



# Configuring Supervisor Engine Redundancy Using RPR and SSO on Supervisor Engine 6-E and Supervisor Engine 6L-E

Catalyst 4500 series switches allow a redundant supervisor engine to take over if the active supervisor engine fails. In software, supervisor engine redundancy is enabled by running the redundant supervisor engine in route processor redundancy (RPR) or stateful switchover (SSO) operating mode.



**Note**

The minimum ROMMON requirement for running SSO is Cisco IOS Release 12.1(20r)EW1 or Cisco IOS Release 12.2(20r)EW1.

This chapter describes how to configure supervisor engine redundancy on the Catalyst 4507R and Catalyst 4510R switches.



**Note**

For information on Cisco nonstop forwarding (NSF) with SSO, see [Chapter 12, “Configuring Cisco NSF with SSO Supervisor Engine Redundancy.”](#)

This chapter contains these major sections:

- [About Supervisor Engine Redundancy, page 11-2](#)
- [About Supervisor Engine Redundancy Synchronization, page 11-4](#)
- [Supervisor Engine Redundancy Guidelines and Restrictions, page 11-5](#)
- [Configuring Supervisor Engine Redundancy, page 11-7](#)
- [Performing a Manual Switchover, page 11-12](#)
- [Performing a Software Upgrade, page 11-13](#)
- [Manipulating Bootflash on the Redundant Supervisor Engine, page 11-14](#)



**Note**

For complete syntax and usage information for the switch commands used in this chapter, see the [Cisco IOS Command Reference Guides for the Catalyst 4500 Series Switch](#).

If a command is not in the *Cisco Catalyst 4500 Series Switch Command Reference*, you can locate it in the [Cisco IOS Master Command List, All Releases](#).

# About Supervisor Engine Redundancy

These sections describe supervisor engine redundancy:

- [Overview, page 11-2](#)
- [RPR Operation, page 11-2](#)
- [SSO Operation, page 11-3](#)

## Overview

With supervisor engine redundancy enabled, if the active supervisor engine fails or if a manual switchover is performed, the redundant supervisor engine becomes the active supervisor engine. The redundant supervisor engine has been automatically initialized with the startup configuration of the active supervisor engine, shortening the switchover time (30 seconds or longer in RPR mode, depending on the configuration; subsecond in SSO mode).

In addition to the reduced switchover time, supervisor engine redundancy supports the following:

- Online insertion and removal (OIR) of the redundant supervisor engine.  
Supervisor engine redundancy allows OIR of the redundant supervisor engine for maintenance. When the redundant supervisor engine is inserted, the active supervisor engine detects its presence, and the redundant supervisor engine boots into a partially-initialized state in RPR mode and a fully-initialized state in SSO mode.
- Software upgrade. See the [“Performing a Software Upgrade” section on page 11-13](#).  
To minimize down time during software changes on the supervisor engine, load the new image on the redundant supervisor engine, and conduct a switchover.

When power is first applied to a switch, the supervisor engine that boots first becomes the active supervisor engine and remains active until a switchover occurs.

A switchover occurs when one or more of the following events take place:

- The active supervisor engine fails (due to either hardware or software function) or is removed.
- A user forces a switchover.
- A user reloads the active supervisor engine.

## RPR Operation

RPR is supported in Cisco IOS Release 12.2(12c)EW and later releases. When a redundant supervisor engine runs in RPR mode, it starts up in a partially-initialized state and is synchronized with the persistent configuration of the active supervisor engine.

**Note**

---

Persistent configuration includes the following components: startup-config, boot variables, config-register, and VLAN database.

---

The redundant supervisor engine pauses the startup sequence after basic system initialization, and in the event that the active supervisor engine fails, the redundant supervisor engine becomes the new active supervisor engine.

In a supervisor engine switchover, traffic is disrupted because in the RPR mode all of the physical ports restart since there is no state maintained between supervisor engines relating to module types and statuses. When the redundant supervisor engine completes its initialization, it reads hardware information directly from the module.

