Sun Remote System Control (RSC) User's Guide

Sun™ Enterprise™ 250 Server

Adobe PostScript

Please Recycle

Contents

Preface

Sun Remote System Control (RSC) User's Guide tells you how to configure and use Sun™ Remote System Control software. These instructions are designed for an experienced system administrator with networking knowledge.

Using UNIX Commands

This document does not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

■ *Solaris Handbook for Sun Peripherals*

■ AnswerBook™ online documentation for the Solaris™ software environment

■ Other software documentation that you received with your system

Typographic Conventions

TABLE P–1 Typographic Conventions

Typeface or Symbol	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output.	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output.	`%` **`su`** `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized. Command-line variable; replace with a real name or value.	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be root to do this. To delete a file, type `rm` *filename*.

Shell Prompts

TABLE P–2 Shell Prompts

Shell	Prompt
C shell	*machine_name*`%`
C shell superuser	*machine_name*`#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell superuser	`#`
Remote System Control shell	`rsc>`
OpenBoot PROM shell	`ok`

Related Documentation

TABLE P–3   Related Documentation

Application	Title
Installation	*Sun Remote System Control (RSC) Installation Guide*
PPP configuration	*Configuring and Using Solstice PPP Clients*
Performing diagnostic tests	*SunVTS User's Guide*
	SunVTS Quick Reference Guide
	SunVTS Test Reference Manual
	Solstice SyMON User's Guide
System and network administration	*Solaris System Administrator AnswerBook*
	SPARC: Installing Solaris Software
Using operating system software	*Solaris User's Guide*

Sun Documentation on the Web

The `docs.sun.com`SM web site enables you to access Sun technical documentation on the Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at:

`http://docs.sun.com`

Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at:

`docfeedback@sun.com`

Please include the part number of your document in the subject line of your email.

Sun Remote System Control (RSC) Software

Sun Remote System Control (RSC) is a server management tool that allows you to monitor and control your server over modem lines and over a network. RSC provides remote system administration for geographically distributed or physically inaccessible systems. The RSC software works with the System Service Processor (SSP) on the Sun™ Enterprise™ 250 main logic board. RSC and the SSP support both serial and Ethernet connections to a remote console.

RSC Access

You can access RSC from a workstation running the Solaris, Windows 95, Windows 98, or Windows NT operating environment and Sun's RSC Java™ application, or from an ASCII terminal or device running ASCII terminal emulation software. Figure 1–1 shows remote access paths to RSC.
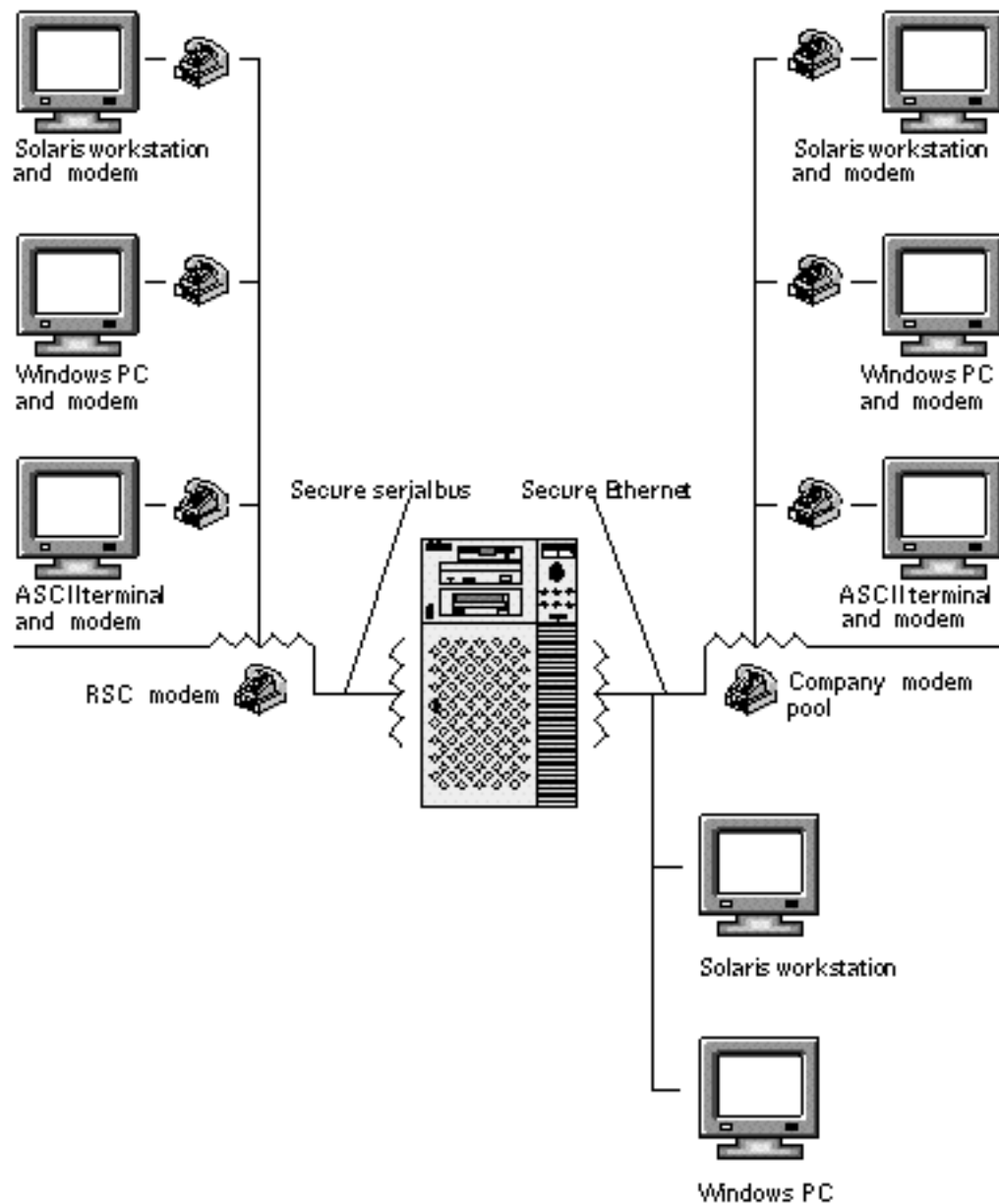
Figure 1–1   RSC Remote Access Paths

Once RSC software is installed and configured to manage your server, you can use it to run diagnostic tests, view diagnostic and error messages, reboot your server, and display environmental status information from a remote console.

The RSC firmware on the SSP runs independently, and uses standby power from the server. Therefore, SSP hardware and RSC software continue to be effective when the server operating system goes offline, and can send notification of hardware failures or other events that may be occurring on your server.

The server can boot and operate normally when RSC software is not enabled, and Sun console features continue to be available on standard RS232 ports.

RSC Features

RSC has the following features:

- Remote system monitoring and error reporting, including output from power-on self-test (POST) and OpenBoot™ Diagnostics (OBDiag)
- Remote server reboot, power-on, and power-off on demand
- Ability to monitor the CPU temperature and fan sensors without being near the managed server, even when the server is offline
- Ability to run diagnostic tests from a remote console
- Remote event notification of server problems
- A detailed log of RSC events
- Remote console functions on both the serial and Ethernet ports

RSC complements existing Sun monitoring and diagnostics tools such as Solstice™ SyMON™, SunVTS™, the `kadb` kernel debugger, OpenBoot PROM (OBP), and OpenBoot Diagnostics (OBDiag). Solstice SyMON operation remains unchanged, and it continues to be the main tool for observing system operation behavior and performance while the server operating system is up and running.

RSC Use

After installing and configuring Sun Remote System Control software on the server and client systems, you use an OpenBoot command and set OpenBoot variables that redirect the console output to RSC.

Part of RSC configuration defines and enables *alert* mechanisms. Alerts provide remote notification of system problems, and can be sent to pagers or an email address; RSC also sends alerts to any clients that are currently logged in to RSC.

RSC sends an alert message whenever any of the following occurs:

- The server system resets.

- Server temperature crosses the lower-fault (high-temperature warning) limit.

- Server temperature crosses the upper-fault (high-temperature shutdown) limit.

- A server redundant power supply fails.

- A power outage occurs at the server site, if an uninterruptible power supply (UPS) is in use and it is configured to send an alert to RSC (see Appendix A).

- RSC receives a server-generated alert.

- The server undergoes a hardware watchdog reset.

- RSC detects five unsuccessful RSC login attempts within five minutes.

Each alert message includes the server name and other important details. RSC configuration controls whether an alert is sent to an email address, to pagers, or to both. In addition, an alert is always sent to any clients currently logged in to RSC accounts for that server. Figure 1–2 shows RSC remote alert paths.
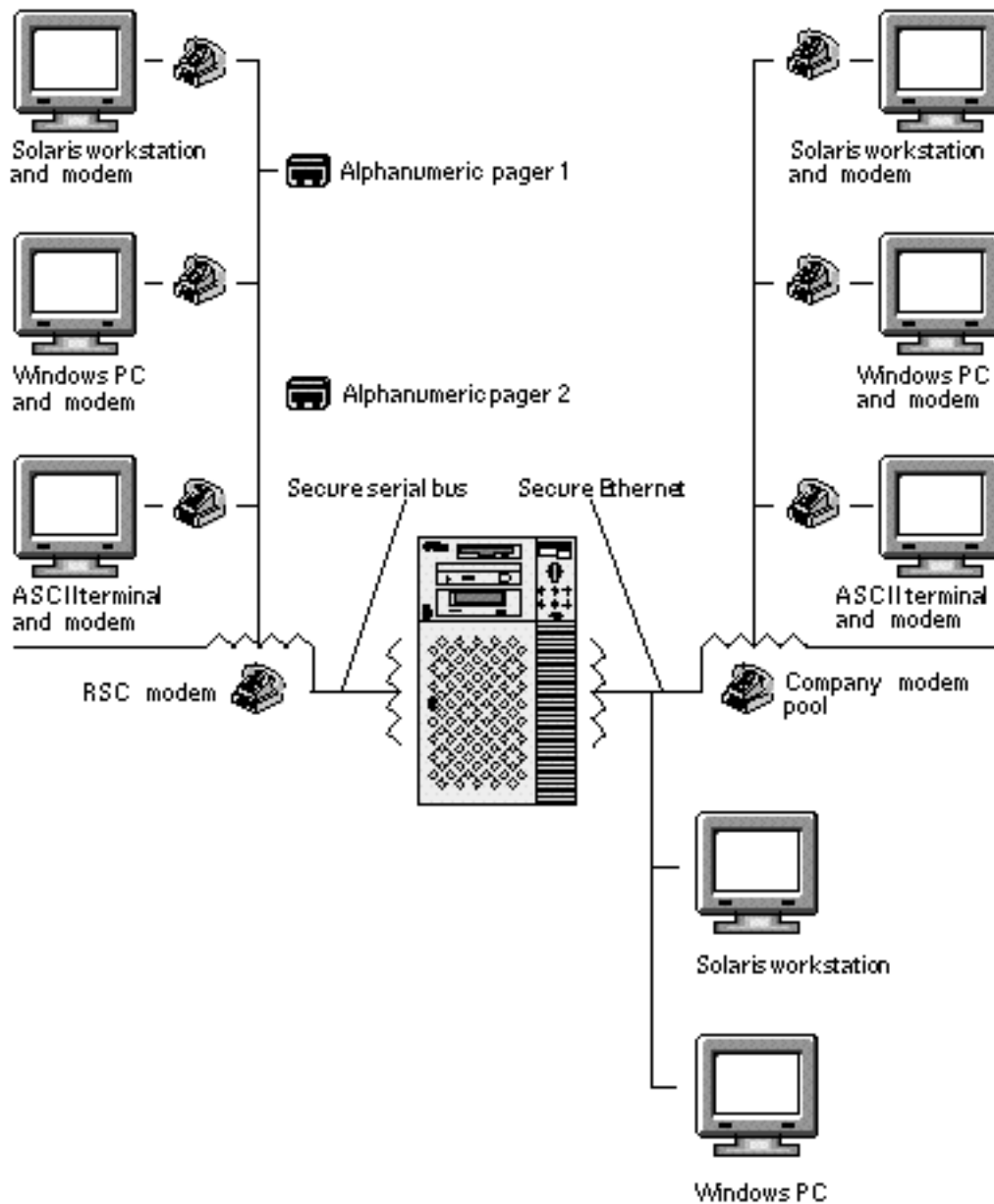
Figure 1–2 RSC Remote Alert Paths

After receiving an alert message, you can connect to your RSC account for the server that caused the alert. Then you can check console messages to determine whether the server has recovered and is running, or whether the server operating system is down.

