



Route-Switch-Controller Handover Redundancy on the Cisco AS5850

Last Updated: December 13, 2011

Feature History

| Release | Modification |
|------------|---|
| 12.2(2)XB1 | This feature was introduced on the Cisco AS5850. |
| 12.2(11)T | This feature was integrated into Cisco IOS Release 12.2(11)T and support was added for the Cisco AS5850 platform. |

This document describes the Route-Switch-Controller Handover Redundancy feature on the Cisco AS5850. It includes the following sections:

- [Finding Feature Information, page 1](#)
- [Feature Overview, page 2](#)
- [Supported Platforms, page 4](#)
- [Supported Standards MIBs and RFCs, page 4](#)
- [Prerequisites, page 5](#)
- [Configuration Tasks, page 5](#)
- [Configuration Examples, page 13](#)
- [Command Reference, page 19](#)
- [Glossary, page 19](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.



Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Feature Overview

Route-Switch-Controller Handover Redundancy on the Cisco AS5850, with its provision of handover-split mode, provides the first phase of high availability to the Cisco AS5850 platform.

If your gateway contains two route-switch-controller (RSC) cards, you can configure your Cisco AS5850 into either of two split modes: classic split or handover split.

Classic-Split Mode

Classic-split (the default) mode maximizes system throughput by splitting slots between two RSCs. Each RSC controls a certain set of slots (slots 0-5 are owned by the RSC in slot 6 and slots 8-13 are owned by the RSC in slot 7), and operates as though slots other than those that it controls contain no cards because those cards are controlled by the other RSC. Configuration on each RSC affects only the slots owned by that RSC. Calls on a failed RSC are lost, but calls on the functioning RSC continue normally. Operating a Cisco AS5850 in classic-split mode is the same as having two Cisco AS5850s, each with a separate set of cards.

Handover-Split Mode

Handover-split mode maximizes system availability by allowing an RSC to automatically take control of the slots, cards, and calls of the other RSC should that other RSC fail. Each RSC is configured identically as appropriate for the full set of cards. During normal operation, both RSCs are active, handling their own slots, cards, and calls just as in classic-split mode. Should an RSC fail, the other RSC takes over control of the failed RSC's slots, goes into extraload state, restarts the failed RSC's cards, and handles newly arrived calls on those cards--although calls on the failed RSC are lost at the moment of failure. The failed RSC, should it recover or be restarted, remains in standby state until you instruct the active RSC to hand back its newly acquired slots to the standby RSC. This is, in effect, split dial shelf with handover capability.

Alternately, to use system resources most efficiently, you can operate with one of the two RSCs initially and intentionally in extraload state. In this configuration, RSCA initially controls all slots in the chassis and RSCB is in standby mode, ready to take over should RSCA fail. This allows you to overcome the limits of normal classic-split mode in which, because only six slots are available per RSC, an optimal combination of trunk and DSP cards is difficult to achieve. For more information on performance loads, see the [Restrictions, page 3](#) section.

- [Benefits, page 2](#)
- [Restrictions, page 3](#)
- [Related Features and Technologies, page 3](#)
- [Related Documents, page 3](#)

Benefits

High Availability

RSC Handover Redundancy for the Cisco AS5850, enabled in handover-split mode, eliminates any single point of failure, subsequent downtime, and required user intervention to resolve unrecoverable hardware faults. This improves service availability and reduces both service-affecting time and service interruption.

Restrictions

RSC Card Requirements

You must have two RSC cards installed in your Cisco AS5850 system chassis.

Performance Load and Possible Trunk-Card and Port-Density Limitations

The number of CT3, T1, or E1 trunk cards that your system can support depends on the split mode in which it is configured to operate. In classic-split mode, an RSC card needs to handle the trunk cards in its own half only. In handover-split mode, an RSC card needs to be able to handle the full load of trunk cards across the entire chassis. In either case, the number of trunk cards allowed should not exceed the performance load of the handling RSC card.

For further information about performance loads, refer to the tables on Cisco AS5850 universal port capacities in the overview chapter of *Cisco AS5850 Universal Gateway Operations, Administration, Maintenance, and Provisioning Guide*

Throughput Versus Availability

You must choose between maximal throughput and maximal availability:

- Disabling the handover redundancy by configuring classic-split mode provides maximal throughput, at the expense of availability.
- Enabling handover redundancy by configuring handover-split mode provides maximal availability, at the expense of throughput.

Dropped Calls

Calls on a failed RSC, regardless of mode, are lost at the moment of failure.

Fixed Slot Assignments

Slot assignments are fixed and cannot be changed except by a system in handover-split mode during handover. Slots 0-5 are owned by the RSC in slot 6, and slots 8-13 are owned by the RSC in slot 7.

Related Features and Technologies

Router-Shelf Redundancy

The Router-Shelf Redundancy feature that is available on the Cisco AS5800 is similar to RSC Handover Redundancy on the Cisco AS5850.

