



Sun™ Remote System Control (RSC) User's Guide

For the Netra™ ct Server Alarm Card

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303-4900 U.S.A.
650-960-1300

Part No. 806-3301-11
February 2001, [Revision A](#)

[Send comments about this document to: docfeedback@sun.com](mailto:docfeedback@sun.com)

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303-4900 U.S.A. All rights reserved.

This product or document is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, AnswerBook2, docs.sun.com, Solaris, Netra, SunVTS, OpenBoot, and Solstice SyMON are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303-4900 Etats-Unis. Tous droits réservés.

Ce produit ou document est distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, AnswerBook2, docs.sun.com, Solaris, Netra, SunVTS, OpenBoot, et Solstice SyMON sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.



Contents

Preface xi

1. Sun Remote System Control Software Overview 1

Accessing RSC 1

RSC Capabilities 3

Server Status and Control 3

View Logs 3

RSC Configuration 3

Using RSC 4

RSC Security 6

2. Configuring RSC Software 7

RSC Software and the Alarm Card 7

Verifying RSC Software Installation 8

▼ To Verify RSC Software Installation 8

Accessing the Alarm Card and RSC 8

Running the RSC Configuration Script 9

▼ To Run the RSC Configuration Script 9

Additional Configuration 11

Using the Environmental Monitoring Daemon 11

▼	To Verify envmond Installation	11
▼	To Verify envmond Is Running	12
▼	To Stop and Start envmond	12
	Configuring PPP	12
	Configuring Alerts	13
	Redirecting the Console to RSC	13
▼	To Redirect the Console to RSC	14
▼	To Redirect the Console From RSC to the CPU Serial Port	14
	Console Session Information	15
	Backing Up the RSC Configuration	15
3.	Using the RSC Command Shell	17
	Logging In to Your RSC Account	17
▼	To Log In to Your RSC Account	18
	RSC Command Overview	19
	Server Status and Control Commands	20
	environment	20
	shownetwork	20
	console	21
	break	21
	xir	21
	reset	21
	poweroff	22
	poweron	22
	powersupply <i>n</i> [on off]	22
	alarm <i>n</i> [on off]	22
	RSC View Log Commands	22
	loghistory [index [+ -] <i>n</i>] [pause <i>n</i>]	23

consolehistory [run|orun]
 [index [+|-]n] [pause n] 24
consolerestart 25

RSC Configuration Commands 25

set *variable value* 25
show [*variable*] 25
date [[mmdd]HHMM|mmddHHMM[cc]yy][.SS] 26
password 27
useradd *username* 27
userdel *username* 28
usershow [*username*] 28
userpassword *username* 28
userperm *username* [c][u][a][r] 28
resetrsc 29

Other RSC Commands 30

help 30
version [-v] 30
logout 30

RSC Configuration Variables 30

▼ To Set a Configuration Variable in the RSC Command Shell 31

▼ To Set a Configuration Variable in the `rscadm` Utility 31

Serial Port Variables 32

Alert Variables 34

Alarm Variables 38

Ethernet Port Variables 41

Console Session Variable 42

Server Variable 43

4. Using the rscadm Utility	45
Overview of the Utility	45
rscadm Subcommands	46
help	46
version	47
date [-s]	
date [[mdd]HHMM mddHHMM[cc]yy]][.SS]	47
set <i>variable value</i>	47
show [<i>variable</i>]	47
resetrsc [-s]	48
download [boot] <i>file</i>	48
send_event [-c] <i>message</i>	48
modem_setup	48
User Account Administration Subcommands	49
Turning on an Alert at a UNIX Prompt	49
5. Using OpenBoot PROM Features that Support RSC	51
rsc	51
6. Troubleshooting	53
Troubleshooting RSC Problems	53
Troubleshooting Server Problems Using RSC	56
A. Configuring Alarm Card Modems	57
B. Sample Script for an Alert Message or RSC Event	61
C. Error Messages	63
Index	71

Figures

- FIGURE 1-1 RSC Remote Access Paths 2
- FIGURE 1-2 RSC Remote Alert Paths 5
- FIGURE 3-1 Message Format 39

Tables

TABLE 2-1	Information for RSC Configuration Script	9
TABLE 3-1	RSC Shell Commands	19
TABLE 3-2	Components of the <code>date</code> Command	26
TABLE 4-1	<code>rscadm</code> Utility Subcommands	46

Preface

The *Sun Remote System Control (RSC) User's Guide for the Netra ct Server Alarm Card* describes how to use the Sun™ Remote System Control (RSC) software with your Sun Netra™ ct server alarm card. This manual is written for experienced system administrators with networking knowledge.