If the server is running, you can log in to it using a separate connection and debug the system using UNIX system administration tools such as SunVTS and Solstice SyMON. If the client is not running the Solaris operating system, these tools may be available through X windows.

If the server is not running, or if UNIX system administration tools are not available, you can use the RSC remote console feature to debug the server.

You can also debug the server by using RSC to:

- Show environmental information

- Put the server into debug mode

- Control server firmware behavior

- Turn server power off and then on, if the server is hung

After diagnosing a problem, you can schedule server downtime and service, if necessary.

RSC Capabilities

Sun Remote System Control provides the following capabilities for remotely monitoring and controlling Sun Enterprise 250 servers.

Server Status and Control

- Show server environmental status.

- Access a console that includes all UNIX console functions.

- Send a break to put the server into debug mode.

- Control the diagnostics level for a server reboot.

- Reset the server on demand. When the system is not responding, you can issue a cold reset to bring the server back online.

- Turn server power off and on.

View Logs

- Display a detailed log of RSC errors, events, and RSC command history.

- Display and reset server console logs.

RSC Configuration

You can control RSC configuration settings for:

- Alerts
- Ethernet port
- Serial port
- RSC date and time
- Your RSC password
- RSC user accounts

After installation, you run a script file that guides you through basic configuration. See the *Sun Remote System Control (RSC) Installation Guide.*

If you want to be able to use RSC in case of a complete power failure, you should install and configure an uninterruptible power supply (UPS) for the server. See Appendix A.

If you plan to use a modem on the RSC serial port, you must configure the modem properly. See Appendix B.

User Interfaces

RSC user interfaces include:

- A graphical interface that runs using Sun's RSC Java application
- A command-line interface you can access using standard telnet to the RSC Ethernet port, and to the RSC serial port using point-to-point protocol (PPP)
- A command-line interface you can access by connecting an ASCII character terminal directly to the RSC serial port

RSC supports a maximum of two concurrent telnet connections per server. These connections can be command-line interface sessions or RSC GUI connections to the server console (Open Console). In addition, RSC supports up to three concurrent RSC GUI sessions.

RSC Security

The hardware and process required to log in to an RSC account on a managed server depend on company security practices, and whether Ethernet or modem dial-in is used. Further security is provided by RSC accounts, individual account access rights, and passwords. In addition, RSC records all logins, and sends an alert if it detects five login failures within five minutes.

Characteristics of the modem and phone line that are connected to the RSC serial port also determine access security for remote serial port connections, for instance, use of a dialback option or a phone line that supports dial-out only. You can set RSC to disconnect a session connected to the serial port after ten minutes of inactivity; see "`serial_hw_handshake`" on page 4-18.

> **Caution -** As for any computer session, do not leave your session unattended. Always use the RSC `logout` command before disconnecting your session. In addition, before logging out of RSC, take care to log out of any server sessions you initiate while using the RSC console.

RSC Client Requirements

To use the RSC graphical user interface, a client must have Sun's RSC Java application installed. To use the RSC command shell, a client must be an ASCII character terminal or have ASCII character terminal emulation software installed.

> **Note -** To run the RSC GUI Java application on a Solaris client machine, the client machine must have the Java Development Kit for Solaris version 1.1.5 or a subsequent compatible version installed. This software is included on the Sun Enterprise 250 Supplement CD-ROM and can be installed along with the RSC software from the CD-ROM. If you install RSC software from the Web, you must download the Java Development Kit from the following location: `http://www.sun.com/solaris/java/`

Installing and Configuring RSC Software

This chapter provides an overview of how to install and configure RSC software; for installation instructions and help configuring RSC, see the *Sun Remote System Control (RSC) Installation Guide.*

The RSC server software installation procedure allows you to set up RSC configuration, and you can change RSC configuration after installation using RSC commands or the `rscadm` utility.

Installing RSC Software

After operating system software is installed on the server to be managed, you can install RSC software on the server.

Note - After RSC software is installed on the host, it resides in the directory `/usr/platform/sun4u`.

Until RSC host software is installed and configured, the system console is available as on any normal Sun computer. You can enable RSC as the system console device by using the OpenBoot PROM command `diag-output-to` and defining the OpenBoot PROM variables `input-device` and `output-device` as `rsc`. See Chapter 6.

You must also install Sun's RSC Java application on any client machine that will be used to access the RSC graphical interface. The RSC Java application is available for clients running the Solaris, Windows 95, Windows 98, and Windows NT operating environments.

Note - To run the RSC GUI Java application on a Solaris client machine, the client machine must have the Java Development Kit for Solaris version 1.1.5 or a subsequent compatible version installed. This software is included on the Sun Enterprise 250 Supplement CD and the Sun Computer Systems Supplement CD, and can be installed along with the RSC software from the CD. If you install RSC software from the Web, you must download the Java Development Kit from the following location: `http://www.sun.com/solaris/java/`

A client machine must have ASCII terminal emulation software installed (or be an ASCII terminal) to use the RSC shell and `rscadm` utility. See *Sun Remote System Control (RSC) Release Notes* for a list of supported hardware and software platforms.

Note - After the RSC Java application is installed on a Solaris client, it resides in the directory `/opt/rsc` by default. On a Windows client, it resides in the directory `C:\Program Files\Sunw\Remote System Control` by default.

Configuring RSC Software

After installing RSC software on the server, you run the following script to configure RSC:

```
# /usr/platform/sun4u/sbin/rsc-config
```

After initial configuration, you can control configuration and add or modify user accounts by using the GUI or RSC shell commands. You can also log in to the host as root and use the `rscadm` utility to control RSC configuration and user accounts; see Chapter 5.

If you are using an uninterruptible power supply for the host or a modem on the RSC serial port, you must configure those devices properly. For configuration examples, see Appendix A and Appendix B.

Configuring PPP

If point-to-point protocol (PPP) is enabled on the serial port, RSC supports multiple sessions, either shell or GUI, through one modem connection. If PPP is not enabled, only one session can be run over the modem, and it must be a shell session. RSC supports multiple sessions over its Ethernet port.

To use PPP for dialing in to the RSC serial port, the RSC configuration variable `ppp_enabled` must be set to `true`, or PPP must be enabled using the RSC GUI. In addition, you must configure PPP on each client machine that will use PPP to dial in to RSC accounts on the RSC serial port.

Configuring PPP on the Client

To allow connection to RSC from a remote client using PPP, you must configure PPP on the client. For a Solaris client, see *Configuring and Using Solstice PPP 3.0.1 Clients.* For a Windows client, see your Windows documentation.

The Solstice PPP 3.0.1 client configuration script `/usr/bin/pppinit` creates a CHAT script in `/etc/opt/SUNWconn/ppp/script` that defines the dialog that occurs between a client and server during the connection phase. The contents of this CHAT script are not needed to establish a PPP connection to RSC. In order to successfully establish connection to RSC from a Solstice PPP client, comment or delete the contents of the CHAT script, but do not delete the file.

Redirecting the Console to RSC

Until RSC software is configured, the system console is available as on any normal Sun machine. To enable RSC as the system console device instead, you must log in to the server and enter the following commands at the `ok` prompt:

```
ok diag-output-to rsc
ok setenv input-device rsc
ok setenv output-device rsc
```

These commands take effect after the next server reset. At any time, you can use the following commands to remove RSC as the default console:

```
ok diag-output-to ttya
ok setenv input-device keyboard
ok setenv output-device screen
```

These commands take effect after the next server reset. See Chapter 6 for more information about these OpenBoot PROM features.

Configuring Alerts

To configure alerts, choose Configure Alerts in the GUI, or set the following configuration variables using the RSC shell or `rscadm` utility:

- `customerinfo`
- `hostname`
- `page_enabled`
- `mail_enabled`
- `page_info1`
- `page_info2` (optional)
- `mailuser`
- `mailhost`

The GUI provides help for configuring RSC alerts. For help configuring RSC alert variables in the shell, see "Alert Variables" on page 4-19. RSC generates alert messages with the following format:

`$HOSTID $EVENT $TIME $CUSTOMERINFO $HOSTNAME` *message*

You can also use the `rscadm` subcommand `send_event -c` to send an alert. You can do this directly at the superuser prompt, and you can also create a command file that runs and sends the alert under special circumstances. See Appendix C.

Backing Up RSC Configuration

You should periodically use `rscadm` commands to create a backup file that records RSC configuration settings on a remote system. For example:

```
# rscadm show > remote_filename
# rscadm usershow >> remote_filename
#
```

Use a meaningful file name that includes the name of the server that RSC controls.

Later, you can refer to this file to restore the settings if you need to reinstall RSC software on the server or replace the server's main logic board.

Using the RSC Graphical Interface

The easiest way to use Sun Remote System Control is through the graphical user interface (GUI). This chapter gives an overview of how to access the RSC GUI, and use of RSC windows and dialog boxes.

If you need to use RSC commands and the GUI is not available, you can use RSC shell commands. After you log in to your RSC account from a terminal window, the RSC shell prompt appears (`rsc>`), and you can enter commands.

Starting the RSC GUI

If you are using a client running the Solaris operating environment, start the RSC GUI by entering this command:

```
% /opt/rsc/bin/rsc
```

If you are using a client running the Windows 95, Windows 98, or Windows NT operating environment, follow these steps to start the RSC GUI:

1. **Click on the Start menu.**

2. **Point to Programs in the Start menu.**

3. **Point to the Sun Remote System Control program group.**

4. **Click on Sun™ Remote System Control.**

When the login screen appears, it prompts you to enter an RSC device name, your
RSC username, and your password.

Note - It is important that you enter the name or IP address of the RSC device, not
the name or IP address of the server.

To monitor or control more than one server, start a separate GUI session for each.

Accessing RSC Functions

The main screen of the Sun Remote System Control GUI lets you access the
following functions:

- Server Status and Control

  - Show Environmental Status
  - Open Console
  - Send Break
  - Send XIR
  - Set Boot Mode
  - Reset Server
  - Power Off/On (depends on current setting)

- View Logs

  - RSC Event Log
  - Console Boot Log
  - Console Run Log
  - Original Console Boot Log
  - Original Console Run Log
  - Reset Console Logs

- RSC Configuration

  - Alert Settings
  - Ethernet Settings
  - Serial Port Settings
  - Set RSC Date and Time
  - Change Password
  - User Administration
  - Reset RSC

- Help

  - Help Topics
  - About RSC

Online Help is available when you use the GUI. Additional detail about RSC functions is provided in the sections of Chapter 4 that describe the corresponding RSC shell commands.