Related Documents

- *Cisco AS5850 Operations, Administration, Maintenance, and Provisioning Guide*, chapter on provisioning, available from the Cisco AS5850 Product Documentation website

Supported Platforms

- Cisco AS5850 universal gateway

Table 1 *Cisco IOS Release and Platform Support for this Feature*

| Platform | 12.2(2)XB1 | 12.2(11)T |
|--------------|------------|-----------|
| Cisco AS5850 | X | X |

Determining Platform Support Through Cisco Feature Navigator

Cisco IOS software is packaged in feature sets that support specific platforms. To get updated information regarding platform support for this feature, access Cisco Feature Navigator. Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.

Cisco Feature Navigator is a web-based tool that enables you to determine which Cisco IOS software images support a specific set of features and which features are supported in a specific Cisco IOS image. You can search by feature or release. Under the release section, you can compare releases side by side to display both the features unique to each software release and the features in common.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/fn>. An account on Cisco.com is not required.

Availability of Cisco IOS Software Images

Platform support for particular Cisco IOS software releases is dependent on the availability of the software images for those platforms. Software images for some platforms may be deferred, delayed, or changed without prior notice. For updated information about platform support and availability of software images for each Cisco IOS software release, refer to the online release notes or, if supported, Cisco Feature Navigator.

Supported Standards MIBs and RFCs

Standards

None

MIBs

- CISCO-RF-MIB

To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at the following URL: <http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

RFCs

None

Prerequisites

RSC Cards

Be sure that you have two RSC cards installed in your Cisco AS5850, one in slot 6 and one in slot 7.

Trunk Cards

If you have CT3, T1, or E1 trunk cards in your Cisco AS5850, be sure that you have a supportable number. For more information on performance loads, see the [Restrictions](#), page 3.

Cisco IOS Image

- For classic-split mode, it is advisable, although not mandatory, to configure each RSC with the same Cisco IOS image.
- For handover-split mode, it is mandatory that you configure each RSC with the same Cisco IOS image and the same configuration except for the IP address on egress interfaces. Your Cisco IOS image must support redundancy (Cisco IOS Release 12.2(2)XB, Cisco IOS Release 12.2(11)T, or later releases).

You must replicate the startup configuration for all line cards in the system in both RSCs' saved configurations to ensure correct operation after a handover.

- You can download software configurations to your Cisco AS5850 using Simple Network Management Protocol (SNMP) or a Telnet connection. To learn how to upgrade your Cisco IOS image, go to the Cisco.com website for Cisco AS5850 Product Documentation , locate the *Cisco AS5850 Universal Gateway Operations, Administration, Maintenance, and Provisioning Guide* , and consult the chapter on provisioning.

Configuration Tasks

See the following sections for configuration tasks for this feature. Each task in the list is identified as either required or optional. Note that you must configure and verify either classic-split mode (the default) or handover-split mode.

- [Configuring Classic-Split Mode](#), page 6
- [Verifying Classic-Split Mode](#), page 7
- [Configuring Handover-Split Mode](#), page 8
- [Configuring Handover-Split Mode](#), page 8
- [Verifying Handover-Split Mode](#), page 9
- [Troubleshooting Tips](#), page 12
- [Monitoring and Maintaining Handover Redundancy](#), page 12

Configuring Classic-Split Mode

SUMMARY STEPS

1. Router# **configuration terminal**
2. Router(config)# **redundancy**
3. Router(config-red)# **mode classic-split**
4. Router# **copy running-config startup-config**
5. Router# **reload**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | Router# configuration terminal | Enters configuration mode. |
| Step 2 | Router(config)# redundancy | Enters configuration-redundancy mode. |
| Step 3 | Router(config-red)# mode classic-split | Selects classic-split (the default) mode. |
| Step 4 | Router# copy running-config startup-config | Copies the running configuration into the startup configuration. |
| Step 5 | Router# reload | Reloads the RSC. |

Connect to each RSC in turn and enter these commands.



Note

Classic-split mode is the default mode. If you do not perform these steps, your system defaults to this mode.



Note

These steps simply configure the system to classic-split mode. You must also configure each of the cards manually.

A classic-split system appears to SNMP management applications as two separate Cisco AS5850s. You must conduct a console session for each RSC (two console sessions) to configure your splits. The system controller manages a classic-split configuration as two separate Cisco AS5850 universal gateways.

Network management systems (NMSs) such as the Cisco Universal Gateway Manager (Cisco UGM) are available that provide a single system view of multiple points of presence (POPs) as they monitor performance and log accounting data. An NMS has a graphical user interface (GUI); runs on a UNIX SPARC station; and includes a database-management system, polling engine, trap management, and map integration. The NMS can be installed at a remote facility so that you can access multiple systems through a console port or Web interface.

In classic-split mode, it is desirable--and, with an NMS, essential--to use four unique IDs, one for each RSC and one for each set of slots. In some cases, however, it is sufficient to use the same ID for the two RSCs.

Verifying Classic-Split Mode

In classic-split mode, most **show** commands (with exceptions noted below) display information for only those slots owned by the RSC; they look and behave as they would if there were no cards in the slots that the RSC does not own. To see **show** command information for a slot, you must connect to the RSC that owns that slot.