## SSO Operation

SSO is supported in Cisco IOS Release 12.2(20)EWA and later releases. When a redundant supervisor engine runs in SSO mode, the redundant supervisor engine starts up in a fully-initialized state and synchronizes with the persistent configuration and the running configuration of the active supervisor engine. It subsequently maintains the state on the protocols listed below, and all changes in hardware and software states for features that support stateful switchover are kept in synchronization. Consequently, it offers zero interruption to Layer 2 sessions in a redundant supervisor engine configuration.

Because the redundant supervisor engine recognizes the hardware link status of every link, ports that were active before the switchover remain active, including the uplink ports. However, because uplink ports are physically on the supervisor engine, they will be disconnected if the supervisor engine is removed.

If the active supervisor engine fails, the redundant supervisor engine become active. This newly active supervisor engine uses existing Layer 2 switching information to continue forwarding traffic. Layer 3 forwarding is delayed until the routing tables have been repopulated in the newly active supervisor engine.

SSO supports stateful switchover of the following Layer 2 features. The state of these features is preserved between both the active and redundant supervisor engines:

- 802.3
- 802.3u
- 802.3x (Flow Control)
- 802.3ab (GE)
- 802.3z (Gigabit Ethernet including CWDM)
- 802.3ad (LACP)
- 802.1p (Layer 2 QoS)
- 802.1q
- 802.1X (Authentication)
- 802.1D (Spanning Tree Protocol)
- 802.3af (Inline power)
- PAgP
- VTP
- Dynamic ARP Inspection
- DHCP snooping
- IP source guard
- IGMP snooping (versions 1 and 2)
- DTP (802.1q and ISL)
- MST

- PVST+
- Rapid-PVST
- PortFast/UplinkFast/BackboneFast
- BPDU guard and filtering
- Voice VLAN
- Port security
- Unicast MAC filtering
- ACL (VACLs, PACLS, RACLs)
- QoS (DBL)
- Multicast storm control/broadcast storm control

SSO is compatible with the following list of features. However, the protocol database for these features is not synchronized between the redundant and active supervisor engines:

- 802.1Q tunneling with Layer 2 Protocol Tunneling (L2PT)
- Baby giants
- Jumbo frame support
- CDP
- Flood blocking
- UDLD
- SPAN/RSPAN
- NetFlow

The following features are learned on the redundant supervisor engine if the SSO feature is enabled:

- All Layer 3 protocols on Catalyst 4500 series switches (Switch Virtual Interfaces)

## About Supervisor Engine Redundancy Synchronization

During normal operation, the persistent configuration (RPR and SSO) and the running configuration (SSO only) are synchronized by default between the two supervisor engines. In a switchover, the new active supervisor engine uses the current configuration.




---

**Note** You cannot enter CLI commands on the redundant supervisor engine console.

---

These sections describe supervisor engine redundancy synchronization:

- [RPR Supervisor Engine Configuration Synchronization, page 11-4](#)
- [SSO Supervisor Engine Configuration Synchronization, page 11-5](#)

## RPR Supervisor Engine Configuration Synchronization

Because the redundant supervisor engine is only partially initialized in RPR mode, it interacts with the active supervisor engine only to receive configuration changes at startup and upon saving the configuration changes.

When a redundant supervisor engine is running in RPR mode, the following events trigger synchronization of the configuration information:

- When the redundant supervisor engine boots, the **auto-sync** command synchronizes the persistent configuration. This command is enabled by default. For details, refer to [“Synchronizing the Supervisor Engine Configurations” section on page 11-11](#).
- When the active supervisor engine detects the redundant supervisor engine, the configuration information is synchronized from the active supervisor engine to the redundant supervisor engine. This synchronization overwrites any existing startup configuration file on the redundant supervisor engine.
- When you make changes to the configuration, you must use the **write** command to save and synchronize the startup configuration of the redundant supervisor engine.