RSC Permissions Required for Access to GUI Functions

All RSC users can look at RSC information. The table in the RSC User Administration dialog box shows additional permissions assigned to each user. These additional user permissions grant access to specific items in the GUI command tree.

Console Permission

Console permission allows the user to choose Open Console in the command tree and connect to the server console, and choose Send Break to put the server in debug mode.

User Permission

User permission allows the user to choose User Administration in the command tree to add, modify, and remove RSC user accounts.

Admin. Permission

Admin. permission allows the user to modify RSC configuration, including the following items in the command tree:

- Alert Settings
- Ethernet Settings
- Serial Port Settings
- Set RSC Date and Time

Reset Permission

Reset permission allows the user to use the following items in the command tree:

- Send XIR

- Set Boot Mode

- Reset Server

- Power Off or Power On

If no user permissions are checked, the user's permission is read-only. A user with read-only permission can use the following options only:

- Show Environmental Status

- Change Password

- View Logs options (except Reset Console Logs)

- Help Topics

- About RSC

Using the RSC Command Shell

The easiest way to use Sun Remote System Control is through the graphical user interface (GUI). Online Help is available when you use the GUI.

If you need to use RSC and the GUI is not available, you can use the RSC command shell. The RSC command shell is a simple command-line interface that supports commands that allow you to administer or diagnose the server. It also has commands for configuring RSC.

RSC supports a maximum of two concurrent telnet connections per server. These connections can be command-line interface sessions or RSC GUI connections to the server console (Open Console). In addition, RSC supports up to three concurrent RSC GUI sessions.

After you log in to your RSC account, the RSC shell prompt appears (`rsc>`), and you can enter RSC shell commands. This chapter tells you how to log in to your RSC account and describes RSC command use and syntax.

The following table summarizes RSC shell commands. Each command is explained in greater detail in the following sections.

TABLE 4–1 RSC Shell Commands

Name	Description
environment	Displays current environmental information
shownetwork	Displays the current network configuration
console	Connects you to the server console
break	Puts the server in debug mode

TABLE 4–1 RSC Shell Commands *(continued)*

Name	Description
xir	Generates an externally initiated reset to the server
bootmode	Controls server firmware behavior, if followed by a server reset within 10 minutes (similar to L1-key combinations on Sun keyboards)
reset	Resets the server immediately
poweroff	Powers off the server
poweron	Powers on the server
loghistory	Displays the history of all events logged in the RSC event buffer
consolehistory	Displays the history of all console messages logged in the buffer
consolerestart	Makes the current boot and run console logs "original"
set	Sets a configuration variable
show	Displays one or more configuration variables
date	Displays or sets the current time and date
password	Changes your RSC password
useradd	Adds an RSC user account
userdel	Deletes an RSC user account
usershow	Shows characteristics of an RSC user account
userpassword	Sets or changes a user's password
userperm	Sets the authorization for a user
resetrsc	Resets RSC immediately
help	Displays a list of RSC shell commands and a brief description of each

TABLE 4–1   RSC Shell Commands   *(continued)*

Name	Description
`version`	Displays the RSC firmware version
`logout`	Ends your current RSC shell session

Note - Some commands require a specific user permission level. See
"`userperm` *username* `[c][u][a][r]`" on page 4-15 for information about user
permission levels.

More detailed descriptions of the commands are provided in the following sections.

Logging In to Your RSC Account

After RSC software is installed and configured and an account has been set up for
you, you can connect to RSC and log in to your account using a Solaris workstation,
Windows PC, standard ASCII character terminal, or a computer running ASCII
terminal emulation software.

Follow these steps to log in to your RSC account:

1. **Connect to RSC using one of these five methods:**
   a. **Use point-to-point protocol (PPP) to connect to your company Ethernet, and
      then use the** `telnet` **command to connect to RSC.**
      Ask your network administrator if you do not know the server's RSC name (it
      is usually *servername*-`rsc`).

   b. **If you are already connected to your company Ethernet, use the** `telnet`
      command to connect to RSC.

   c. **Use PPP to connect to the RSC modem. To use this option, PPP must be
      enabled.**

   d. **If PPP is not enabled on the serial port, dial in to the RSC modem.**

   e. **Connect your client machine directly to the RSC serial port.**
      To use this option, you must disable PPP by using the RSC shell or `rscadm`
      utility to set the `ppp_enabled` configuration variable to `false`, or by using
      the GUI.

When connection is established, the following screen appears:

```
RSC version 1.0
Please login:
Please enter password:
```

2. **Enter your RSC login name.**

3. **Enter your RSC password.**

   Your password is not echoed on the screen. After you correctly enter your password, RSC displays this command prompt:

```
rsc>
```

   You can enter RSC shell commands at the `rsc>` prompt.

Logins are recorded in the RSC event log. In addition, RSC sends an alert if it detects five login failures within five minutes. You can also set RSC to disconnect a session connected to the serial port after ten minutes of inactivity; see "`serial_hw_handshake`" on page 4-18.

Server Status and Control Commands

The following RSC commands show server status or control server operation:

- `environment`
- `shownetwork`
- `console`
- `break`
- `xir`
- `bootmode`
- `reset`
- `poweroff`
- `poweron`

environment

Use the `environment` command to display a snapshot of server environmental status, such as temperatures, power supply status, front panel LED status, keyswitch position, and so forth. The display uses similar format to that used for the UNIX command `prtdiag(1m)`. For example:

```
rsc> environment


=================== Environmental Status ===================


System Temperatures (Celsius):
———————————————
      CPU0    42
      CPU1    52
      MB0     33
      MB1     27
      PDB     29
      SCSI    25


===============================
Front Status Panel:
————————-

Keyswitch position is in On mode.

System LED Status:  DISK ERROR      POWER
                  [OFF]        [ ON]
              POWER SUPPLY ERROR  ACTIVITY
                  [OFF]        [ ON]
                GENERAL ERROR    THERMAL ERROR
                  [OFF]        [OFF]


===============================


Disk LED Status:    OK = GREEN  ERROR = YELLOW
      DISK  5: [EMPTY]    DISK  3: [EMPTY]    DISK  1:   [OK]
      DISK  4: [EMPTY]    DISK  2:   [OK]    DISK  0:   [OK]


===============================


Fan Bank:
————

Bank      Speed    Status
          (0-255)
—-        ——      ———
 SYS       151        OK


===============================
```

(continued)

Power Supplies:
——————

Supply   Status
——   ——
  0   OK
  1   OK

===============================

rsc>

shownetwork

The `shownetwork` command displays the current network configuration, for example:

```
rsc> shownetwork
RSC network configuration is:
DHCP server: 129.149.2.3
IP Address: 129.149.2.6
Gateway Address: 129.149.2.7, 129.149.2.8
Netmask: 255.255.255.0
Ethernet Address: ae:30:30:00:00:01
rsc>
```

console

Use the `console` command to enter RSC console mode and connect to the server console from the RSC shell. When you use this command, the system displays a standard Solaris login prompt. If RSC is not designated as the server console, nothing is displayed.

You must have `C` level user permission to use this command. An escape character sequence is used to return to the RSC prompt. The default escape sequence is ~. (tilde period). See "`escape_char`" on page 4-24.

break

Use the `break` command to put the server into debug mode. You must have `C` level user permission to use this command. The server front panel keyswitch must not be in the Lock position, and the server serial port must be designated as the console (see Chapter 6). Debug mode can use either `kadb` or OBP.

xir

This command generates the equivalent of an externally initiated reset (XIR) of the server. You must have `R` level user permission to use this command. The server enters OBP mode and displays the `ok` prompt. This command is useful for driver or kernel debugging, as most of the contents of the server's memory and registers is preserved. To resume the system after using the `xir` command, you must reboot the server.

bootmode [-u] [normal|forth|reset_nvram|diag| skip_d

This command provides control over the server firmware behavior after a server reset. Functionality of this command is identical to that available on Sun keyboards using L1 key combinations. You must have `R` level user permission to use this command. If you use the `bootmode` command without arguments, RSC displays the current `boot_mode`.

The `bootmode` setting overrides the server's OBDiag `diag-switch?` setting immediately after the next reset only. If RSC does not detect a server reset within 10 minutes, the `bootmode` command is ignored. For example:

```
rsc> bootmode forth
rsc> reset
```

To set the `diag` or `skip_diag` option, you must follow the `bootmode` command with the `poweroff` and `poweron` commands within 10 minutes. For example:

```
rsc> bootmode skip_diag
rsc> poweroff
rsc> poweron
```

To force console input and output to RSC, use the `-u` option before specifying the boot mode.

The following table describes modes you can specify using the `bootmode` command.

TABLE 4–2 Modes Used With the `bootmode` Command

Mode	Description
`-u`	Force the server to direct the console to RSC; the `-u` option must precede any *boot_mode* you specify; requires server reset
`normal`	Normal boot; server runs low-level diagnostics; requires server reset
`forth`	Enter Forth interpreter as soon as possible (equivalent to L1-F on keyboard); requires server reset
`reset_nvram`	Reset all NVRAM variables to default values (equivalent to L1-N on keyboard); requires server reset
`diag`	Force the server to run full diagnostics (equivalent to L1-D on keyboard); requires server power-off and power-on
`skip_diag`	Force the server to skip diagnostics (equivalent to L1-S on the keyboard); requires server power-off and power-on

Note - The `diag` and `skip_diag` modes take effect only if you follow the `bootmode` command with the `poweroff` and `poweron` commands within 10 minutes.

reset

This command forcibly resets the server immediately. You must have R level user permission to use this command. The server reboots according to the server's designated `boot_mode`.

Note - The default configuration of Sun Enterprise 250 firmware does not invoke POST when the server resets. However, you can change this behavior through NVRAM variable settings; for additional details, see *Platform Notes: Sun Enterprise 250 Server.*

poweroff

Use the `poweroff` command to power off the server. You must have R level user permission to use this command. This command has no effect if the server is already powered off. RSC remains available since it uses the server's standby power.

poweron

Use the `poweron` command to power on the server. You must have `R` level user permission to use this command. This command has no effect if any of the following is true:

- The server's keyswitch is in the Standby position.

- The server is already powered on.

RSC View Log Commands

Use the following RSC commands to work with RSC and console log files:

- `loghistory` (or `lhist`)

- `consolehistory` (or `chist`)

- `consolerestart`

loghistory [index [+│-]*n*] [pause *n*]

Use the `loghistory` command without subcommands to display the history of all events logged in the RSC event buffer. These events include server reset events and all RSC commands that change the state of the system. You can also use the command abbreviation `lhist`.

Use the following subcommands to control `loghistory` display.

index [+│-]*n*

Use the `index` subcommand to designate a buffer position at which to begin the display, as follows:

- `index` +*n* to designate a line number relative to the beginning of the buffer

- `index` -*n* to designate a line number relative to the end of the buffer

- `index` *n* to designate a line number relative to the beginning of the buffer (same as `index` +*n*)

The origin of counting is 1; that is, `index +1` indicates the first line in the buffer, `index -1` indicates the last. For example:

```
rsc> loghistory index -30
```

This command prints the last 30 lines and any additional lines that were appended to the buffer between the time that the command began execution and the time that it terminated.

pause *n*

Use the `pause` subcommand to display *n* lines of the log at a time (similar to the `more` command). The value of *n* must be a decimal integer. The default is to display the entire RSC log without pausing.