Enter any of the following commands, in any order.

- To display information about all slots, regardless of ownership, enter the **show context all** command in EXEC mode.
- To display information about owned slots, enter the **show context** command in EXEC mode without the **all** option.
- To display additional relevant output, including whether an RSC is running in classic-split mode and, if so, which slots it owns, enter the **show chassis** command in EXEC mode.

```
RouterA# show chassis
System is in classic-split mode, RSC in slot 6.
Slots owned: 0 1 2 3 4 5
Slots configured: 0 1 2 3 4 5
Slots owned by other: 8 9 10 11 12 13
Slot   Board   CPU      DRAM          I/O Memory   State      Elapsed
      Type   Util    Total (free)  Total (free)
1     UP324   0%/0%   60159040( 51%) 67108864( 73%) Up         6d01h
2     UP324   0%/0%   60159040( 56%) 67108864( 73%) Up         6d01h
3     UP324   0%/0%   60159040( 56%) 67108864( 73%) Up         6d01h
4     CT3_UP216 0%/0%   60159040( 50%) 67108864( 72%) Up         6d01h
```

System set for auto boot

```
RouterB# show chassis
System is in classic-split mode, RSC in slot 7.
Slots owned: 8 9 10 11 12 13
Slots configured: 8 9 10 11 12 13
Slots owned by other: 0 1 2 3 4 5
Slot   Board   CPU      DRAM          I/O Memory   State      Elapsed
      Type   Util    Total (free)  Total (free)
9     CT3_UP216 0%/0%   60159040( 65%) 67108864( 72%) Up         00:21:46
10    UP324   0%/0%   60159040( 62%) 67108864( 73%) Up         00:21:48
11    UP324   0%/0%   60159040( 62%) 67108864( 73%) Up         00:21:49
```

System set for auto boot

- To display all configured clock sources, even those from non-owned cards, enter the **show chassis clocks** command in EXEC mode. Only one RSC can provide the master clock, and it may need to have backup clock sources configured from all cards present, regardless of ownership.

```
RouterA# show chassis clocks
Primary Clock:
-----
Slot 6:
System primary is Slot: 4 Port: 1 of priority 10
TDM Bus Master Clock Generator State = NORMAL
Backup clocks:
Source Slot Port DS3-Port Priority Status State
-----
Trunk 9 1 0 8 Good Configured
Trunk 4 21 0 498 Good Default
Trunk 9 21 0 503 Good Default
Status of trunk clocks:
-----
Ds3
Slot Port Type 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1
4 0 T3 B B B B B B G G G G G G G G G G G G G G G G G G G G G G G G
9 0 T3 B B B B B B G G G G G G G G G G G G G G G G G G G G G G G G
```

Configuring Handover-Split Mode

Perform the following steps on both RSCs so that all cards are configured on both RSCs.

SUMMARY STEPS

1. Router# **configuration terminal**
2. Router(config)# **redundancy**
3. Router(config-red)# **mode handover-split**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|---------------------------------------|
| Step 1 | Router# configuration terminal | Enters configuration mode. |
| Step 2 | Router(config)# redundancy | Enters redundancy configuration mode. |
| Step 3 | Router(config-red)# mode handover-split | Selects handover-split mode. |

Configuring Handover-Split Mode

Connect to each RSC in turn, change the running configuration so that all cards are configured on this RSC, and perform the following steps.

SUMMARY STEPS

1. Router# **copy running-config startup-config**
2. Router# **dir[/all][filesystem:][file-url]**
3. Router# **reload**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|---|
| Step 1 | Router# copy running-config startup-config | Copies the running configuration into the startup configuration. |
| Step 2 | Router# dir[/all][filesystem:][file-url] | Displays a list of files on a file system. Use to verify that the new image is loaded to system Flash memory or the FTP server. |
| Step 3 | Router# reload | Reloads the RSC. |

The net result, when you are done, is that all cards are configured on each RSC.



Note

These steps simply configure the system to handover-split mode. You must also manually configure each card on both RSCs.

**Note**

By default, a single RSC can handle only up to two CT3 cards. You can release this restriction by using the **no dial-config-guidelines** command. For more information on performance loads, see the [Restrictions, page 3](#).

Verifying Handover-Split Mode

Enter any of the following commands, in any order.

