2077      ISSU STP MSTP            Non-Base
2078      ISSU STP IEEE            Non-Base
2079      ISSU STP RSTP            Non-Base
2081      ISSU DHCP Snooping clientNon-Base
2082      ISSU IP Host client      Non-Base
2083      ISSU Inline Power client Non-Base
```

```
2084      ISSU IGMP Snooping clientNon-Base
4001      ISSU C4K Chassis client  Base
4002      ISSU C4K Port client     Base
4003      ISSU C4K Rkios client    Base
4004      ISSU C4K HostMan client  Base
4005      ISSU C4k GaliosRedundancyBase
```

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 cat4500-ENTSERVICES-M(112) - cat4500-ENTSERVICES-M(112)
        ==========================================
        Start Flag (0xDEADBABE)

                My Image ver:  12.2(53)SG
                Peer Version    Compatibility
                ------------    -------------
                12.2(31)SGA5            Base(2)
                12.2(44)SG              Base(2)
                12.2(31)SGA6            Base(2)
                12.2(31)SGA7            Base(2)
                12.2(46)SG              Base(2)
                12.2(44)SG1             Base(2)
                12.2(31)SGA8            Base(2)
                12.2(50)SG              Dynamic(0)
                12.2(31)SGA9            Base(2)
                12.2(50)SG1             Dynamic(0)
                12.2(50)SG2             Dynamic(0)
                12.2(52)SG              Dynamic(0)
                12.2(31)SGA10           Base(2)
                12.2(50)SG3             Dynamic(0)
                12.2(53)SG              Comp(3)
```

Dynamic(0) was introduced in Cisco IOS Release 12.2(50)SG with the Dynamic Image Version Compatibility (DIVC) feature. With DIVC, Dynamic(0) is stored instead of Incomp(1), Base(2), or Comp(3). Compatibility is determined during runtime when two different DIVC-capable 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 rollback negotiation between the two images. You never observe Incomp(1) as long as the other image name is present in the stored compatibility matrix.

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>` **`enable`**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	`Switch#` **`show issu comp-matrix`** {**`negotiated`** | **`stored`** | **`xml`**}	Displays information regarding the ISSU compatibility matrix. - **negotiated**—Displays negotiated compatibility matrix information. - **stored**—Displays negotiated compatibility matrix information. - **xml**—Displays negotiated compatibility matrix information in XML format. **Note** 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#` **`show package compatibility`**	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#
```

With Dynamic Image Version Compatibility (DIVC) , Dynamic(0) is stored instead of Incomp(1), Base(2), or Comp(3). Compatibility is determined during runtime when two different DIVC-capable 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 software images. You never observe Incomp(1) as long as the other image 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
......
......
```

Related Documents

Related Topic	Document Title
Performing ISSU	*Cisco IOS Software: Guide to Performing In Service Software Upgrades*
Information about Cisco Nonstop Forwarding	*Cisco Nonstop Forwarding* *http://www.cisco.com/en/US/docs/ios/12_2s/feature/guide/fsnsf20s.html*
Information about Stateful Switchover	*Stateful Switchover* *http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/sso120s.html*
ISSU and MPLS clients	ISSU MPLS Clients