How This Book Is Organized

This manual contains the following chapters and appendices:

Chapter 1 is an overview of the RSC software.

Chapter 2 describes how to configure and back up the RSC software.

Chapter 3 describes how to use the RSC command shell, including commands and variables.

Chapter 4 describes how to use the `rscadm` utility and its subcommands.

Chapter 5 describes OpenBoot™ PROM features that support RSC.

Chapter 6 contains information on troubleshooting RSC problems.

Appendix A describes using external modems with the alarm card.

Appendix B contains a sample script to log an RSC event or send an alert when certain conditions occur.

Appendix C contains the error messages generated by the RSC software.

Using UNIX Commands

This document may 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*
- AnswerBook2[™] online documentation for the Solaris[™] operating environment
- Other software documentation that you received with your system

Typographic Conventions

TABLE P-1 Typographic Conventions

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

Shell Prompts

TABLE P-2 Shell Prompts

Shell	Prompt
C shell	<i>machine_name%</i>
C shell superuser	<i>machine_name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#
Remote System Control shell	rsc>
OpenBoot PROM shell	ok

Related Documentation

In addition to the *Sun Remote System Control (RSC) User's Guide for the Netra ct Server Alarm Card*, the Netra ct server documentation set includes the following manuals:

TABLE P-3 Netra ct Server Documentation

Title	Part Number
<i>Netra ct Server Start Here</i>	806-5161
<i>Netra ct Server Product Notes</i>	806-3299
<i>Netra ct Server Safety and Compliance Manual</i>	806-3295
<i>Netra ct Server Product Overview</i>	806-3298
<i>Netra ct Server Installation Guide</i>	806-3294
<i>Netra ct Server Service Manual</i>	806-3296

Depending on the options you might have purchased for your machine, you might have also received manuals for network interface cards.

As mentioned above, as a Netra ct server purchaser, you received a suite of online documentation for the Solaris operating environment.

Accessing Sun Documentation Online

The `docs.sun.comsm` web site enables you to access Solaris 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`

Documentation and product information for the Netra product line are available at:

`http://www.sun.com/netra`

Ordering Sun Documentation

Fatbrain.com, an Internet professional bookstore, stocks select product documentation from Sun Microsystems, Inc.

For a list of documents and how to order them, visit the Sun Documentation Center on Fatbrain.com at:

`http://www.fatbrain.com/documentation/sun`

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 (806-3301-11) of your document in the subject line of your email.

Sun Remote System Control Software Overview

Sun Remote System Control (RSC) is a server management tool that enables you to monitor and control your server over modem lines and over a network. RSC provides remote system administration (sometimes called *lights-out management*) for geographically distributed or physically inaccessible systems.

The RSC software works with the alarm card for the Netra ct server. The alarm card supports serial and Ethernet connections to the Solaris console.

This chapter is organized as follows:

- “Accessing RSC” on page 1
- “RSC Capabilities” on page 3
- “Using RSC” on page 4
- “RSC Security” on page 6

Accessing RSC

You can access RSC from a workstation running the Solaris operating environment, from an ASCII terminal, or from a device running ASCII terminal emulation software. FIGURE 1-1 shows remote access paths to RSC.

The RSC firmware on the alarm card runs independently, and it uses standby power from the server. Therefore, alarm card hardware and RSC software continue to be effective when the server operating system goes offline; RSC 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 RS-232 ports.