- To indicate whether handover is enabled and whether this RSC is active or standby, enter the **show redundancy states** command in EXEC mode.

```
RouterA# show redundancy states
my state = 13 -ACTIVE
peer state = 13 -ACTIVE
Mode = Duplex
Unit = Preferred Primary
Unit ID = 6
Redundancy Mode = Handover-split: If one RSC fails, the peer RSC will take over the
feature boards
Maintenance Mode = Disabled
Manual Swact = Enabled
Communications = Up
client count = 3
client_notification_TMR = 30000 milliseconds
keep_alive TMR = 4000 milliseconds
keep_alive count = 0
keep_alive threshold = 7
RF debug mask = 0x0
```

- To display logged handover event, enter the **show redundancy history** command in EXEC mode.

```
RouterA# show redundancy history
Redundancy Facility Event Log:
00:00:00 client added: RF_INTERNAL_MSG(0) seq=0
00:00:00 client added: RF_LAST_CLIENT(65000) seq=65000
00:00:09 client added: Rsc split dshelf client(19) seq=800
00:00:09 *my state = INITIALIZATION(2) *peer state = DISABLED(1)
00:00:09 RF_PROG_INITIALIZATION(100) RF_INTERNAL_MSG(0) op=0 rc=11
00:00:09 RF_PROG_INITIALIZATION(100) Rsc split dshelf client(19) op=0 rc=11
00:00:09 RF_PROG_INITIALIZATION(100) RF_LAST_CLIENT(65000) op=0 rc=11
00:00:09 *my state = NEGOTIATION(3) peer state = DISABLED(1)
00:00:11 RF_STATUS_PEER_PRESENCE(400) op=1
00:00:11 RF_STATUS_PEER_PRESENCE(400) Rsc split dshelf client(19) op=1
00:00:11 RF_STATUS_PEER_COMM(401) op=1
00:00:11 RF_STATUS_PEER_COMM(401) Rsc split dshelf client(19) op=1
00:00:11 my state = NEGOTIATION(3) *peer state = UNKNOWN(0)
00:00:15 RF_EVENT_CLIENT_PROGRESSION(503) RF_LAST_CLIENT(65000) op=1
00:00:15 RF_PROG_PLATFORM_SYNC(300) RF_LAST_CLIENT(65000) op=1 rc=0
00:00:15 RF_EVENT_CLIENT_PROGRESSION(503) RF_LAST_CLIENT(65000) op=1 rc=0
00:00:17 RF_STATUS_REDUNDANCY_MODE_CHANGE(405) Rsc split dshelf client(19) op=3
00:00:17 RF_EVENT_GO_STANDBY(512) op=0
00:00:17 *my state = STANDBY COLD(4) peer state = UNKNOWN(0)
00:00:17 RF_PROG_STANDBY_COLD(101) RF_INTERNAL_MSG(0) op=0 rc=11
00:00:17 RF_PROG_STANDBY_COLD(101) Rsc split dshelf client(19) op=0 rc=11
00:00:17 RF_PROG_STANDBY_COLD(101) RF_LAST_CLIENT(65000) op=0 rc=11
00:00:19 my state = STANDBY COLD(4) *peer state = ACTIVE_EXTRALOAD(14)
00:00:51 Configuration parsing complete
00:00:53 System initialization complete
00:01:11 RF_STATUS_PEER_PRESENCE(400) op=0
00:01:11 RF_STATUS_PEER_PRESENCE(400) Rsc split dshelf client(19) op=0
00:01:11 my state = STANDBY COLD(4) *peer state = DISABLED(1)
00:01:11 Reloading peer (peer presence lost)
00:01:11 *my state = ACTIVE-FAST(9) peer state = DISABLED(1)
00:01:11 RF_STATUS_MAINTENANCE_ENABLE(403) Rsc split dshelf client(19) op=0
```

```

00:01:11 RF_PROG_ACTIVE_FAST(200) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_FAST(200) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_FAST(200) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE-DRAIN(10) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE_DRAIN(201) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_DRAIN(201) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_DRAIN(201) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE_PRECONFIG(11) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE_PRECONFIG(202) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_PRECONFIG(202) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_PRECONFIG(202) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE_POSTCONFIG(12) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE_POSTCONFIG(203) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_POSTCONFIG(203) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE_POSTCONFIG(203) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 *my state = ACTIVE(13) peer state = DISABLED(1)
00:01:11 RF_PROG_ACTIVE(204) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_ACTIVE(204) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_ACTIVE(204) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 RF_STATUS_PEER_COMM(401) op=0
00:01:11 RF_STATUS_PEER_COMM(401) Rsc split dshelf client(19) op=0
00:01:11 Reloading peer (communication down)
00:01:11 RF_EVENT_GO_ACTIVE_EXTRALOAD(513) RF_INTERNAL_MSG(0) op=0
00:01:11 RF_PROG_EXTRALOAD(301) RF_INTERNAL_MSG(0) op=0 rc=11
00:01:11 RF_PROG_EXTRALOAD(301) Rsc split dshelf client(19) op=0 rc=11
00:01:11 RF_PROG_EXTRALOAD(301) RF_LAST_CLIENT(65000) op=0 rc=11
00:01:11 RF_EVENT_GO_ACTIVE_EXTRALOAD(513) RF_INTERNAL_MSG(0) op=0
00:03:02 RF_STATUS_PEER_PRESENCE(400) op=1
00:03:02 RF_STATUS_PEER_PRESENCE(400) Rsc split dshelf client(19) op=1
00:03:02 RF_STATUS_PEER_COMM(401) op=1
00:03:02 RF_STATUS_PEER_COMM(401) Rsc split dshelf client(19) op=1
00:03:02 *my state = ACTIVE_EXTRALOAD(14) *peer state = UNKNOWN(0)
00:03:02 RF_PROG_PLATFORM_SYNC(300) RF_INTERNAL_MSG(0) op=0 rc=11
00:03:02 RF_PROG_PLATFORM_SYNC(300) Rsc split dshelf client(19) op=0 rc=11
00:03:02 RF_PROG_PLATFORM_SYNC(300) RF_LAST_CLIENT(65000) op=0 rc=0
00:03:02 RF_EVENT_CLIENT_PROGRESSION(503) RF_LAST_CLIENT(65000) op=1 rc=0
00:03:02 my state = ACTIVE_EXTRALOAD(14) *peer state = NEGOTIATION(3)
00:03:02 RF_EVENT_PEER_PROG_DONE(506) RF_LAST_CLIENT(65000) op=300
00:03:06 my state = ACTIVE_EXTRALOAD(14) *peer state = STANDBY COLD(4)
6d01h RF_EVENT_GO_ACTIVE_HANDBACK(514) RF_INTERNAL_MSG(0) op=0
6d01h RF_PROG_HANDBACK(302) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_HANDBACK(302) Rsc split dshelf client(19) op=0 rc=0
6d01h RF_EVENT_CLIENT_PROGRESSION(503) Rsc split dshelf client(19) op=1 rc=0
6d01h RF_EVENT_GO_ACTIVE(511) op=0
6d01h Reloading peer (this unit becoming active)
6d01h *my state = ACTIVE-FAST(9) peer state = STANDBY COLD(4)
6d01h RF_STATUS_MAINTENANCE_ENABLE(403) Rsc split dshelf client(19) op=0
6d01h RF_PROG_ACTIVE_FAST(200) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_FAST(200) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_FAST(200) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE-DRAIN(10) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE_DRAIN(201) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_DRAIN(201) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_DRAIN(201) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE_PRECONFIG(11) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE_PRECONFIG(202) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_PRECONFIG(202) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_PRECONFIG(202) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE_POSTCONFIG(12) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE_POSTCONFIG(203) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE_POSTCONFIG(203) Rsc split dshelf client(19) op=0 rc=11
6d01h RF_PROG_ACTIVE_POSTCONFIG(203) RF_LAST_CLIENT(65000) op=0 rc=11
6d01h *my state = ACTIVE(13) peer state = STANDBY COLD(4)
6d01h RF_PROG_ACTIVE(204) RF_INTERNAL_MSG(0) op=0 rc=11
6d01h RF_PROG_ACTIVE(204) Rsc split dshelf client(19) op=0 rc=0
6d01h RF_EVENT_CLIENT_PROGRESSION(503) Rsc split dshelf client(19) op=1 rc=0
6d01h my state = ACTIVE(13) *peer state = ACTIVE(13)
6d01h my state = ACTIVE(13) *peer state = UNKNOWN(0)
6d01h Reloading peer (notification timeout)
6d01h my state = ACTIVE(13) *peer state = ACTIVE(13)
6d01h RF_STATUS_REDUNDANCY_MODE_CHANGE(405) Rsc split dshelf client(19) op=1
6d01h RF_EVENT_GO_ACTIVE(511) op=0