## SSO Supervisor Engine Configuration Synchronization

When a redundant supervisor engine runs in SSO mode, the following events trigger synchronization of the configuration information:

- When the active supervisor detects the redundant supervisor engine, synchronization of the persistent and running configuration takes place, allowing the redundant supervisor engine to arrive at a fully-initiated state.
- When real-time changes occur, the active supervisor engine synchronizes the running-config and (or) the persistent configuration (if necessary) with the redundant supervisor engine.
- When you change the configuration, you must use the **write** command to allow the active supervisor engine to save and synchronize the startup configuration of the redundant supervisor engine.

## Supervisor Engine Redundancy Guidelines and Restrictions

The following guidelines and restrictions apply to supervisor engine redundancy:

- If SSO mode cannot be established between the active and standby supervisor engines because of an incompatibility in the configuration file, a mismatched command list (MCL) is generated at the active supervisor engine and a reload into RPR mode is forced for the standby supervisor engine. Subsequent attempts to establish SSO, after removing the offending configuration and rebooting the standby supervisor engine with the exact same image, might cause the C4K\_REDUNDANCY-2-`IOS_VERSION_CHECK_FAIL` and ISSU-3-`PEER_IMAGE_INCOMPATIBLE` messages to appear because the peer image is listed as incompatible. If the configuration problem can be corrected, you can clear the peer image from the incompatible list with the **redundancy config-sync ignore mismatched-commands EXEC** command while the peer is in a standby cold (RPR) state. This action allows the standby supervisor engine to boot in standby hot (SSO) state when it reloads.

Here are the steps:

- 
- Step 1** Clear the offending configuration (that caused an MCL) while the standby supervisor engine is in standby cold (RPR) state.
  - Step 2** Enter the **redundancy config-sync ignore mismatched-commands EXEC** command at the active standby supervisor engine.
  - Step 3** Perform **write memory**.

**Step 4** Reload the standby supervisor engine with the **redundancy reload peer** command.

- If you configure Supervisor Engine V-10GE to use both Gigabit Ethernet and 10-Gigabit Ethernet uplinks without WS-X4302-GB in slot 10, module 10 is disabled and you cannot rollback the configuration to use gigabit ports.

Enter the following commands to recover:

```

config t
hw-module uplink select tengigabitethernet // This sets the switch back to default
mode
!
ctrl z
wr me
redundancy reload shelf // The switch will reload with all 10 modules working ok

//The switch reloads

config t
hw-module uplink select gigabitethernet // This sets the switch to the desired link
!
ctrl z
wr me
redundancy reload shelf // The switch reloads with module 10 active with the gigabit
ethernet port(s) ON and the ten gigabit ethernet port(s) Off

```

- RPR requires Cisco IOS Release 12.1(12c)EW, Release 12.1(19)E or later releases. SSO requires Cisco IOS Release 12.2(20)EWA or later releases.
- The Catalyst 4507R switch and the 4510R switch are the only Catalyst 4500 series switches that support supervisor engine redundancy.
- In Cisco IOS Release 12.2(25)SG and later releases on a Catalyst 4507R series switch, 10-Gigabit Ethernet and Gigabit Ethernet uplinks are concurrently usable.
- Redundancy requires both supervisor engines in the chassis to have the same components (model, memory, NFL daughter card), and to use the same Cisco IOS software image.
- When the WS-X4516 active and redundant supervisor engines are installed in the same chassis, the four uplink ports (Gig1/1, Gig2/1, Gig 1/2, and Gig2/2) are available.
- The active and redundant supervisor engines in the chassis must be in slots 1 and 2.
- Each supervisor engine in the chassis must have its own flash device and console port connections to operate the switch on its own.
- Each supervisor engine must have a unique console connection. Do not connect a Y cable to the console ports.
- Supervisor engine redundancy does not provide supervisor engine load balancing.
- The Cisco Express Forwarding (CEF) table is cleared on a switchover. As a result, routed traffic is interrupted until route tables reconverge. This reconvergence time is minimal because the SSO feature reduces the supervisor engine redundancy switchover time from 30+ seconds to subsecond, so Layer 3 also has a faster failover time if the switch is configured for SSO.
- Static IP routes are maintained across a switchover because they are configured from entries in the configuration file.
- Information about Layer 3 dynamic states that is maintained on the active supervisor engine is not synchronized to the redundant supervisor engine and is lost on switchover.