Each event recorded in the log has the following format:

```
$TIME $HOSTNAME $EVENTID $message
```

`EVENTID` is a unique identifier for the event, `TIME` is the time the event occurred (as measured by RSC time), and *message* is a user-friendly description of the event.

The following is an example event log entry:

```
FRI JAN 01 07:33:03 1999 sst4828: 00060003: ''RSC System booted''
```

consolehistory [boot|run|oboot|orun] [index [+|-]*n*]

Use the `consolehistory` command to display console messages logged in RSC buffers. With no arguments, this command prints the entire contents of all non-empty console buffers. You can use the command abbreviation `chist`.

There are four console logs:

- The `boot` buffer contains POST, OBP, and UNIX boot messages received from the server for the most recent boot.
- The `run` buffer contains the most recent data received from the server operating system.
- The `oboot` buffer contains POST, OBP, and UNIX boot messages for the first power-on boot, the *original boot*.
- If the system panics and resets, the `orun` buffer will contain the most recent messages printed to the console before a reboot, which are the panic messages.

The `boot` and `oboot` buffers can contain up to 8 Kbytes of information, and the `run` and `orun` buffers can contain up to 16 Kbytes of information.

When the first power-on boot begins, RSC fills the original boot (`oboot`) buffer with data from the server's console. After that buffer fills up, it writes data to the original run (`orun`) log. When the `orun` log fills up, it overwrites old data in the `orun` log.

When RSC senses a server reset while in the `orun` log, it switches to the `boot` log. After that fills up, it switches to the `run` log. When the `run` log fills up, it overwrites old data in the `run` log.

When RSC senses a server reset while in the current `run` log, it switches to the current `boot` log again.

pause *n*

Use the `pause` subcommand to display *n* lines of the log at a time (similar to the `more` command). The value of *n* must be a decimal integer. The default is to display 10 lines of the log at a time.

See "`loghistory [index [+|-]n] [pause n]`" on page 4-9 for a description of the `index` subcommand.

Note - Time stamps recorded in console logs reflect server time. These time stamps may be offset from RSC time stamps recorded in the RSC event log. To find information on synchronizing RSC time with server time using the `rscadm` utility, see "`date [-s] date [[mmdd]HHMM|mmddHHMM[cc]yy]][.SS]`" on page 5-2.

---

consolerestart

Use the `consolerestart` command to make the current boot and run logs the "original" logs (designated `oboot` and `orun`). This command copies the current `boot` and `run` buffers to the `oboot` and `orun` buffers, overwriting the previous contents. Then it clears the current `boot` and `run` buffers. You must have `A` level user permission to use this command.

After adding a hardware component to the server, reboot the server and then use the `consolerestart` command so that the new part will appear in the original console logs.

RSC Configuration Commands

Use the following RSC commands to set or show characteristics of the RSC or server configuration:

- `set`
- `show`
- `date`

- password
- useradd
- userdel
- usershow
- userpassword
- userperm
- resetrsc

set *variable value*

Use the `set` command to set an RSC configuration variable. You must have A level user permission to use this command. See "RSC Configuration Variables" on page 4-17 for descriptions of these variables.

Changes to some variables do not take effect until the SSP that RSC is running on is reset using the command `resetrsc` or `rscadm resetrsc`.

You can use the null string ("") to set a variable to null. To set a variable to a string that includes spaces, enclose the string in double quotes. For example:

```
rsc> set page_info2 ""
rsc> set page_init1 "&F &E0"
```

show [*variable*]

Use the `show` command to display the value of RSC configuration variables. You can specify one variable only; if you do not specify a variable, RSC displays all configuration variables. See "RSC Configuration Variables" on page 4-17 for descriptions of these variables.

date [[mmdd]HHMM|mmddHHMM[cc]yy][.SS]

Use the `date` command without arguments to show RSC's current date and time. If you have A level user permission, you can use the `date` command to set the current date and time. The following table describes components of the date format.

TABLE 4–3   Components of the `date` Command

Option	Description
mm	Month number
dd	Day-of-the-month number
HH	Hour number (24-hour system)
MM	Minute number
.SS	Second number
cc	First two digits of year
YY	Last two digits of year

You may omit the month, day, and year; the current values are applied as defaults.

Examples:

```
rsc> date 091521451998
rsc> date 09152145
rsc> date 2145
```

The first example sets the time to September 15, 9:45 p.m., 1998. The second example sets the time to September 15, 9:45 p.m. of the current year. The third example sets the time to 9:45 p.m. of the current month, day, and year.

Note - Whenever the server boots, it sets the RSC current date and time. However, to keep RSC time in sync with server time, Sun Microsystems recommends that you implement a script that uses the command `rscadm date -s` to update RSC time periodically from the server time. The RSC shell `date` command cannot synchronize RSC time with the server time.

password

Use the `password` command to change the RSC password for the account to which you are logged in. This command behaves similarly to the UNIX `passwd(1)` command.

When used to change a password, RSC prompts for your current password, and if you enter it correctly, it prompts for the new password. RSC prompts again for the new password, and updates it if entered identically both times. For example:

```
rsc> password
password: Changing password for username
Enter login password:
Enter new password:
Re-enter new password:
rsc>
```

Passwords have the following restrictions:

- They must contain at least six characters (only the first eight characters are significant).
- They must contain at least two alphabetic characters and at least one numeric or special character; alphabetic characters can be both uppercase and lowercase.
- They must differ from the user's login name and any reverse or circular shift of that login name; for comparison purposes, uppercase and lowercase letters are equivalent.
- The new password must differ from the old by at least three characters; for comparison purposes, uppercase and lowercase letters are equivalent.

useradd *username*

Adds an RSC user account. You must have U level user permission to use this command. The maximum number of RSC user accounts is four. Valid characters for *username* include:

- Alphabetic characters
- Numeric characters
- Period (.)
- Underscore (_)
- Hyphen (-)

The username field has a maximum length of eight characters, must contain at least one lowercase alphabetic character, and the first character must be alphabetic. If these restrictions are not met, the system issues a warning and the command fails.

userdel *username*

Deletes an RSC user account. You must have U level user permission to use this command.

usershow [*username*]

Shows RSC user accounts. You must have `U` level user permission to use this command. If no argument is supplied, all accounts are shown. Information displayed includes username, permissions, and whether a password is assigned. For example:

```
rsc> usershow
Username Permissions Password?
setup    cuar        Assigned
msmith   c--r        None
rsc>
```

userpassword *username*

Sets or changes the password for the specified user account. You must have `U` level user permission to use this command. RSC does not prompt for an existing password. See the `password` command for details on password format and restrictions. For example:

```
rsc> userpassword msmith
New password:
Re-enter new password:
rsc>
```

userperm *username* [c][u][a][r]

Sets or changes permission levels for a specified user account.

All RSC users can look at RSC information. The following arguments increase a user's authorization level:

- `c` – Console permission; authorized to connect to the server console

- `u` – User administration permission; authorized to use commands that add and delete users, change user permissions, and change the authorization level of other users

- `a` – Administration permission; authorized to change the state of RSC configuration variables

- `r` – Reset/power permission; authorized to reset, power on, and power off the server, and reboot RSC

You must have `U` level user permission to use this command. You can specify zero through four authorizations. The default authorization level for a new RSC account is none of the above.

If you do not specify authorization levels, RSC sets the permissions for *username* to read-only. However, the default user permission for the account you create during the installation procedure is `cuar` (full authorization).

A user with read-only permission can use the following commands only:

- `help`
- `password`
- `date` (show only)
- `shownetwork`
- `environment`
- `loghistory`
- `consolehistory`
- `show`
- `version`
- `logout`

resetrsc

The `resetrsc` command performs a hard reset of RSC. This terminates all current RSC sessions. You must have `A` level user permission to use this command. You can also reset RSC by using the `rscadm resetrsc` command.

Other RSC Commands

help

The `help` command displays a list of all RSC shell commands and a brief description of each.

version [-v]

The `version` command displays the firmware version running on RSC. Use the `-v` option to display more verbose information. Examples:

```
rsc> version
RSC Bootmon version: 0.23
RSC version: 0.23
rsc> version -v
```

```
RSC Bootmon version: 0.23
RSC Bootmon checksum: A91695A9
RSC version: 0.23
RSC firmware checksum: 4FF151DA
RSC firmware built Apr 28 1998, 12:31:15
rsc>
```

logout

The `logout` command ends your RSC session and closes your RSC connection.

RSC Configuration Variables

RSC has non-volatile configuration variables you can use to change RSC behavior. The installer sets most configuration variables during installation. After installation, you can use the GUI, RSC shell, or the `rscadm set` command to set or change configuration variables.

You must have A level user permission to set configuration variables from the GUI or RSC shell; you must log in to the server as root to use the `rscadm set` command.

Use the `show` command to show configuration variables, and use the `set` command to set a variable. Changes to variables take effect immediately except as noted.

You can use the null string ("") to set a variable to null. To set a variable to a string that includes spaces, enclose the string in double quotes.

Serial Port Variables

serial_baud

This variable sets the RSC serial port baud rate. Valid values are:

- 9600
- 19200
- 38400
- 57600
- 115200

The default setting is 9600. Changes to this variable take effect on the next login connection over the RSC serial port.

If you change the serial baud rate after configuring the serial port modem, you must reconfigure the modem; see "Modem Reconfiguration Required After Changing RSC Serial Port Settings" on page B-10.

serial_parity

This variable sets the RSC serial port parity; valid values are none, odd, or even. The default setting is none. Changes to this variable take effect on the next login connection over the RSC serial port.

If you change the serial parity setting after configuring the serial port modem, you must reconfigure the modem; see "Modem Reconfiguration Required After Changing RSC Serial Port Settings" on page B-10.

serial_stop

This variable sets the number of RSC stop bits; valid values are 1 or 2. The default setting is 1. Changes to this variable take effect on the next login connection over the RSC serial port.

If you change the serial stop bits setting after configuring the serial port modem, you must reconfigure the modem; see "Modem Reconfiguration Required After Changing RSC Serial Port Settings" on page B-10.

serial_data

This variable sets the number of RSC data bits; valid values are 7 and 8. The default setting is 8. Changes to this variable take effect on the next login connection over the RSC serial port.

If you change the serial data bits setting after configuring the serial port modem, you must reconfigure the modem; see "Modem Reconfiguration Required After Changing RSC Serial Port Settings" on page B-10.

serial_hw_handshake

This variable controls whether RSC uses hardware handshaking on the serial port. Valid values are true and false; the default setting is true. If set to true, hardware flow control and modem control are enabled. If set to false, RSC disconnects a session connected to the serial port after 10 minutes of inactivity.

ppp_local_ip_addr

Use this variable to specify the Internet protocol (IP) address for RSC to use during a PPP session; use standard dot notation. If this variable is empty, RSC expects the remote node to set the Internet address dynamically. The default setting is `0.0.0.0` (empty). Changes to this variable take effect on the next PPP connection over the RSC serial port.

ppp_remote_ip_addr

Use this variable to specify the IP address for the remote node during a PPP session; use standard dot notation. If this variable is empty, RSC expects that the remote node already has an Internet address assigned for the PPP session. The default setting is `0.0.0.0` (empty). Changes to this variable take effect on the next PPP connection over the RSC serial port.

ppp_enabled

Use this variable to specify whether PPP is the default protocol on the RSC serial port. Valid values are `true` or `false`; the default is `false`. Changes to this variable take effect on the next login connection over the RSC serial port.