```

```
6d01h RF_STATUS_REDUNDANCY_MODE_CHANGE(405) Rsc split dshelf client(19) op=3
6d01h RF_EVENT_GO_ACTIVE(511) op=0
```

- To display details of any pending handover, enter the **show redundancy handover** command in EXEC mode.

```
RouterA# show redundancy handover
No handover pending
```

- To display up to 256 relevant debug entries, enter the **show redundancy debug-log** command in EXEC mode.
- To display additional relevant output, enter the **show chassis** command in EXEC mode. In handover-split mode, this command shows the RSC to be configured with all slots of the entire chassis, regardless of whether the RSC owns the slots or not. Slots owned by the peer RSC are shown to be in the ignore state, properly configured and ready to go.

The following example shows output for two RSCs in normal-load state.

```
RouterA# show chassis
System is in handover-split mode, RSC in slot 6.
  Slots owned: 0 1 2 3 4 5
  Slots configured: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots owned by other: 8 9 10 11 12 13
Slot  Board      CPU      DRAM      I/O Memory  State      Elapsed
   Type      Util      Total (free)  Total (free)
1    UP324     17%/17%  60159040( 50%) 67108864( 73%) Up          6d01h
2    UP324     1%/0%    60159040( 56%) 67108864( 73%) Up          6d01h
3    UP324     0%/0%    60159040( 56%) 67108864( 73%) Up          6d01h
4    CT3_UP216 1%/0%    60159040( 49%) 67108864( 72%) Up          6d01h
9    CT3_UP216      60159040( 0%) 67108864( 0%) Ignore     00:00:20
10   UP324     60159040( 0%) 67108864( 0%) Ignore     00:00:19
11   UP324     60159040( 0%) 67108864( 0%) Ignore     00:00:18
System set for auto boot
RouterB# show chassis
System is in handover-split mode, RSC in slot 7.
  Slots owned: 8 9 10 11 12 13
  Slots configured: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots owned by other: 0 1 2 3 4 5
Slot  Board      CPU      DRAM      I/O Memory  State      Elapsed
   Type      Util      Total (free)  Total (free)
1    UP324           0( 0%)      0( 0%) Ignore     00:00:38
2    UP324           0( 0%)      0( 0%) Ignore     00:00:37
3    UP324           0( 0%)      0( 0%) Ignore     00:00:36
4    CT3_UP216 0( 0%)      0( 0%)      0( 0%) Ignore     00:00:35
9    CT3_UP216 0%/0%    60159040( 65%) 67108864( 72%) Up          00:23:14
10   UP324     0%/0%    60159040( 62%) 67108864( 73%) Up          00:23:16
11   UP324     0%/0%    60159040( 62%) 67108864( 73%) Up          00:23:17
System set for auto boot
```