- Starting with Cisco IOS Release 12.2, if an unsupported condition is detected (such as when the active supervisor engine is running Cisco IOS Release 12.2(20)EW and the redundant supervisor engine is running Cisco IOS Release 12.1(20)EW), the redundant supervisor engine is reset multiple times and then placed in ROMMON mode. It is important to follow the procedures outlined in the [“Performing a Software Upgrade” section on page 11-13](#).
- If you are running (or upgrading to) Cisco IOS Release 12.2(20)EWA or Cisco IOS Release 12.2(25)EW and are using a single supervisor engine in a redundant chassis (Catalyst 4507R or Catalyst 4510R series switch), and you intend to use routed ports, do one of the following:
  - Use SVIs instead of routed ports.
  - Change the redundancy mode from SSO to RPR.

- Configuration changes made to the redundant supervisor engine through SNMP synchronization and SNMP set operations in SSO mode are not synchronized to the redundant supervisor engine. Even though you can still perform SNMP set operations in SSO mode, you might experience unexpected behavior.

After you configure the switch through SNMP in SSO mode, copy the running-config file to the startup-config file on the active supervisor engine to trigger synchronization of the startup-config file on the redundant supervisor engine. Reload the redundant supervisor engine so that the new configuration is applied on the redundant supervisor engine.

- You cannot perform configuration changes during the startup (bulk) synchronization. If you attempt to make configuration changes during this process, the following message is generated:

```
Config mode locked out till standby initializes
```

- If configuration changes occur at the same time as a supervisor engine switchover, these configuration changes are lost.
- If you remove a line card from a redundant switch and initiate an SSO switchover, and then reinsert the line card, all interfaces are shutdown. The rest of the original line card configuration is preserved.

This situation only occurs if a switch had reached SSO before you removed the line card.

## Configuring Supervisor Engine Redundancy

These sections describe how to configure supervisor engine redundancy:

- [Configuring Redundancy, page 11-8](#)
- [Virtual Console for Standby Supervisor Engine, page 11-10](#)
- [Synchronizing the Supervisor Engine Configurations, page 11-11](#)

## Configuring Redundancy

To configure redundancy, perform this task:

	Command	Purpose
Step 1	Switch(config)# <b>redundancy</b>	Enters redundancy configuration mode.
Step 2	Switch(config-red)# <b>mode {sso   rpr}</b>	Configures SSO or RPR. When this command is entered, the redundant supervisor engine is reloaded and begins to work in SSO or RPR mode.
Step 3	Switch# <b>show running-config</b>	Verifies that SSO or RPR is enabled.
Step 4	Switch# <b>show redundancy</b> [clients   counters   history   states]	Displays the redundancy information (counter, state, and so on) for the active and redundant supervisor engines.

When configuring redundancy, note the following:

- The **sso** keyword is supported in Cisco IOS Release 12.2(20)EWA and later releases.
- The **rpr** keyword is supported in Cisco IOS Release 12.1(12c)EW and later releases.