Alert Variables

page_enabled

Use this variable to specify whether paging is enabled for RSC alerts. Valid values are `true` or `false`; the default is `false`.

mail_enabled

Use this variable to specify whether an email message is enabled for RSC alerts. Valid values are `true` or `false`; the default is `false`.

page_info1

This variable contains the phone number and associated pager ID number for sending a TAP alert to a pager. Use the character `@` to separate the telephone number and ID. Valid characters are:

- digits (`0-9`)

- * (asterisk)

- # (pound sign

- , (comma)

If both the `page_info1` and `page_info2` variables are empty, or if the `page_enabled` variable is set `false`, no paging will occur. The default setting for `page_info1` is empty. For example:

```
rsc> set page_info1 18004420500@1234332
```

page_init1

This variable specifies a string consisting of attention (AT) commands used for configuring the RSC modem prior to sending an alert to pager 1.

page_password1

This variable contains the pager service password used to send an alert to pager 1. The password must be an alphanumeric string of six characters or fewer.

page_baud1

This variable specifies the serial port baud rate to use when sending an alert to pager 1. Valid values are `300`, `1200`, `2400`, `4800`, and `9600`. After RSC sends the page alert, the baud setting reverts to the setting specified in the `serial_baud` variable.

page_data1

This variable specifies the number of RSC serial port data bits for sending an alert to pager 1. Valid values are `7` and `8`. After RSC sends the page alert, the data bits revert to the setting specified in the `serial_data` variable.

page_parity1

This variable specifies the RSC serial parity for sending an alert to pager 1. Valid values are `none`, `odd`, and `even`. After RSC sends the page alert, the parity reverts to the setting specified in the `serial_parity` variable.

page_stop1

This variable specifies the number of RSC serial port stop bits for sending an alert to pager 1. Valid values are 1 and 2. After RSC sends the page alert, the stop bits revert to the setting specified in the serial_stop variable.

page_info2

This variable contains the phone number and associated pager ID number for sending a TAP alert to another pager. Use the character @ to separate the telephone number and ID. Valid characters are:

- digits (0-9)
- * (asterisk)
- # (pound sign
- , (comma)

If both the page_info1 and page_info2 variables are empty, or if the page_enabled variable is set false, no paging will occur. The default setting for page_info2 is empty. For example:

```
rsc> set page_info2 18004420596@4433444
```

page_init2

This variable specifies a string consisting of attention (AT) commands used for configuring the RSC modem prior to sending an alert to pager 2.

page_password2

This variable contains the pager service password used to send an alert to pager 2. The password must be an alphanumeric string of six characters or fewer.

page_baud2

This variable specifies the serial port baud rate to use when sending an alert to pager 2. Valid values are 300, 1200, 2400, 4800, and 9600. After RSC sends the page alert, the baud setting reverts to the setting specified in the serial_baud variable.

page_data2

This variable specifies the number of RSC serial port data bits for sending an alert to pager 2. Valid values are 7 and 8. After RSC sends the page alert, the data bits revert to the setting specified in the serial_data variable.

page_parity2

This variable specifies the RSC serial parity for sending an alert to pager 2. Valid values are none, odd, and even. After RSC sends the page alert, the parity reverts to the setting specified in the serial_parity variable.

page_stop2

This variable specifies the number of RSC serial port stop bits for sending an alert to pager 2. Valid values are 1 and 2. After RSC sends the page alert, the stop bits revert to the setting specified in the serial_stop variable.

customerinfo

This variable contains customer information used in the message generated for a pager or email alert, for instance, one of the following:

- Server service contract number (recommended)
- Server location
- Name or telephone extension of the server's system administrator
- Name of the department that owns the server

This string can be up to eight characters, including alphanumeric characters and hyphen (-). The default setting is empty.

hostname

This variable contains the name of the server connected directly to RSC; RSC includes this name in alert messages. This string can be up to eight characters, including alphanumeric characters and hyphen (-). The default setting is empty.

mailuser

This variable contains the address used for email alerts. The string has a 40-character limit; the default setting is empty. Although only a single mail address is permitted, you can notify several people that a problem has occurred by using an email alias.

```
mailhost
```

This variable contains a colon-separated list of Internet addresses to which RSC sends SMTP mail alerts. Each address is tried until the SMTP alert is successfully transmitted. Specify addresses using standard dot notation. For example:

```
rsc> set mailhost 139.143.4.2:139.142.4.15
```

You can specify a maximum of two addresses. If this variable is empty, or if the `mail_enabled` variable is set to `false`, no email is sent. The default setting is empty.

Ethernet Port Variables

ip_mode

Use this variable to control how RSC configures the IP address for its local Ethernet port. Choose the `ip_mode` value according to the services available on the network to which RSC is connected. The following list describes the available values.

- `none` – Ethernet port is disabled and is not accessible.
- `dhcp` – Use the dhcp protocol to obtain the IP address.
- `config` – Use the ip_addr configuration variable to obtain the IP address.

The default setting is `dhcp`. Changes to this variable take effect after the next RSC reset.

ip_addr

This variable is only used if the `ip_mode` variable is set to `config`. Specify the IP address in standard Internet dot notation; the default setting is `0.0.0.0` (empty). Changes to this variable take effect after the next RSC reset.

ip_netmask

This variable is only used if the `ip_mode` variable is set to `config`. Specify the subnet mask in standard Internet dot notation; the default setting is `0.0.0.0` (empty). Changes to this variable take effect after the next RSC reset.

```
ip_gateway
```

This variable is only used if the `ip_mode` variable is set to `config`. This is the default gateway to which RSC will send IP packets when the destination is not on the same subnet as RSC. Specify the IP address in standard Internet dot notation; the default setting is `0.0.0.0` (empty). Changes to this variable take effect after the next RSC reset.

Console Session Variable

```
escape_char
```

This variable sets the escape sequence character for ending a console session and returning to the RSC shell. The escape sequence applies to all RSC users for the server. The escape sequence consists of the escape character followed by a period. The default escape character is ~ (tilde).

The escape sequence character can be a single alphanumeric character. It can also be a control character. To enter a control character as the escape character, type "^" (Shift-6) to represent the Control key, followed by another character. If the second character is a question mark (?), the Delete key is selected; otherwise the second character is converted to a control character and used as the escape character. For example, if you enter ^y to set the escape character, users will use Control-y to end a console session.

RSC Shell Error Messages

This section provides additional information about error messages received at the `rsc>` prompt.

Usage Errors

Error messages 1 through 19 are usage error messages, displayed due to improper command syntax; see the description of the command in this chapter for the correct syntax.

```
Invalid command. Type 'help' for list of commands.

Usage: bootmode [-u] [normal|forth|reset_nvram|diag|skip_diag]

Usage: break
```

```
Usage: consolehistory [boot|run|oboot|orun] [index [+|-]<n>]

Usage: consolerestart

Usage: date [[mmdd]HHMM | mmddHHMM[cc]yy][.SS]

Usage: loghistory [index [+|-]<n>]

Usage: password

Usage: poweroff

Usage: poweron

Usage: reset

Usage: resetrsc

Usage: set <variable> <value>

Usage: shownetwork

Usage: useradd <username>

Usage: userdel <username>

Usage: userpassword <username>

Usage: userperm <username> [c][u][a][r]

Usage: version [-v]

Usage: xir
```

General Errors

RSC reports the following general errors.

`Could not get username for user <username>`

During execution of the `userpassword` command, a SEEPROM error occurred.

`Error adding user <username>`

An error occurred during execution of the `useradd` command; this message is followed by a more detailed message.

`Error changing password for <username>`

An error occurred during execution of the `userpassword` command; this message is followed by a more detailed message.

`Error changing password for <username>`

During execution of the `userpassword` command, a SEEPROM error occurred.

```
Error changing password for <username> - password must be at
least three characters different from old password - password
must not be based on username
```

You entered an invalid password.

`Error deleting user <username>`

An error occurred during execution of the `userdel` command; this message is followed by a more detailed message.

`Error displaying user <username>`

An error occurred during execution of the `usershow` command; this message is followed by a more detailed message.

`Error setting permission for <username>`

An error occurred during execution of the `userperm` command; this message is followed by a more detailed message.

```
ERROR: username did not start with letter or did not contain
lowercase letter.
```

You entered an invalid username.

`Failed to allocate buffer for console mode.`

During execution of the `console` command, RSC could not allocate enough memory to connect to the console.

`Failed to allocate memory!`

During execution of the `show` command, RSC could not allocate enough memory to show the variable's value.

`Failed to get password for <username>`

During execution of the `userpassword` command, a SEEPROM error occurred.

`Failed to set <variable> to <value>`

During execution of the `set` command, RSC encountered a SEEPROM error.

```
Invalid login
```

Login attempt failed; this message appears at the login prompt.

```
Invalid password
```

You entered an illegal password with the `userpassword` command.

```
Invalid permission: <permission>
```

You entered an invalid user permission. Valid permissions are [c] [u] [a] [r]. See
"`userperm` *username* `[c][u][a][r]`" on page 4-15.

```
Malformed username
```

You specified a nonexistent user when entering the `userpassword`, `userperm`,
or `userdel` command.

```
No free user slots
```

This error occurs if you try to add a user account when RSC already has four
accounts configured. RSC supports only four user accounts; you must delete an
account before you can add another.

```
Passwords don't match
```

The two entries for a new password did not match.

```
Permission denied
```

You attempted to execute a shell command for which you do not have the proper
user permission level.

```
Sorry, wrong password
```

You entered an incorrect current password.

```
Unable to get value of variable <variable>
```

During execution of the `show` command, you used an invalid variable name.

```
User already registered
```

The user you are trying to add already has an RSC account on this server.

```
User does not exist
```

The username you specified is not associated with an RSC account on this server.

Using the `rscadm` Utility

The `rscadm` utility and its subcommands allow administration of RSC from the host. You must log in to the server as root to use `rscadm`. Most `rscadm` subcommands are also available using the RSC command shell, but `rscadm` is convenient for:

- Reconfiguring RSC when account passwords are unknown
- Resetting RSC when it fails to respond
- Backing up configuration data
- Downloading RSC firmware
- Synchronizing RSC date and time to the server's date and time.

The `rscadm` utility uses the following subcommands.

TABLE 5–1   `rscadm` Subcommands

Subcommand	Description
`help`	Displays a list of `rscadm` commands and brief descriptions for each
`date`	Displays or sets the current time and date
`set`	Sets a configuration variable
`show`	Displays one or more configuration variables
`resetrsc`	Resets RSC immediately
`download`	Downloads firmware to the RSC flash PROM

TABLE 5–1   rscadm Subcommands   *(continued)*

Subcommand	Description
send_event	Logs an event; can also send an alert message
modem_setup	Changes configuration of the modem connected to the RSC serial port
useradd	Adds an RSC user account
userdel	Deletes an RSC user account
usershow	Shows characteristics of an RSC user account
userpassword	Sets or changes a user's password
userperm	Sets the authorization for a user

`rscadm` Subcommands

`help`

This subcommand prints out a basic Help message including a list of `rscadm` subcommands and a short description of each.