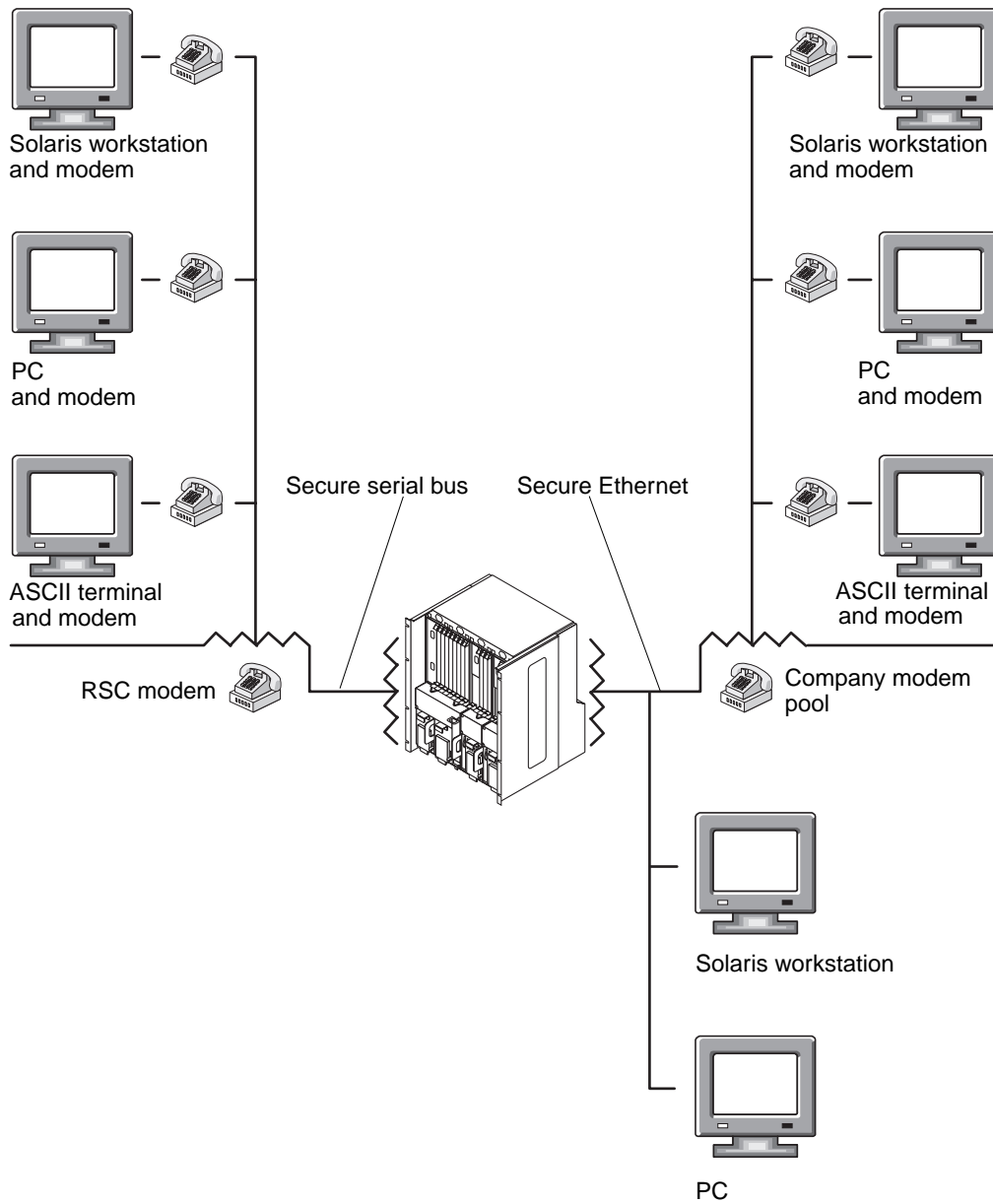


FIGURE 1-1 RSC Remote Access Paths

RSC Capabilities

Sun Remote System Control provides the following capabilities for remotely monitoring and controlling Sun Netra ct servers.

Server Status and Control

Server status and control features enable you to:

- Show server environmental status.
- Access a console that includes all UNIX console functions.
- Send a break to put the server into debug mode.
- Run diagnostic tests from a remote console.
- Monitor the system remotely and receive error reporting.
- 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.
- Receive remote event notification of server problems.

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.

View Logs

Logging capabilities enable you to:

- 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
- Alarms

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

Using RSC

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

After installation, you must run a script (`rsc-config`) that guides you through basic configuration (see Chapter 2 for more information). If you plan to use a modem on the RSC serial port, you must configure the modem properly. See Appendix A for more information on modem configuration.

RSC user interfaces include:

- A command-line interface you can access using standard telnet to the RSC Ethernet port, and to the RSC serial port COM2 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 COM1

RSC supports a maximum of four concurrent telnet connections per server.

Part of RSC configuration defines and enables *alert* mechanisms. Alerts provide remote notification of system problems.

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

- A server redundant power supply fails.
- A fan fails.
- 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.