This example shows how to configure the system for SSO and display the redundancy facility information:

```
Switch> enable
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# redundancy
Switch(config-red)# mode sso
Switch(config-red)# end
Switch# show redundancy
Redundant System Information :
-----
        Available system uptime = 2 days, 2 hours, 39 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 1
        Current Software state = ACTIVE
Uptime in current state = 2 days, 2 hours, 39 minutes
        Image Version = Cisco Internetwork Operating System Software
IOS (tm) Catalyst 4000 L3 Switch Software (cat4000-I5S-M), Version 12.2(20)EWA(3
.92), CISCO INTERNAL USE ONLY ENHANCED PRODUCTION VERSION
Copyright (c) 1986-2004 by cisco Systems, Inc.
Compiled Wed 14-Jul-04 04:42 by esi
        BOOT = bootflash:cat4000-i5s-mz.122_20_EWA_392,1
Configuration register = 0x2002
```



```

Peer Processor Information :
-----
          Standby Location = slot 2
          Current Software state = STANDBY HOT
          Uptime in current state = 2 days, 2 hours, 39 minutes
          Image Version = Cisco Internetwork Operating System Software
IOS (tm) Catalyst 4000 L3 Switch Software (cat4000-I5S-M), Version 12.2(20)EWA(3
.92), CISCO INTERNAL USE ONLY ENHANCED PRODUCTION VERSION
Copyright (c) 1986-2004 by cisco Systems, Inc.
Compiled Wed 14-Jul-04 0
          BOOT = bootflash:cat4000-i5s-mz.122_20_EWA_392,1
          Configuration register = 0x2002

Switch#

```

This example shows how to display redundancy facility state information:

```

Switch# show redundancy states
my state = 13 -ACTIVE
    peer state = 8 -STANDBY HOT
        Mode = Duplex
        Unit = Primary
        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 = 21
    client_notification_TMR = 240000 milliseconds
        keep_alive TMR = 9000 milliseconds
            keep_alive count = 0
            keep_alive threshold = 18
                RF debug mask = 0x0

Switch#

```

This example shows how to change the system configuration from RPR to SSO mode:

```

Switch(config)# redundancy
Switch(config-red)# mode
Switch(config-red)# mode sso
Changing to sso mode will reset the standby. Do you want to continue?[confirm]
Switch(config-red)# end
Switch#
*Aug 1 13:11:16: %C4K_REDUNDANCY-3-COMMUNICATION: Communication with the peer Supervisor
has been lost
*Aug 1 13:11:16: %C4K_REDUNDANCY-3-SIMPLEX_MODE: The peer Supervisor has been lost

```

This example shows how to change the system configuration from SSO to RPR mode:

```

Switch(config)# redundancy
Switch(config-red)# mode rpr
Changing to rpr mode will reset the standby. Do you want to continue?[confirm]
Switch(config-red)# end
*Aug 1 13:11:16: %C4K_REDUNDANCY-3-COMMUNICATION: Communication with the peer Supervisor
has been lost
*Aug 1 13:11:16: %C4K_REDUNDANCY-3-SIMPLEX_MODE: The peer Supervisor has been lost

```

## Virtual Console for Standby Supervisor Engine

Catalyst 4500 series switches can be configured with two supervisor engines to provide redundancy. When the switch is powered, one of the supervisor engines becomes active and remains active until a switchover occurs. The other supervisor engine remains in standby mode.

Each supervisor engine has its own console port. Access to the standby supervisor engine is possible only through the console port of the standby supervisor engine. You must connect to the standby console to access, monitor or debug the standby supervisor.

The virtual console for a standby supervisor Engine enables you to access the standby console from the active supervisor engine without requiring a physical connection to the standby console. It uses IPC over EOBC to communicate with the standby supervisor engine, which emulates the standby console on the active supervisor engine. Only one active standby console session is active at any time.

The virtual console for the standby supervisor engine allows users who are logged onto the active supervisor engine to remotely execute **show** commands on the standby supervisor engine and view the results on the active supervisor engine. Virtual console is available only from the active supervisor engine.

You can access the standby virtual console from the active supervisor engine with the **attach module**, **session module**, or **remote login** commands on the active supervisor engine. You must be in privilege EXEC mode (level 15) to run these commands to access the standby console.

Once you enter the standby virtual console, the terminal prompt automatically changes to *hostname-standby-console* where *hostname* is the configured name of the switch. The prompt is restored back to the original prompt when you exit the virtual console.

You exit the virtual console with the **exit** or **quit** commands. When the inactivity period of the terminal on the active supervisor engine where you logged in exceeds the configured idle time, you are automatically logged out of the terminal on the active supervisor engine. In this instance, the virtual console session is also terminated. Virtual console session is also automatically terminated when the standby is rebooted. After the standby boots up, you need to create another virtual console session.