`date [-s] date [[mmdd]HHMM|mmddHHMM[cc]yy]][.SS]`

Use the `date` subcommand to show the current date and time; use the `-s` option to synchronize RSC time to the server time. Supply a date with the subcommand to set RSC time to a time other than the server time.

See "`date [[mmdd]HHMM|mmddHHMM[cc]yy][.SS]`" on page 4-12 for a description of date formats.

set *variable value*

Use the `set` subcommand to set a RSC configuration variable. See "RSC Configuration Variables" on page 4-17 for descriptions of these variables.

You can use the null string ("") to set a variable to null. To set a variable to a string that includes spaces or UNIX shell special characters, enclose the string in double quotes. For example:

```
# rscadm set page_info2 ""
# rscadm set page_init1 "&F &E0"
```

show [*variable*]

Use the show subcommand to display the value of one or more RSC configuration variables. If you do not specify a variable, RSC displays all configuration variables. See "RSC Configuration Variables" on page 4-17 for descriptions of these variables.

resetrsc [-s]

This subcommand resets RSC immediately. To terminate all connections cleanly before the reset, use the `-s` option. If no argument is supplied, this subcommand performs a hard reset and drops all connections.

download [boot] *file*

This subcommand supports downloading new firmware residing in *file* into RSC. If you specify `boot`, the contents of *file* will be installed in the boot section of RSC non-volatile memory. If you do not specify `boot`, the contents of *file* will be installed in the main firmware section of RSC non-volatile memory.

When the transfer completes, RSC resets itself. If the host keyswitch is in the Lock position, you cannot update RSC firmware, and an error message displays.

send_event [-c] *message*

Use this subcommand to enter an event into the RSC event log. Use the `-c` option to also send an alert message that RSC forwards according to the alert configuration. The *message* is an ASCII string of no more than 80 characters. To use a string that includes spaces or UNIX shell special characters, enclose the string in double quotes.

See Appendix C for an example of a Perl script that uses this subcommand to send
an alert.

modem_setup

Use this subcommand to change configuration of the modem connected to the RSC
serial port. If the modem is properly connected, you can enter standard AT
commands and see responses from the modem. At the beginning of a line, enter the
escape character (~) followed by a period to exit from this subcommand. Example:

```
# rscadm modem_setup
AT <enter>
OK
~.
#
```

The escape sequence for `rscadm modem_setup` is always `~.`. The RSC console has a
configurable exit character, but the escape character for `rscadm modem_setup` is
hard-coded.

User Account Administration Subcommands

To administer user accounts from the host using `rscadm`, log in to the host as root
and use the `rscadm` utility with the following subcommands:

- `useradd` (see "`useradd` *username*" on page 4-14)
- `userdel` (see "`userdel` *username*" on page 4-14)
- `usershow` (see "`usershow` [*username*] " on page 4-15)
- `userpassword` (see "`userpassword` *username*" on page 4-15)
- `usserperm` (see "`userperm` *username* `[c][u][a][r]`" on page 4-15)

`rscadm` Error Messages

This section offers further explanation of error messages received from the `rscadm`
utility. The last 12 messages are usage error messages.

```
Passwords didn't match, try again
```

During execution of the `userpassword` subcommand, you enter the password
twice. If passwords do not match, you are prompted to enter again.

```
rscadm: all user slots are full
```

This error occurs if you try to add a user account when RSC already has four accounts configured; RSC supports only four user accounts. You must delete an account before you can add another.

```
rscadm: command line too long
```

Check for an extremely long command line.

```
rscadm: command unknown
```

You used an invalid rscadm subcommand.

```
rscadm: could not connect to modem
```

During execution of the modem_setup subcommand, RSC could not connect to the RSC modem. Is the modem connected? Perhaps the modem is currently being used by the paging subsystem, or RSC is connected to another modem.

```
rscadm: could not disconnect from modem
```

During execution of the modem_setup subcommand, RSC refused to disconnect from the modem.

```
rscadm: could not read date from RSC
```

An undefined error in RSC firmware occurred when trying to get the RSC date.

```
rscadm: could not send alert
```

During execution of the send_event subcommand, RSC firmware couldn't send event.

```
rscadm: could not set date on RSC
```

RSC detected an internal error while trying to set the date.

```
rscadm: couldn't add user
```

RSC encountered an internal error when trying to add a user account. The RSC SEEPROM may be faulty.

```
rscadm: couldn't change password
```

RSC encountered an internal error when trying to change a user password. The RSC SEEPROM may be faulty.

```
rscadm: couldn't change permissions
```

RSC encountered an internal error when trying to change user permissions. The RSC SEEPROM may be faulty.

```
rscadm: couldn't create thread
```

During execution of the `modem_setup` subcommand, a thread create call failed.

```
rscadm: couldn't delete user
```

RSC encountered an internal error when trying to delete a user account. The RSC SEEPROM may be faulty.

```
rscadm: couldn't get information on user
```

RSC encountered an internal error when trying to access user information during a `usershow` subcommand. The RSC SEEPROM may be faulty.

```
rscadm: download failed, RSC reported erase error
```

During execution of the `download` subcommand, RSC reported a hardware problem when trying to program the RSC EEPROM.

```
rscadm: download failed, RSC reported int_wp error
```

During execution of the `download` subcommand, RSC reported a hardware problem when trying to program the RSC EEPROM.

```
rscadm: download failed, RSC reported range error
```

During execution of the `download` subcommand, RSC reported a hardware problem when trying to program the RSC EEPROM.

```
rscadm: download failed, RSC reported verify error
```

During execution of the `download` subcommand, RSC reported a hardware problem when trying to program the RSC EEPROM.

```
rscadm: download failed, RSC reported vpp error
```

During execution of the `download` subcommand, RSC reported a hardware problem when trying to program the RSC EEPROM.

```
rscadm: download failed, RSC reported wp error
```

During execution of the `download` subcommand, RSC reported a hardware problem when trying to program the RSC EEPROM.

```
rscadm: download rejected, keyswitch in secure mode?
```

You cannot execute the `download` subcommand when the server keyswitch is in the Lock position.

```
rscadm: either the RSC hardware was not detected or a lock file
was found. Only one instance of rscadm can run at a given time.
```

As stated.

`rscadm: Error downloading file`

An internal error occurred during execution of the `download` subcommand.

`rscadm: ERROR, callback init failed`

An internal error occurred during execution of the `download` subcommand.

`rscadm: ERROR, passwords didn't match`

During execution of the `userpassword` subcommand, you enter the password twice. If passwords do not match, you are prompted to enter again. If they do not match again, the subcommand fails.

`rscadm: ERROR, unable to set up message queue`

An internal error occurred during execution of the `download` subcommand.

`rscadm: event message can't exceed 80 characters`

The message for the `send_event` subcommand cannot exceed 80 characters.

`rscadm: file could not be opened`

During execution of the `download` subcommand, could not open the file specified on the command line.

`rscadm: file not a valid s-record`

During execution of the `download` subcommand, the file to download is not a valid s-record file.

`rscadm: INTERNAL ERROR in set date`

This is an internal `rscadm` software error.

`rscadm: INTERNAL ERROR, overflow in callback`

An internal error occurred during execution of the `download` subcommand.

`rscadm: invalid variable`

During execution of the `set` subcommand, you entered an invalid variable. See "RSC Configuration Variables" on page 4-17 for correct values.

`rscadm: invalid variable or value`

During execution of the `set` subcommand, you entered an invalid variable or value. See "RSC Configuration Variables" on page 4-17 for correct values.

```
rscadm: malformed password
```

You entered an invalid password. A valid password has between 6 and 8 characters, at least two of which are alphabetic characters, and at least one of which is a numeric or special character.

```
rscadm: malformed username
```

You entered invalid characters in a username.

```
rscadm: maximum username length is - 8
```

The username you entered exceeded eight characters; the maximum length for username is eight characters.

```
rscadm: RSC did not respond during boot initialization
```

An internal error occurred during execution of the download subcommand.

```
rscadm: RSC failed to respond during download
```

During execution of the download subcommand, RSC did not enter boot mode correctly.

```
rscadm: RSC firmware not responding
```

RSC main firmware is not responding. Perhaps RSC is booting, main firmware is corrupt, or RSC has a hardware problem.

```
rscadm: RSC not responding to requests
```

The response expected from RSC was not sent. Is RSC working?

```
rscadm: RSC returned fatal error
```

During execution of the download subcommand, RSC returned an undocumented error.

```
rscadm: RSC returned garbage
```

This error can occur in various situations; as stated.

```
rscadm: RSC returned unknown error
```

During execution of the download subcommand, RSC returned undocumented status (neither success nor failure).

```
rscadm: RSC returned wrong response
```

RSC returned an invalid response during a user* subcommand. This is considered an internal error in RSC or rscadm.

```
rscadm: RSC unable to free up memory
```

This message can occur in various situations. The `rscadm` utility was unable to free the received message from the RSC firmware.

```
rscadm: Unable to reset RSC hardware
```

During execution of the `resetrsc` subcommand, an attempt to hard reset RSC failed.

```
rscadm: unable to send data to RSC
```

RSC did not acknowledge sent data. Is RSC working?

```
rscadm: Unable to send modem data to RSC
```

RSC did not acknowledge data sent to it. Is RSC working?

```
rscadm: user already exists
```

The user you are trying to add already has an RSC account on this server.

```
rscadm: username did not start with letter or did not contain
lowercase letter
```

You used an invalid username format when trying to add an RSC user account.

```
rscadm: username does not exist
```

The username you specified is not associated with an RSC account on this server.

```
This program MUST be run as root
```

As stated.

```
USAGE: rscadm <command> [options]
```

For a list of subcommands, type `rscadm help`.

```
USAGE: rscadm date [-s] | [[mmdd]HHMM | mmddHHMM[yyyy]][.SS]
```

Date parameters have the following ranges:

- 01 <= mm <= 12
- 01 <= dd <= 31
- 01 <= HH <= 23
- 00 <= MM <= 59
- 19 <= cc <= 22
- 00 <= yy <= 99

- 00<= SS <= 59

```
USAGE: rscadm download [boot] <file>

USAGE: rscadm resetrsc [-s]

USAGE: rscadm send_event [-c] "message"

USAGE: rscadm set <variable> <value>

USAGE: rscadm show [variable]

USAGE: rscadm useradd <username>

USAGE: rscadm userdel <username>

USAGE: rscadm userpassword <username>

USAGE: rscadm userperm <username> [cuar]
```

The user you are trying to add already has an RSC account on this server.

```
USAGE: rscadm usershow [username]
```

Using OpenBoot PROM Features That Support RSC

This chapter contains information about OpenBoot PROM features that support Sun Remote System Control.

OBP Commands

Several new OBP commands have been added to support RSC. You enter these commands at the `ok` prompt.

rsc-hard-reset

Performs a hard reset of RSC; this is the same as using the command `rscadm resetrsc`.

rsc-soft-reset

Performs a soft reset of RSC; this is the same as using the command `rscadm resetrsc -s`.

`diag-output-to rsc|ttya`

Directs POST output to either RSC (1) or ttya (0). This command takes effect after the next server reset.

`rsc-mac-update`

Updates the RSC Ethernet address from the contents of the server ID PROM. Use this command after replacing the server NVRAM module.

`.rsc`

Displays RSC information, including the `diag-output-to` setting and the RSC POST status word.

OBP Environment Variable Properties

Two environment variable properties are available to the user for specifying the RSC console to OBP. Define these variables at the `ok` prompt. For example:

`ok` **`setenv input-device rsc`**

These properties take effect after the next server reset.