The following example shows output for one RSC in extraload state.

```
RouterA# show chassis
System is in handover-split mode, RSC in slot 6.
  Slots owned: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots configured: 0 1 2 3 4 5 8 9 10 11 12 13
  Slots owned by other: none
Slot  Board      CPU      DRAM      I/O Memory  State      Elapsed
   Type      Util      Total (free)  Total (free)
1    UP324     0%/0%    60159040( 50%) 67108864( 73%) Up          6d02h
2    UP324     1%/0%    60159040( 56%) 67108864( 73%) Up          6d02h
3    UP324     0%/0%    60159040( 56%) 67108864( 73%) Up          6d02h
4    CT3_UP216 6%/5%    60159040( 49%) 67108864( 72%) Up          6d02h
9    CT3_UP216 5%/4%    60159040( 56%) 67108864( 72%) Up          00:10:29
10   UP324     20%/20%  60159040( 56%) 67108864( 73%) Up          00:10:30
11   UP324     0%/0%    60159040( 56%) 67108864( 73%) Up          00:10:30
System set for auto boot
```

Troubleshooting Tips

| Command | Purpose |
|--|---|
| <pre>Router# debug redundancy as5850</pre> | <p>Enables or disables redundancy-related debug options (hardware lines, master RSC, FSM events, mode, RF client). Use to view specific relevant debug options. All debug entries continue to be logged even if you disable an option here, and you can always use the show redundancy debug-log command to view them.</p> |

Monitoring and Maintaining Handover Redundancy

| Command | Purpose |
|--|--|
| <pre>Router# redundancy handover cancel peer-resources shelf-resources [busyout-period mins at hh:mm day month year]</pre> | <p>Specifies or cancels handover of slots between RSCs. Use during Cisco IOS image upgrades and to return control of slots to an RSC that failed but is now back in service. Specify handover of slots belonging either to the peer RSC (peer-resources) or to the RSC on which the command is run (shelf-resources). Optionally, specify either or both of the following: length of time for which and exact time at which slots should be busied out before handover.</p> <p>Note The shelf-resources option causes the RSC to reload.</p> |



Note

You can detect if an RSC is in extraload with control of the entire chassis resources by observing that the master LED for that RSC is on. You can also detect this state by using the **show redundancy states** command.

The following example shows two instances of handover scheduling, verification, cancellation, and verification of cancellation:

```
RouterA# redundancy handover shelf-resources busyout-period 10 at 16:15 5 Sept 2001
Newly entered handover schedule:
Busyout period at 16:15:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:25:00 PST Wed Sep 5 2001
Clear calls, handover and reload as specified above?[confirm]
RouterA# show redundancy handover
Busyout period at 16:15:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:25:00 PST Wed Sep 5 2001
RouterA# redundancy handover cancel
Scheduled handover is cancelled
RSC-Slot6# show redundancy handover
No handover pending
RouterA# redundancy handover peer-resources busyout-period 10 at 16:37 5 Sep 2001
Newly entered handover schedule:
Busyout period at 16:37:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:47:00 PST Wed Sep 5 2001
```

```

Clear calls and handover as specified above?[confirm]
RouterA# show redundancy handover
Busyout period at 16:37:00 PST Wed Sep 5 2001 for a duration of 10 minute(s)
Handover pending at 16:47:00 PST Wed Sep 5 2001
RouterA# redundancy handover cancel
Scheduled handover is cancelled
RouterA# show redundancy handover
No handover pending