To log in to the standby supervisor engine using a virtual console, enter the following command:

```
Switch# session module 2
Connecting to standby virtual console
Type "exit" or "quit" to end this session

Switch-standby-console# exit
Switch#
```

If the standby console is not enabled, the following message appears:

```
Switch-standby-console#
Standby console disabled.
Valid commands are: exit, logout
```



### Note

The standby virtual console provides the standard features that are available from the supervisor console such as command history, command completion, command help and partial command keywords.

The following limitations apply to the standby virtual console:

- All commands on the virtual console run to completion. It does not provide the auto-more feature; it functions as if the **terminal length 0** command has been executed. It is also noninteractive. You cannot interrupt or abort an executing command by any key sequence on the active supervisor engine. If a command produces considerable output, the virtual console displays it on the supervisor engine screen.

- The virtual console is noninteractive. Because the virtual console does not detect the interactive nature of a command, any command that requires user interaction causes the virtual console to wait until the RPC timer aborts the command.

The virtual console timer is set to 60 seconds. The virtual console returns to its prompt after 60 seconds. During this time, you cannot abort the command from the key board. You must wait for the timer to expire before you continue.

- You cannot use virtual console to view debug and syslog messages that are being displayed on the standby supervisor engine. The virtual console only displays the output of commands that are executed from the virtual console. Other information that is displayed on the real standby console does not appear on the virtual console.

## Synchronizing the Supervisor Engine Configurations

To manually synchronize the configurations used by the two supervisor engines, perform this task on the active supervisor engine:

	Command	Purpose
Step 1	Switch(config)# <b>redundancy</b>	Enters redundancy configuration mode.
Step 2	Switch(config-red)# <b>main-cpu</b>	Enters main-cpu configuration submode.
Step 3	Switch(config-r-mc)# <b>auto-sync</b> { <b>startup-config</b>   <b>config-register</b>   <b>bootvar</b>   <b>standard</b> }	Synchronizes the configuration elements.
Step 4	Switch(config-r-mc)# <b>end</b>	Returns to privileged EXEC mode.
Step 5	Switch# <b>copy running-config startup-config</b>	Synchronizes the running configuration in dynamic random-access memory (DRAM) to the startup configuration file in NVRAM.  <b>Note</b> This step is not required to synchronize the running configuration file in (DRAM).



### Note

Configuration changes made to the active supervisor engine through SNMP are not synchronized to the redundant supervisor engine. For information on how to handle this situation, see the “[Supervisor Engine Redundancy Guidelines and Restrictions](#)” section on page 11-5.



### Note

The **auto-sync** command controls the synchronization of the config-reg, bootvar, and startup/private configuration files only. The calendar and VLAN database files are always synchronized when they change. In SSO mode, the running-config is always synchronized.

This example shows how to reenable the default automatic synchronization feature using the **auto-sync standard** command to synchronize the startup-config and config-register configuration of the active supervisor engine with the redundant supervisor engine. Updates for the boot variables are automatic and cannot be disabled.

```
Switch(config)# redundancy
Switch(config-red)# main-cpu
Switch(config-r-mc)# auto-sync standard
Switch(config-r-mc)# end
Switch# copy running-config startup-config
```

**Note**

To manually synchronize individual elements of the standard auto-sync configuration, disable the default automatic synchronization feature.

**Note**

When you configure the auto-sync standard, the individual sync options such as no auto-sync startup-config are ignored.

This example shows how to disable default automatic synchronization and allow only automatic synchronization of the config-registers of the active supervisor engine to the redundant supervisor engine, while disallowing synchronization of the startup configuration:

```
Switch(config)# redundancy
Switch(config-red)# main-cpu
Switch(config-r-mc)# no auto-sync standard
Switch(config-r-mc)# auto-sync config-register
Switch(config-r-mc)# end
```

## Performing a Manual Switchover

This section describes how to perform a manual switchover (from the active supervisor engine to the redundant supervisor engine) for test purposes. We recommend that you perform a manual switchover prior to deploying SSO in your production environment.