`rsc`

This property is a new device alias that is valid for `ttyio`, `input-device`, and `output-device` OBP console variables. Screen output and keyboard input are typically still set to be the default, and `ttya` and `ttyb` continue to be other valid options.

`rsc!`

This property is a new device alias that is valid for `ttyio`, `input-device`, and `output-device` OBP console variables. Screen output and keyboard input are typically still set to be the default, and `ttya` and `ttyb` continue to be other valid

options. This variable does not allow OBP to autosearch for a working console if RSC is broken.

Troubleshooting

This chapter includes information about troubleshooting RSC problems, and troubleshooting the server using RSC.

Troubleshooting RSC Problems

Can't log in to RSC

- Check the RSC device name you are connecting to (normally *servername*-rsc).

- Check that you are using your correct RSC username; it may not be the same as your system username.

- Check that you are using your correct RSC password.

Can't connect to RSC using telnet

RSC supports a maximum of two simultaneous telnet connections. This includes command line interface sessions and RSC GUI connections to the server console (Open Console). When two telnet sessions are active, further attempts to connect using the `telnet` command will receive a connection closed error. For example, from UNIX:

```
% telnet bert-rsc
Trying 129.148.49.120...
Connected to bert-rsc.
Escape character is '^]'.
Connection closed by foreign host.
```

Can't connect to RSC through the Ethernet connection

First, log in to the server as root and see if the command `rscadm status` reports a good status. If it does, then RSC is working and there is an Ethernet configuration problem. Use the `rscadm show` command to check that Ethernet configuration variables are set correctly.

You can also:

■ Log in to RSC through the serial port and use the GUI or the `shownetwork` command to see the current settings.

■ Log in to another machine on the network and use the `ping` command to see if RSC is alive. Use RSC's name (for instance, *servername*-`rsc`), not the server name, as the argument to the `ping` command.

■ Run SunVTS diagnostics to check the Ethernet connection; the external Ethernet test requires that the device be connected to a functional 10-Mbit hub.

■ Run SunVTS diagnostics to check the SSP and paddle card.

Can't connect to RSC through the serial port

First, log in to the server as root and see if the command `rscadm status` reports a good status. If it does, then RSC is working. Log in to the server as root and use the `rscadm show` command to check that serial port configuration variables are set correctly.

If the problem is with a modem:

■ Log in to the server as root and use the command `rscadm modem_setup` to check and verify modem configuration. It is very important that the modem is configured to use a fixed baud rate for incoming calls.

■ Verify that the modem is working properly by attaching it to another computer.

■ Attach a terminal or another system to the serial port and verify that the serial port is working. Set the configuration variable `serial_hw_handshake false` and then reset RSC before trying this.

■ Run SunVTS to check the serial port; the external serial port test requires a serial loopback connector.

■ Run SunVTS to check the SSP and paddle card.

No alerts received from RSC

- RSC does not send alerts for all RSC events; check that the alert you are expecting is an event for which RSC sends alerts.

- Log in to the server as root and use the command `rscadm send_event -c` *message* to send an alert.

If email alerts are not received:

- Check email configuration variables.

- Check that the mail server and backup mail server are up and configured properly.

- Try sending an email message to the recipients without using RSC.

- Check Ethernet configuration.

If pager alerts are not received:

- Check pager configuration variables.

- Try sending a pager message by some method other than RSC.

- Check with your pager service for proper pager configuration, or problems with the paging service.

RSC passwords are unknown

If users have forgotten RSC passwords or passwords are not working, log in to the server as root and use the command `rscadm userpassword` to assign new passwords. Inform RSC users of the new passwords.

Times in RSC event log do not match times in server console logs

Log in to the server as root and use the command `rscadm date -s` to synchronize RSC time to server time. Consider setting up a repetitive job to do this using the `cron` utility.

You can perform some RSC functions, but not others

Specific user permissions are required to perform functions; check your permission level. In addition, the following problems may exist:

- Cannot put the server in debug mode or use the RSC `break` command:

The server keyswitch is in the Lock position.

- The poweroff command has no effect:

  The server is already powered off.

- The poweron command has no effect:

  The server is already powered on, or the keyswitch is in the Standby position.

RSC is not working after you replace the main logic board

After replacing the server main logic board, you must reconfigure RSC. Use the `rscadm` utility to restore configuration settings from a backup file. You may also need to update the server flash PROM with the latest version of the Sun Enterprise 250 OpenBoot firmware.

Troubleshooting Server Problems Using RSC

RSC is useful for troubleshooting a server that is not responding. If the server is responsive, connect to it as usual and use standard tools such as Solstice SyMON, SunVTS, and OpenBoot Diagnostics.

If the server is not responding, log in to your RSC account and:

- Check the RSC event log and server environmental status for problems.
- Check console logs for recent error messages.
- Try connecting to the server console to reboot the system.

Scenario: An unreliable SIMM in memory bank 2 is causing continuous reboots

1. **Log in to RSC and check the RSC event log.**

   You will see repeated server resets in the log.

2. **Examine console logs.**

   You will see many SIMM bank 2 errors in the original run log, and eventually a panic message.

3. **Connect to the server console.**

   You may see that the system is up now, but receiving periodic SIMM errors.

4. **Log in to the server as root, and halt the system.**

5. **Enter the following OBP command to disable SIMM bank 2:**

```
ok> asr-disable bank2
```

6. **Reboot the server.**

7. **Order new memory for the server.**

Installing and Configuring an Uninterruptible Power Supply (UPS)

To allow use of RSC when a complete power failure occurs, connect the host power cord to an uninterruptible power supply (UPS). SunExpress™ sells several suitable UPS devices.

This appendix provides an example of installing and configuring an American Power Conversion SmartUPS 1000 with PowerChute *plus* 4.2.2 software, for use with RSC. The UPS uses RSC software on the server to record events and generate alerts.

How to Install the UPS

1. **Install the UPS as documented in the *SmartUPS Quick Reference Guide*.**

2. **Install and configure PowerChute *plus* monitoring software as documented in the SmartUPS document *Software Installation: Instruction Sheet*.**

3. **Connect the UPS to one of the server serial ports (not the RSC serial port).**

How to Configure the UPS to Record Events and Send Alerts

PowerChute *plus* allows you to run a command file or shell script whenever a designated "event action" occurs. See "Configuring Actions" in the *PowerChute plus User Guide*. Use the command `rscadm send_event` to log events and send alerts to RSC from the server.

1. **Log in to the server and start the PowerChute *plus* application.**

2. **Choose Event Actions from the Configuration menu.**

3. **In the Event Actions dialog box, click on an event that you want to notify RSC about (for example, "UPS on Battery").**

4. **Select the Run Command File and then click on the associated Options button.**
   The Run Command File dialog box opens.

5. **Using a text editor, create a command file with the following content:**

   `rscadm send_event -c "UPS: UPS On Battery"`

   You can use any message string in place of `"UPS: UPS On Battery"`. Be sure to enclose the message string in double quotation marks.

   Use the `-c` option to send an alert and also record the event in the RSC event log. To log the event only, do not use the `-c` option.

6. **Save the file as** `/usr/platform/sun4u/sbin/ups_batt.com`.

7. **Type the name of this file into the Run Command File dialog box:**

   `/usr/platform/sun4u/sbin/ups_batt.com`

8. **Enter an integer for Seconds Before Executing.**
   This value will normally be 0.

9. **Click on OK.**

10. **Repeat steps 2 through 9 for each event that you want to create a log entry or send an alert.**

11. **Click on OK in the Event Actions dialog box to confirm changes.**

12. **Choose Exit from the PowerChute *plus* System menu.**

Configuring the RSC Serial Port Modem

You can access RSC using an Ethernet connection, and you can also install a dedicated modem connected to the RSC serial port for access from outside your company network. For increased security, you can specify a dialback number if the modem supports this feature.

To install and configure the modem, follow the instructions that came with it. This appendix gives examples for configuring a MultiTech MultiModem II, MT2834 Series Intelligent Data/Fax Modem (Sun part number 370-2234-03), and a Courier V.Everything modem.

To enable and set up the RSC modem, you should use the following configuration settings unless otherwise indicated. You can use the GUI to change configuration settings, or you can use the RSC shell or the `rscadm` utility to change configuration variables.

General Settings

Use the following settings to configure the RSC serial port.

TABLE B–1   Serial Port General Settings

GUI Setting	Configuration Variable Setting
Serial Port Settings	
Baud Rate: 9600	`serial_baud 9600`
Parity: None	`serial_parity none`
Stop Bits: One	`serial_stop 1`
Data Bits: 8	`serial_data 8`
Enable hardware handshaking	
Enable PPP on serial port	`serial_hw_handshake true`
	`ppp_enabled true`

Pager Variable Settings

For sending pager alerts, RSC is designed to operate with any modem paging service that complies with the Telocator Alphanumeric Protocol (TAP). If you want to support both dial-in to the RSC modem and dial-out to pagers, you should set the RSC serial port baud rate and pager baud rates to 9600.

Use the following settings to enable and configure pager alerts. Use of pager 2 is optional; you do not need to enter settings for pager 2 if you are using a single pager. In this table, the characters -> indicate menu hierarchy; for example, Alert Settings->Pager means open Alert Settings and then choose Pager.

TABLE B–2   Pager Alert Settings

GUI Setting	Configuration Variable Setting
Alert Settings->Pager	
Send RSC alerts to pagers	`page_enabled true`
Alert Settings->Pager->Pager 1	
Number: *phone number to dial*	`page_info1` *number@ID*
PIN: *pager user's PIN*	

TABLE B–2 Pager Alert Settings *(continued)*

GUI Setting	Configuration Variable Setting
Alert Settings->Pager->Pager 1->Advanced	
Baud Rate: 9600	`page_baud1 9600`
Parity: even	`page_parity1 even`
Stop Bits: 1	`page_stop1 1`
Data Bits: 7	`page_data1 7`
Password: *password*	`page_password1 password`
Modem Init. String: see "Modem Initialization Strings" on page B-4	`page_init1`
Alert Settings->Pager->Pager 2	
Number: *phone number to dial*	`page_info2 number@ID`
PIN: *pager user's PIN*	
Alert Settings->Pager->Pager 2->Advanced	
Baud Rate: 9600	`page_baud2 9600`
Parity: even	`page_parity2 even`
Stop Bits: 1	`page_stop2 1`
Data Bits: 7	`page_data2 7`
Password: *password*	`page_password2 password`
Modem Init. String: see "Modem Initialization Strings" on page B-4	`page_init2`

Pager Phone Number

Pager phone numbers are the number for the paging service's TAP connection, and not the number for voice page notifications. It must be a phone number that connects to a modem that adheres to the TAP protocol. You may need to contact your paging service to obtain this number as well as other pager configuration parameters.

Pager phone numbers must include the complete string to dial; for instance, if your internal phone network requires a 9 before the number, include it. You can include a

comma (,) after the 9 if a pause is required in order to wait for a dial tone. For example, `9,8005551212`.

Storing Configuration Strings in the Modem's NVRAM

Use the `rscadm modem_setup` command to set modem configuration. Arguments to the `modem_setup` command consists of attention (AT) commands. Then store the settings in the modem's NVRAM.

Use the following initialization:

- Turn off RTS flow control.
- Reset modem when DTR drops.
- Set the serial port to fixed baud rate for incoming calls.