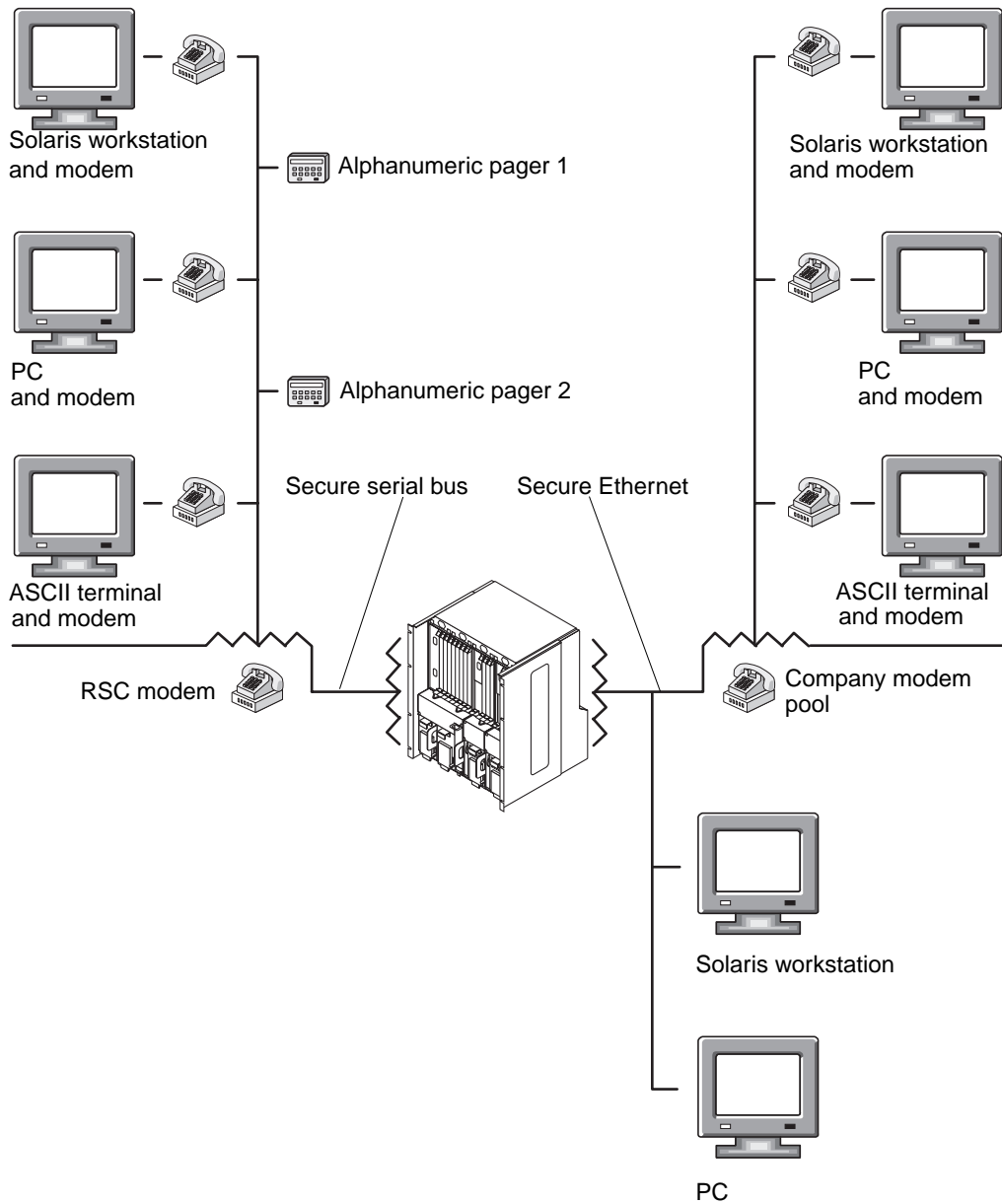


FIGURE 1-2 RSC Remote Alert Paths

When you receive an alert message, you can connect to your RSC account for the server that caused the alert. You can then 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 environment, 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
- Reset the server, optionally forcing a panic core dump
- Turn the 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 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. 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 example, the 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 10 minutes of inactivity; see “serial2_hw_handshake” on page 33. The time-out on serial port 1 is always set to disconnect a session after 10 minutes of inactivity

Note – As for any computer session, do not leave your session unattended. Always use the RSC `logout` command before disconnecting your session. In addition, if you initiated a server session, log out of it before logging out of RSC.