**Note**

This discussion assumes that SSO has been configured as the redundant mode.

To perform a manual switchover, perform this task on the active supervisor engine:

	Command	Purpose
Step 1	Switch# <b>show redundancy</b>	Verifies that the peer state is in the standby hot state. See the example of the <b>show redundancy states</b> command on page 6-10.
Step 2	Switch# <b>redundancy force-switchover</b>	Launches switchover from the active supervisor engine to the redundant supervisor engine. If the state of the redundant supervisor engine is not standby hot, this command does not execute.

Be aware of these usage guidelines:

- To force a switchover, the redundant supervisor engine must be in a standby hot state. You can verify the state with the **show redundancy** command. If the state is not standby hot, the **redundancy force-switchover** command does not execute.
- Use the **redundancy force-switchover** command, rather than the **reload** command, to initiate a switchover. The **redundancy force-switchover** command first verifies that the redundant supervisor engine is in the correct state. If you enter the **reload** command and the status is not standby hot, the **reload** command resets the current supervisor engine only.

After an initial switchover, there might be occasions when you want to make the supervisor engine in slot 1 of the chassis the active supervisor engine. If the image on supervisor engine 1 is the one you intend to run on both supervisor engines, it is not necessary to reboot the image on the supervisor engine in slot 1 to make it redundant. Instead, you can force another switchover. However, if you want a newer version of the image to run on both supervisor engines, follow the steps under “Performing a Software Upgrade” on page 13. Use the **show module** command to see which slot contains the active supervisor engine, and force another switchover if necessary.

## Performing a Software Upgrade

The software upgrade procedure supported by supervisor engine redundancy allows you to reload the Cisco IOS software image on the redundant supervisor engine, and once complete, reload the active supervisor engine once.

The software upgrade procedure supported by supervisor engine redundancy allows you to reload the Cisco IOS software image on the redundant supervisor engine, and once complete, reloads the active supervisor engine once.

The following scenario is not supported: An active supervisor engine running Cisco IOS Release 12.1(x)E, and a standby supervisor engine running Cisco IOS Release 12.2(x)S. The standby supervisor engine resets repeatedly.

If you are trying to upgrade redundant supervisor engines from Cisco IOS Release 12.1(x)E to 12.2(x)S, this requires a full system reboot.

To perform a software upgrade, perform this task:

	Command	Purpose
<b>Step 1</b>	Switch# <b>copy</b> <i>source_device:source_filename</i> <b>slot0:target_filename</b>  Or:  Switch# <b>copy</b> <i>source_device:source_filename</i> <b>bootflash:target_filename</b>	Copies the new Cisco IOS software image to bootflash on the supervisor engine.
<b>Step 2</b>	Switch# <b>copy</b> <i>source_device:source_filename</i> <b>slaveslot0:target_filename</b>  Or:  Switch# <b>copy</b> <i>source_device:source_filename</i> <b>slavebootflash:target_filename</b>	Copies the new image to a slave device (such as slavebootflash and slaveslot0).
<b>Step 3</b>	Switch# <b>config terminal</b> Switch(config)# <b>config-register 0x2</b> Switch(config)# <b>boot system flash</b> <i>device:file_name</i>	Configures the supervisor engines to boot the new image.  If your system was configured to autoboot an earlier image, enter the following command string to boot the new image instead: <b>no boot system flash</b> <i>device:old_file_name</i>
<b>Step 4</b>	Switch(config)# <b>redundancy</b>	Enters redundancy configuration mode.
<b>Step 5</b>	Switch(config-red)# <b>main-cpu</b>	Enters main-cpu configuration submenu.
<b>Step 6</b>	Switch(config-r-mc)# <b>auto-syn standard</b>	Synchronizes the configuration elements.
<b>Step 7</b>	Switch(config-r-mc)# <b>end</b>	Returns to privileged EXEC mode.