Modem Initialization Strings

Use the modem initialization strings for pagers 1 and 2 to allow RSC to set modem parameters required by the paging service during modem setup. The initialization string consists of attention (AT) commands.

The variables `page_init1` and `page_init2` have a maximum length of 15 characters, and are not stored in the modem's NVRAM. Settings in the `page_init1` and `page_init2` variables complement or override the settings you store in the modem's NVRAM memory using the `rscadm modem_setup` command. Note that if you use a different phone number or paging service for pagers 1 and 2, you may need to enter different initialization strings for each pager.

See "Setting Up the MultiTech Modem" on page B-5 for the MultiTech II modem initialization string, and see "Setting Up the Courier V.Everything Modem" on page B-7 for the Courier V.Everything modem initialization string.

Pager Passwords

The pager passwords are the password for the TAP paging service. In most cases in the United States, this password is not used. However, some paging services (for instance, in the United Kingdom) require a password.

Setting Up the MultiTech Modem

Use the following procedure to set up the MultiTech MultiModem II.

Set Modem DIP Switches

Check that all modem DIP switches for the MultiTech MultiModem II are in the default position, as specified in the following table

TABLE B–3   MultiTech Modem II DIP Switch Settings

Switch Function	Setting: Up/Down
Forced DTR	DTR function normally: Up
Flow control	Hardware flow control: Up
Enable/suppress responses	Enable responses: Down
UNIX UUCP spoofing	UUCP spoofing disabled: Up
Enable/disable auto-answer	Auto-answer enabled: Up
Maximum throughput setting	Maximum throughput enabled: Up
Request to send forced	RTS forced on: Down
Enable/disable command mode	Enable command mode: Down
Digital loopback test	Remote digital loopback test: Down
Leased line/dial-up operation	Dial-up operation: Up
AT responses/MultiTech responses	MultiTech responses: Down
Asynchronous/synchronous operation	Asynchronous operation: Down
Speed switch	28,800-bps operation: Up
Speed switch	28,800-bps operation: Up
Carrier detect/DSR forcing	CD and DSR normal: Up

.

Modify RSC Configuration Variables

Using the RSC GUI or RSC shell, set the following configuration variables as shown.

TABLE B–4 MultiTech Modem II RSC Settings

GUI Setting	Configuration Variable Setting
Serial Port Settings	
Baud Rate: 9600	`serial_baud 9600`
Enable hardware handshaking	`serial_hw_handshake true`
Alert Settings->Pager->Pager 1->Advanced	
Baud Rate: 9600	`page_baud1 9600`
Modem Init. String: `&D3&E0&E14$BA1`	`page_init1 &D3&E0&E14$BA1`
Alert Settings->Pager->Pager 2->Advanced	
Baud Rate: 9600	`page_baud2 9600`
Modem Init. String: `&D3&E0&E14$BA1`	`page_init2 &D3&E0&E14$BA1`

The string `&D3` sets the modem to reset when DTR drops, the string `&E0` disables error correction, the string `&E14` disables data compression, and the string `$BA1` turns off speed conversion.

To check that these settings are correct, use the `rscadm show` command.

Modify Modem Settings

Use the `rscadm modem_setup` command to enter the following AT commands:

```
# rscadm modem_setup
OK
ATZ (Reset modem)
OK
AT$BA1 (Turn off speed conversion)
OK
AT&W0 (Write out settings to modem)
OK
AT$SB9600 (Set serial port baud rate to 9600)
```

```
OK
AT&W0 (Write out settings to modem)
OK
AT$BA0 (Turn on speed conversion)
OK
AT&W0 (Store current settings to modem NVRAM)
OK
```

After executing these AT commands, you can verify the modem settings in NVRAM by using the `ATL5` command:

```
OK
ATL5
B1 E1 M1 Q0 R0 V1 X0 &E0 &E4 &E6 &E8 &E10 &E13 &E14 %C0 #C1 *C0 &C1
$MB9600 $SB9600 $BA0 &W0
OK
```

After confirming settings, turn the modem off, reset RSC using the GUI or the `resetrsc` shell command, and then turn the modem on again.

Setting Up the Courier V.Everything Modem

Use the following procedure to set up the Courier V.Everything modem.

Set Modem DIP Switches

DIP switches for the Courier V.Everything modem should be set according to the following table.

TABLE B–5   Courier V.Everything DIP Switch Settings

Switch Function	Setting: Off/On
1 Normal response to DTR signal/ignore DTR signal	Normal response to DTR signal: Off
2 Verbal/numeric result code format	Verbal result code format: Off
3 Display/suppress result codes	Display result codes: On

TABLE B–5   Courier V.Everything DIP Switch Settings   *(continued)*

Switch Function	Setting: Off/On
Echo/do not echo offline commands	Do not echo offline commands: On
Auto-answer/do not auto-answer	Auto-answer: Off
Send carrier detect signal normally/always send CD	Send carrier detect signal normally: Off
Display all result codes/display only when originating	Display all result codes: Off
Act on AT commands (smart)/ignore AT commands (dumb)	Act on AT commands: On
Disconnect on receipt of +++/enter online command mode on receipt of +++	Disconnect on receipt of +++: Off
Load the configuration that is stored in non-volatile memory/load the &F0 configuration from read-only memory	Load the configuration that is stored in non-volatile memory: Off

Modify RSC Configuration Variables

Using the RSC GUI or RSC shell, set the following configuration variables as shown.

TABLE B–6   Courier V.Everything RSC Settings

GUI Setting	Configuration Variable Setting
Serial Port Settings	
Baud Rate: 9600	serial_baud 9600
Enable hardware handshaking	serial_hw_handshake true
Alert Settings->Pager->Pager 1->Advanced	
Baud Rate: 9600	page_baud1 9600
Modem Init. String: E1&B1&M0&K0&N6	page_init1 E1&B1&M0&K0&N6
Alert Settings->Pager->Pager 2->Advanced	
Baud Rate: 9600	page_baud2 9600
Modem Init. String: E1&B1&M0&K0&N6	page_init2 E1&B1&M0&K0&N6

The string E1 turns echo on, the string &B1 sets fixed serial port baud rate, the string &M0 disables error correction, the string &K0 disables data compression, and the string &N6 sets baud rate to 9600.

To use a baud rate other than 9600, change the baud rate for pager 1 or 2, and also use the corresponding AT command in the modem initialization string:

- &N1 – 300 baud
- &N2 – 1200 baud
- &N3 – 2400 baud
- &N4 – 4800 baud
- &N6 – 9600 baud

To check that these settings are correct, use the rscadm show command.

Modify Modem Settings

Use the rscadm modem_setup command to enter the following AT commands:

```
# rscadm modem_setup
OK
ATZ (Reset modem)
OK
ATE1 (Turn echo on for this session)
OK
AT&N6 (Set serial port baud rate to 9600)
```

```
OK
AT&R1 (Turn off RTS flow control)
OK
ATS13=1 (Set S-register 13 to 1; reset modem when DTR drops)
OK
AT&B1 (Set serial port to fixed rate)
OK
AT&W (Store current settings to modem NVRAM)
OK
```

After executing these AT commands, you can verify the modem settings in NVRAM by using the ATI5 command:

```
OK
ATI5
USRobotics Courier V.Everything NVRAM Settings...

   BAUD=9600    PARITY=N  WORDLEN=8  DIAL=TONE

  B0   F1   M1   X1   &A1  &B1  &G0  &H0  &I0  &K0
  &L0  &M0  &N6  &P0  &R1  &S0  &T5  &U0  &X0  &Y1  %N6  #CID=0

  S00=001 S02=043 S03=013 S04=010 S05=008 S06=002 S07=060 S08=002
  S09=006 S10=014 S11=070 S12=050 S13=001 S15=000 S19=000 S21=010
  S22=017 S23=019 S24=150 S25=005 S26=001 S27=000 S28=008 S29=020
  S31=000 S32=009 S33=000 S34=000 S35=000 S36=000 S37
OK
```

After confirming settings, turn the modem off, reset RSC using the GUI or the resetrsc shell command, and then turn the modem on again.

Modem Reconfiguration Required After Changing RSC Serial Port Settings

If you change the RSC serial port baud rate after configuring the modem, you must reconfigure the modem to use the new baud rate. Use a similar procedure if you change the serial parity, data bits, or stop bits setting.

1. **Log in as root and use the** rscadm modem_setup **command.**

   The modem adjusts automatically to the new baud rate.

2. **If desired, adjust the modem's phone line baud rate to match the serial port baud rate.**

   For example, if you changed the serial port baud rate to 19200, use the following AT commands:

For the MultiTech II modem: `AT&$MB19200`.

For the Courier modem: `AT&N10`.

3. **Write the modem's current settings to the modem's NVRAM, so that the modem will default to the current settings.**

For the MultiTech II modem, use this AT command at the `OK` prompt: `AT&W0`.

For the Courier modem, use this AT command at the `OK` prompt: `AT&W`.

Troubleshooting Modem Problems

You can log in to the host as root and use the command `rscadm modem_setup` to debug modem problems. Then you can issue AT commands to the modem and attempt to dial the paging terminal of your paging service. When you connect successfully to a paging terminal that uses the TAP protocol, you should receive the `ID=` prompt, indicating the start of a successful TAP connection.

If you cannot log in successfully using AT commands, check modem hardware configuration such as DIP switches, and RSC serial port settings.

Instead of using the `rscadm` command, you can connect the modem to one of the server's serial ports and use a UNIX utility such as `tip` to connect to the TAP paging terminal. This allows you to check the modem independent of RSC. If you can connect successfully using this method, you know you are using the correct paging terminal number. See the man pages for `tip` and `stty` for UNIX-level troubleshooting.

To check pager alerts, use the command `rscadm send_event -c` *message*. This sends an alert message to pagers specified by RSC configuration.

Creating a Script That Sends an Alert Message or RSC Event

You can embed the `rscadm` subcommand `send_event` in a script or command file to log an RSC event or send an alert when certain conditions occur. Use the `-c` option to send an alert.

This appendix provides an example Perl script file named `dmon.pl` that sends an RSC alert when a specified disk partition exceeds a specified percent of its capacity. To use this script as intended, submit a separate entry to the `crontab` utility for each disk partition you want to monitor.

```
#!/usr/bin/perl

# Disk Monitor
# USAGE: dmon <mount> <percent>
#  e.g.: dmon /usr 80


@notify_cmd = '/usr/platform/sun4u/sbin/rscadm';


if (scalar(@ARGV) != 2)
{
   print STDERR "USAGE: dmon.pl <mount_point> <percentage>\n";
   print STDERR " e.g. dmon.pl /export/home 80\n\n";
   exit;
}

open(DF, "df -k|");
$title = <DF>;

$found = 0;
while ($fields = <DF>)
{
   chop($fields);
```

(continued)

```
($fs, $size, $used, $avail, $capacity, $mount) = split(' ', $fields);
if ($ARGV[0] eq $mount)
{
   $found = 1;
   if ($capacity > $ARGV[1])
   {
      print STDERR "ALERT: \"", $mount, "\" is at ", $capacity,
            " of capacity, sending notification\n";
      $nofify_msg = 'mount point "'.$mount.'" is at '.
            $capacity.' of capacity';
      exec (@notify_cmd, 'send_event', '-c', $nofify_msg)
         || die "ERROR: $!\n";
   }
}
}

if ($found != 1)
{
   print STDERR "ERROR: \"", $ARGV[0],
         "\" is not a valid mount point\n\n";
}

close(DF);
```