Configuring RSC Software

This chapter provides information on configuring RSC software, including:

- “RSC Software and the Alarm Card” on page 7
- “Verifying RSC Software Installation” on page 8
- “Accessing the Alarm Card and RSC” on page 8
- “Running the RSC Configuration Script” on page 9
- “Additional Configuration” on page 11
- “Redirecting the Console to RSC” on page 13
- “Backing Up the RSC Configuration” on page 15

RSC Software and the Alarm Card

The alarm card is shipped installed in the Netra ct server. One alarm card is supported per server.

The alarm card is supported by the Solaris 8 1/01 operating environment, which includes the appropriate software drivers for the card, and by an enhanced version of RSC 2.0 software for the alarm card. The RSC software is in the SUNWctac package, which is on the *Software Supplement for the Solaris 8 1/01 Operating Environment CD*.

The alarm card has hot-swap capability; information on hot swapping an alarm card is located in the *Netra ct Server Service Manual*.

Verifying RSC Software Installation

After the Solaris operating environment is installed on the Netra ct server, verify that the `SUNWctac` package is installed.

▼ To Verify RSC Software Installation

- As root on the server, run the `pkginfo` command.

```
# pkginfo SUNWctac
system      SUNWctac      Netra ct Alarm Card Firmware and Utilities
```

After RSC software is installed on the server, it resides in the directory `/usr/platform/SUNW,UltraSPARC-III-Netract/`.

If this package is not installed, you must install it from the *Software Supplement CD* that was shipped with the Solaris operating environment. If you need instructions on how to do this, refer to the *Netra ct Server Installation Guide* and the Solaris documentation that came with the Solaris CDs.

Note that an online version of the *Remote System Control (RSC) User's Guide* in the package `SUNWrsd` should be disregarded, as that version does not apply to the Netra ct server.

Accessing the Alarm Card and RSC

You can access and configure the alarm card over:

- The Ethernet port, using `telnet`
- A modem connection (terminal or PPP)
- The serial line (console), using an ASCII terminal or the `tip` program.

If you have a rear-access Netra ct server, to use the console, a cable should be connected to the rear serial port on the alarm card.

Running the RSC Configuration Script

After verifying the software installation, run the RSC configuration script (`rsc-config`) to configure RSC for the alarm card. After this initial configuration, you can change RSC configuration at any time (with appropriate user permissions) with any of these methods:

- Use RSC commands in the RSC shell
- Use the `rscadm` utility as root on the server
- Run the `rsc-config` script again as root on the server. The script updates your existing entries, except that it does not delete user names.

▼ To Run the RSC Configuration Script

Before running the `rsc-config` script, make sure the environmental monitoring daemon (`envmond`) is stopped. See “Using the Environmental Monitoring Daemon” on page 11 for information on `envmond`.

1. As root, enter the following command:

```
# /usr/platform/SUNW,UltraSPARC-IIIi-Netract/rsc/rsc-config
```

The `rsc-config` script prompts you for the information shown in TABLE 2-1. Certain information is required the first time you run the script.

TABLE 2-1 Information for RSC Configuration Script

Information	Description
Server hostname (required)	The Netra ct server host name.
Customer information	The customer information string identifies the server in any alert messages. The string can contain 40 characters maximum, including alphanumeric characters and the hyphen character.
RSC Ethernet Interface	If you enable the Ethernet connection to RSC, you must provide the IP mode, IP address, IP netmask, and IP gateway.
RSC IP Mode	Set to <code>config</code> (the default), to <code>DHCP</code> , or to <code>none</code> . Choose the <code>config</code> mode to configure manually and enter Internet addresses or choose <code>DHCP</code> if you want to use the dynamic host configuration protocol. Choose <code>none</code> to leave the Ethernet connection disabled.

TABLE 2-1 Information for RSC Configuration Script (Continued)

Information	Description
RSC IP Address	The IP address for RSC. You must provide the address if you do not use the DHCP mode.
RSC IP Netmask	The subnet mask. You must provide the subnet mask if you do not use the DHCP mode.
RSC IP Gateway	The IP address of the default gateway to use when the destination is not on the same subnet as RSC.
RSC Alert use	If you enable alerts, the alerts can be sent to an email address, to a pager, or to both.
Email alerts	If you want alerts sent to an email address, provide the SMTP server IP address and the email address you want used.
Pager alerts	If you want alerts sent to a pager, provide the telephone number, modem initialization string, account password, baud rate, data bits, parity, and stop bits for the pager. You can designate one or two pagers to receive the alerts.
RSC Modem Interface	If you enable the modem interface, provide the PPP local and remote IP addresses if you are using PPP, and provide the serial port 2 baud rate, data bits, parity, and stop bits.
User account (required)	Add the RSC user account for superuser, or root. With the appropriate permissions, this account enables the user to use the RSC shell.
User name (required)	The user name can be a maximum length of 16 characters, it must contain at least one lowercase alphabetic character, and the first character must be alphabetic. The user name can contain alphanumeric characters and the period, underscore, and hyphen characters.
User permissions (required)	Choose the default, <code>cuar</code> , as this will be the only user account until you add more accounts. The account with all four permissions (console, user, admin, reset) is considered the superuser, or root, account.