```

Configuration Examples

The following example shows a startup configuration that supports redundancy. Note, in the sections on resource-pool range and controller numbers, that every card in the chassis is configured.

```

RouterA# show startup-config
version 12.2
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service compress-config
!
hostname RouterA
!
redundancy
 mode handover-split
aaa new-model
!
!
aaa group server tacacs+ redline2
!
aaa group server radius RADIUS-GROUP
 server 172.22.51.9 auth-port 1645 acct-port 1646
!
aaa authentication login CONSOLE none
aaa authentication login VTY none
aaa authentication ppp default group RADIUS-GROUP
aaa authentication ppp RADIUS-LIST group RADIUS-GROUP
aaa authorization exec CONSOLE none
aaa authorization exec RADIUS-LIST group RADIUS-GROUP
aaa authorization network default group RADIUS-GROUP if-authenticated
aaa authorization network RADIUS-LIST group RADIUS-GROUP if-authenticated
aaa accounting network default start-stop group RADIUS-GROUP
aaa nas port extended
aaa session-id common
enable password xxx
!
username RouterB password 0 xxx
username 54006
username 54006_1 password 0 xxx
username RouterA password 0 xxx
username 54006_d_119 password 0 xxx
!
resource-pool enable
!
resource-pool group resource group1
 range port 1/0 1/323
 range port 4/20 4/30
!
resource-pool group resource group2
 range port 9/0 9/215
 range port 10/0 10/120
!
resource-pool group resource digital_group_6
 range limit 207
!
resource-pool group resource digital_group
 range limit 116
!

```

```

resource-pool group resource vpdn_dig
  range limit 92
!
resource-pool profile customer 54006_customer
  limit base-size all
  limit overflow-size 0
  resource group1 speech
  dnis group 54006_dnis
!
resource-pool profile customer 54007_customer
  limit base-size all
  limit overflow-size 0
  resource group2 speech
  dnis group 54007_dnis
!
resource-pool profile customer 54006_customer_sync
  limit base-size all
  limit overflow-size 0
  resource digital_group_6 digital
  dnis group 54006_sync_dnis
!
resource-pool profile customer 54007_sync
  limit base-size all
  limit overflow-size 0
  resource digital_group digital
  dnis group 54007_sync_dnis
!
resource-pool profile customer 54007_sync_vpdn
  limit base-size all
  limit overflow-size 0
  resource vpdn_dig digital
  dnis group 54007_sync_vpdn_dnis
clock timezone PST -7
dial-tdm-clock priority 8 trunk-slot 9 ds3-port 0 port 1
dial-tdm-clock priority 10 trunk-slot 4 ds3-port 0 port 1
spe country tl-default
!
spe link-info poll voice 5
!
ip subnet-zero
ip cef distributed
ip ftp source-interface FastEthernet6/0
ip ftp username root
ip ftp password xxxxx
no ip domain-lookup
!
vpdn enable
!
vpdn-group 1
  request-dialin
  protocol l2f
  source-ip 30.0.0.1
!
chat-script dial "" "ATZ" OK "ATDT\T" TIMEOUT 60 CONNECT
isdn switch-type primary-5ess
!
controller T3 4/0
  framing c-bit
  cablelength 224
  t1 1-28 controller
!
controller T1 4/0:1
  framing esf
  pri-group timeslots 1-24
!
controller T1 4/0:2
  framing esf
  pri-group timeslots 1-24
!
controller T1 4/0:3
  framing esf
  pri-group timeslots 1-24
!

```

```
.
.
.
controller T1 4/0:28
 shutdown
 framing esf
 pri-group timeslots 1-24
!
controller T3 9/0
 framing c-bit
 cablelength 224
 t1 1-28 controller
!
controller T1 9/0:1
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
controller T1 9/0:2
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
controller T1 9/0:3
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
.
.
.
controller T1 9/0:12
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
controller T1 9/0:13
 framing esf
 pri-group timeslots 1-24
!
.
.
.
controller T1 9/0:21
 framing esf
 pri-group timeslots 1-24
!
controller T1 9/0:22
 shutdown
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
.
.
.
controller T1 9/0:28
 shutdown
 framing esf
 ds0-group 0 timeslots 1-24 type e&m-fgb dtmf dnis
!
!
interface Loopback0
 ip address 111.111.111.11 255.255.255.0
 no ip mroute-cache
!
interface Serial4/0:1:23
 no ip address
 encapsulation ppp
 ip mroute-cache
 isdn switch-type primary-5ess
 isdn incoming-voice modem
!
interface Serial4/0:2:23
 no ip address
 encapsulation ppp
 ip mroute-cache
```

```

    isdn switch-type primary-5ess
    isdn incoming-voice modem
    !
interface Serial4/0:3:23
no ip address
encapsulation ppp
ip mroute-cache
isdn switch-type primary-5ess
isdn incoming-voice modem
!
.
.
interface Serial4/0:10:23
no ip address
encapsulation ppp
ip mroute-cache
isdn switch-type primary-5ess
isdn incoming-voice modem
!
interface Serial4/0:11:23
no ip address
encapsulation ppp
ip mroute-cache
isdn switch-type primary-5ess
isdn incoming-voice modem
!
interface Serial9/0:21:23
ip unnumbered Loopback0
encapsulation ppp
ip mroute-cache
dialer rotary-group 1
dialer-group 1
isdn switch-type primary-5ess
!
interface Group-Async0
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
async default routing
async mode dedicated
peer default ip address pool KRAMER
ppp max-bad-auth 3
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterB
ppp chap password 7 xxxxx
group-range 9/00 11/323
!
interface Group-Async1
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
async default routing
async mode dedicated
peer default ip address pool KRAMER1
ppp max-bad-auth 3
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterA
ppp chap password 7 xxxxx
group-range 1/00 4/215
!
interface Dialer0
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003