	Command	Purpose
Step 8	Switch# <b>copy running-config start-config</b>	Saves the configuration.
Step 9	Switch# <b>redundancy reload peer</b>	Reloads the redundant supervisor engine and brings it back online (using the new release of the Cisco IOS software).  <b>Note</b> Before proceeding to Step 10, ensure that the switch is operating in RPR mode.
Step 10	Switch# <b>redundancy force-switchover</b>	Conducts a manual switchover to the redundant supervisor engine. The redundant supervisor engine becomes the new active supervisor engine using the new Cisco IOS software image.  The old active supervisor engine reboots with the new image and becomes the redundant supervisor engine.

This example shows how to perform a software upgrade:

```
Switch# config terminal
Switch(config)# config-register 0x2
Switch(config)# boot system flash slot0:cat4000-i5s-mz.122-20.EWA
Switch(config)# redundancy
Switch(config-red)# main-cpu
Switch(config-r-mc)# auto-syn standard
Switch(config-r-mc)# end
Switch# copy running-config start-config
Switch# redundancy reload peer
Switch# redundancy force-switchover
Switch#
```

This example illustrates how to verify that the running configuration on the active supervisor engine has successfully synchronized with the redundant supervisor engine:

```
Switch# config terminal
Switch(config)# redundancy
Switch(config-red)# main-cpu
Switch(config-r-mc)# auto-sync standard
4d01h: %C4K_REDUNDANCY-5-CONFIGSYNC: The bootvar has been successfully synchronized to the
standby supervisor
4d01h: %C4K_REDUNDANCY-5-CONFIGSYNC: The config-reg has been successfully synchronized to
the standby supervisor
4d01h: %C4K_REDUNDANCY-5-CONFIGSYNC: The startup-config has been successfully synchronized
to the standby supervisor
4d01h: %C4K_REDUNDANCY-5-CONFIGSYNC: The private-config has been successfully synchronized
to the standby supervisor
```

The example above shows that the boot variable, the config-register, and the startup configuration from the active supervisor engine have successfully synchronized to the redundant supervisor engine.

## Manipulating Bootflash on the Redundant Supervisor Engine



**Note** The console port on the redundant supervisor engine is not available.

To manipulate the redundant supervisor engine bootflash, perform one or more of the following commands:

Command	Purpose
Switch# <b>dir slaveslot0:</b> <i>target_filename</i>  or Switch# <b>dir slavebootflash:</b> <i>target_filename</i>	Lists the contents of the <b>slot0:</b> device on the redundant supervisor engine.  Lists the contents of the <b>bootflash:</b> device on the redundant supervisor engine.
Switch# <b>delete slaveslot0:</b> <i>target_filename</i>  or Switch# <b>delete slavebootflash:</b> <i>target_filename</i>	Deletes specific files from the <b>slot0:</b> device on the redundant supervisor engine.  Deletes specific files from the <b>bootflash:</b> device on the redundant supervisor engine.
Switch# <b>squeeze slaveslot0:</b>  or Switch# <b>squeeze slavebootflash:</b>	Squeezes the <b>slot0:</b> device on the redundant supervisor engine.  Squeezes the <b>bootflash:</b> device on the redundant supervisor engine.
Switch# <b>format slaveslot0:</b>  or Switch# <b>format slavebootflash:</b>	Formats the <b>slot0:</b> device on the redundant supervisor engine.  Formats the <b>bootflash:</b> device on the redundant supervisor engine.
Switch# <b>copy source_device:source_filename</b> <b>slaveslot0:</b> <i>target_filename</i>  or Switch# <b>copy source_device:source_filename</b> <b>slavebootflash:</b> <i>target_filename</i>	Copies a file from the active supervisor engine to the <b>slot0:</b> device on the redundant supervisor engine.  Copies a file to the <b>bootflash:</b> device on a redundant supervisor engine.  <b>Note</b> Source could be the active supervisor engine or a TFTP server.