2. Enter your information when prompted by the script.

The `rsc-config` script displays a summary for each section and asks whether it is correct.

3. For each section, respond y (for yes) to confirm your choices or respond n (for no) to return to that section of the configuration script.

When you have confirmed all sections, the configuration script updates the RSC flash PROM. The configuration information will remain on the RSC flash PROM even if you reinstall the Solaris operating environment.

Note – It takes a few minutes to update the RSC flash PROM. Do not interrupt this process by aborting the configuration procedure or removing system power. If the update is interrupted, you must restart the configuration script and it must complete successfully before RSC can function properly.

The configuration script prompts you for the password for the user account.

4. Enter the password for the user account.

The configuration script prompts you to reenter the password for the user account.

5. Reenter the password for the user account.

6. If you enabled the Ethernet connection to RSC, the script asks if you want to reset RSC.

A reset is required before Ethernet configuration takes effect.

If you are using the environmental monitoring daemon, start the daemon.

After initial configuration, you can control configuration and add or modify user accounts by using the RSC shell commands (see Chapter 3). You can also log in to the server as root and use the `rscadm` utility to control RSC configuration and user accounts (see Chapter 4).

Additional Configuration

Using the Environmental Monitoring Daemon

If you want to use the environmental monitoring daemon (`envmond`) to monitor the server and send messages and *heartbeats* to the alarm card, make sure the daemon is installed and that it is running.

▼ To Verify `envmond` Installation

- As root, run the `pkginfo` command.

```
# pkginfo SUNWcteux SUNWctevx
system    SUNWcteux    Netra ct EnvMon Daemon and FRU policy (Usr) (64-bit)
system    SUNWctevx    Netra ct Environment Monitor Daemon and FRU policy (64-bit)
```

▼ To Verify envmond Is Running

- As root, run the `pgrep` command.

```
# pgrep -l envmond
194 envmond
```

If you want to run the `rsc-config` script, you must stop the daemon.

▼ To Stop and Start envmond

1. As root, run the following command to stop `envmond`:

```
# /etc/init.d/envmon stop
```

2. As root, run the following command to start `envmond`:

```
# /etc/init.d/envmon start
```

For more information on using `envmond`, see “Environmental Monitoring and Alarms” on page 39.

Configuring PPP

If point-to-point protocol (PPP) is enabled on serial port COM2, RSC supports multiple shell sessions through one modem connection. If PPP is not enabled, only one shell session can be run over the modem. 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`. 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.

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 PC client, see your PC 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. 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.

Configuring Alerts

To configure alerts, set the following configuration variables using the RSC shell or `rscadm` utility:

- `customerinfo`
- `hostname`
- `page_enabled`
- `page_infol`
- `page_initl`
- `page_baudl`
- `page_stopl`
- `page_parityl`
- `page_passwordl`
- `mail_enabled`
- `mailuser`
- `mailhost`

For help configuring RSC alert variables, see “Alert Variables” on page 34. RSC generates alert messages with the following format:

```
$EVENT $TIME $CUSTOMERINFO $HOSTNAME $IPADDR message
```

You can also use the `rscadm` subcommand `send_event -c` to send an alert. You can do this directly at the superuser prompt (see Chapter 4, section “Turning on an Alert at a UNIX Prompt” on page 49); you can also create a command file that runs and sends the alert under special circumstances (see Appendix B for an example of a script to do this).

Redirecting the Console to RSC

Until RSC software is configured, the system console is available as on any normal Solaris server. You can set the input and output devices to RSC (redirect the console to RSC) to allow the alarm card to connect to the server (start a console session) from the RSC shell, and to create panic dumps and console logs of the server.



Caution – The console is directed to the `ttya` port (also known as the COM 1 port) on the CPU card or CPU transition card by default. If the console has been redirected to RSC, the alarm card is *not* hot-swappable without a system reboot. