CHAPTER 4

Configuring EHSA Supervisor Engine Redundancy

With 12.1 E releases earlier than Release 12.1(13)E, the Catalyst 6500 series switch supports dual supervisor engines with EHSA.

Note EHSA is not supported in Release 12.1(13)E and later releases (see Chapter 5, "Configuring RPR and RPR+ Supervisor Engine Redundancy," for information about RPR or RPR+ redundancy in Release 12.1(13)E and later releases.

This chapter consists of these sections:

Note For complete syntax and usage information for the commands used in this chapter, refer to the *Catalyst 6500 Series Switch Cisco IOS Command Reference* publication and the Release 12.1 publications at this URL:
http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/tsd_products_support_eol_series_home.html

Supervisor Engine Redundant Operation

Catalyst 6500 series switches support fault resistance by allowing a redundant supervisor engine to take over if the primary supervisor engine fails. The redundant supervisor engine runs in EHSA standby mode.

Note The EHSA feature is not supervisor engine mirroring or load balancing. Network services are disrupted until the redundant supervisor engine takes over and the switch recovers.

EHSA standby mode provides the following features:

- Auto-startup and bootvar synchronization between active and redundant supervisor engines
- Hardware signals that detect and decide the active or redundant status of supervisor engines
- Clock synchronization every 60 seconds from the active to the redundant supervisor engine
- A redundant supervisor engine that is booted but not all subsystems are up: if the active supervisor engine fails, the redundant supervisor engine becomes fully operational
- An operational supervisor engine present in place of the failed unit becomes the redundant supervisor engine

Note The two Gigabit Ethernet interfaces on the redundant supervisor engine are always active.

When the switch is powered on, EHSA runs between the two supervisor engines. The supervisor engine that boots first, either in slot 1 or 2, becomes the EHSA active supervisor engine. The Multilayer Switch Feature Card (MSFC or MSFC2) and Policy Feature Card (PFC or PFC2) become fully operational. The MSFC and PFC on the redundant supervisor engine come out of reset but are not operational.

The following events cause an EHSA switchover:

- Clock synchronization failure between supervisor engines
- MSFC or PFC failure on the active supervisor engine

In a switchover, the redundant supervisor engine becomes fully operational and the following occurs:

- All switching modules power up again
- Remaining subsystems on the MSFC (including Layer 2 and Layer 3 protocols) are brought up
- Access control lists (ACLs) are reprogrammed into supervisor engine hardware

Note In a switchover, there is a disruption of traffic because some address states are lost and then restored after they are dynamically redetermined.

Supervisor Engine Redundancy Requirements

For redundant operation, the following requirements must be met:

- The active and redundant supervisor engines must be in slots 1 and 2.
- Each supervisor engine must have the resources to run the switch on its own, which means all supervisor engine resources are duplicated. In other words, each supervisor engine has its own Flash device and console port connections.

  Note Make a separate console connection to each supervisor engine. Do not connect a "Y" cable to the console ports.

- Both supervisor engines must have the same system image (see the "Copying Files to the Redundant Supervisor Engine" section on page 4-4).

Note If the redundant supervisor engine is running Catalyst operating system software, remove the active supervisor engine and boot the switch with only the redundant supervisor engine installed. Follow the procedures in the release notes to convert the redundant supervisor engine from Catalyst operating system software.

- The configuration register in the startup-config must be set to autoboot (see the "Modifying the Boot Field" section on page 3-23).

Note EHSA does not support booting from the network.

If these requirements are met, the switch functions in EHSA mode by default.

Synchronizing the Supervisor Engine Configurations

During normal operation, the startup-config and config-registers configuration are synchronized by default between the two supervisor engines. In a switchover, the new active supervisor engine uses the current configuration.

Note The boot variables are not synchronized by default.

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

	Command	Purpose
Step 1	`Router(config)#` **`redundancy`**	Enters redundancy configuration mode.
Step 2	`Router(config-r)#` **`main-cpu`**	Enters main-cpu configuration submode.
Step 3	`Router(config-r-mc)#` **`auto-sync`** `{`**`startup-config`** `|` **`config-register`** `|` **`bootvar`** `|` **`standard`**`}`	Synchronizes the configuration elements.
Step 4	`Router(config-r-mc)#` **`end`**	Returns to privileged EXEC mode.
Step 5	`Router#` **`copy running-config startup-config`**	Forces a manual synchronization of the configuration files in NVRAM. **Note** This step is not required to synchronize the running configuration file in DRAM.

Note The **auto-sync standard** command does not synchronize the boot variables.

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:

```
Router(config)# redundancy
Router(config-r)# main-cpu
Router(config-r-mc)# auto-sync standard
Router(config-r-mc)# auto-sync bootvar
```

```
Router(config-r-mc)# end
Router# copy running-config startup-config
```

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

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

```
Router(config)# redundancy
Router(config-r)# main-cpu
Router(config-r-mc)# no auto-sync standard
Router(config-r-mc)# auto-sync config-register
Router(config-r-mc)# end
Router# copy running-config startup-config
```

Displaying the Supervisor Engine Redundancy

To display both supervisor engines, perform this task:

Command	Purpose
`Router# show module all`	Displays the redundancy configuration.

This example shows how to display the supervisor engine redundancy configuration:

```
Router# show module all
Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------
  1    2  Catalyst 6000 supervisor 2 (Active)    WS-X6K-SUP2-2GE    SAD0620046D
  5   48  48 port 10/100 mb RJ-45 ethernet       WS-X6248-RJ-45     SAD03181291

Mod MAC addresses                     Hw     Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  1  0001.c9db.3788 to 0001.c9db.3789   3.7   6.1(3)       7.5(0.6)HUB6 Ok
  5  0050.f0ac.3054 to 0050.f0ac.3083   1.0   4.2(0.24)VAI 7.5(0.6)HUB6 Ok

Mod Sub-Module                  Model           Serial          Hw     Status
--- --------------------------- --------------- --------------- ------- -------
  1 Policy Feature Card 2       WS-F6K-PFC2     SAD06200415      3.2    Ok
  1 Cat6k MSFC 2 daughterboard  WS-F6K-MSFC2    SAD06210067      2.3    Ok

Mod Online Diag Status
--- -------------------
  1 Pass
  5 Pass
Router#
```

Copying Files to the Redundant Supervisor Engine

Use the following command to copy a file to the **slot0:** device on a redundant supervisor engine:

```
Router# copy source_device:source_filename slaveslot0:target_filename
```

Use the following command to copy a file to the **bootflash:** device on a redundant supervisor engine:

```
Router# copy source_device:source_filename slavesup-bootflash:target_filename
```

Use the following command to copy a file to the **bootflash:** device on a redundant MSFC:

```
Router# copy source_device:source_filename slavebootflash:target_filename
```