```



```
dialer-group 1
peer default ip address pool KRAMER1_d_m
no fair-queue
no cdp enable
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterA
ppp chap password 7 xxxxx
ppp multilink
!
interface Dialer1
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
peer default ip address pool KRAMER_d
no cdp enable
ppp max-bad-auth 3
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterB
ppp chap password 7 xxxxx
!
interface Dialer2
ip unnumbered Loopback0
encapsulation ppp
dialer in-band
dialer idle-timeout 36000 either
dialer string 6003
dialer-group 1
peer default ip address pool KRAMER1_d
no fair-queue
no cdp enable
ppp authentication chap pap callin RADIUS_LIST
ppp chap hostname RouterA
ppp chap password 7 xxxxx
!
interface Dialer5
no ip address
no cdp enable
!
interface Dialer6
no ip address
no cdp enable
!
interface Dialer7
no ip address
no cdp enable
!
.
.
.
interface Dialer26
no ip address
no cdp enable
!
ip local pool KRAMER1 10.6.1.1 10.6.1.108
ip local pool KRAMER1 10.6.2.1 10.6.2.108
ip local pool KRAMER1 10.6.3.1 10.6.3.60
ip local pool KRAMER 10.7.1.1 10.7.1.108
ip local pool KRAMER 10.7.2.1 10.7.2.108
ip local pool KRAMER 10.7.3.1 10.7.3.60
ip local pool KRAMER1_d 10.6.4.1 10.6.4.115
ip local pool KRAMER_d 10.7.4.1 10.7.4.115
ip local pool KRAMER1_d_m 10.6.4.116 10.6.4.163
ip classless
no ip http server
!
ip radius source-interface FastEthernet6/0
!
dialer dnis group 54006_dnis
number 1002
number 1002100212
```

```

!
dialer dnis group 54007_dnis
  number 38327
!
dialer dnis group 54006_sync_dnis
  number 6666
  number 6600
  number 6666666666
!
dialer dnis group 54007_sync_dnis
  number 7700
  number 7700000000
!
dialer dnis group 54007_sync_vpdn_dnis
  number 7777
  number 7777777777
!
dialer dnis group 54007_vpdn_dnis
  number 38777
dialer-list 1 protocol ip permit
no cdp run
!
tacacs-server host 152.22.51.64
tacacs-server timeout 30
tacacs-server key cisco
snmp-server community public RW
snmp-server enable traps rf
!
radius-server configure-nas
radius-server host 172.22.51.9 auth-port 1645 acct-port 1646 non-standard
radius-server retransmit 3
radius-server attribute nas-port format c
radius-server key lab
call rsvp-sync
!
voice-port 4/0:1:D
!
voice-port 4/0:2:D
!
.
.
.
voice-port 4/0:28:D
!
voice-port 9/0:1:0
!
voice-port 9/0:2:0
!
.
.
.
voice-port 9/0:28:0
!
!
line con 0
  password xxxxxx
  logging synchronous
line aux 0
  logging synchronous
  modem InOut
  transport input all
line vty 0 4
  password xxx
  transport preferred telnet
  transport input telnet
line 1/00 4/215
  modem InOut
  no modem status-poll
  no modem log rs232
  transport preferred none
  transport input all
  autoselect during-login
  autoselect ppp

```

```
line 9/00 9/215
modem InOut
no modem status-poll
no modem log rs232
transport preferred none
transport input all
autoselect during-login
autoselect ppp
line 10/00 11/323
modem InOut
no modem status-poll
no modem log rs232
transport preferred none
transport input all
autoselect during-login
autoselect ppp
!
end
```

Command Reference

The following commands are introduced or modified in the feature or features documented in this module. For information about these commands, see the *Cisco IOS Interface and Hardware Component Command Reference* at http://www.cisco.com/en/US/docs/ios/interface/command/reference/ir_book.html. For information about all Cisco IOS commands, go to the Command Lookup Tool at <http://tools.cisco.com/Support/CLILookup> or to the *Cisco IOS Master Commands List*.

- **debug redundancy as5850**
- **mode (redundancy)**
- **redundancy handover**
- **show redundancy (5850)**
- **show chassis**

Glossary

classic-split mode --Mode in which system throughput is maximized because slots are split between two RSCs.

handover --The ability of one part of a system to take over resources that were managed by another part of the system when the latter part fails.

handover-split mode --Mode in which system availability is maximized because an RSC can automatically take control over the slots, cards, and calls of the other RSC, should that other RSC fail.

RSC --route switch controller. The card that provides switch functions, routing, management control, clock control, and egress ports.

service-affecting time --Amount of time during which the system is unable to take new calls or carry the full number of calls.

service interruption --Event during which an in-progress call is dropped, requiring the user to call back.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks.

Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2011 Cisco Systems, Inc. All rights reserved.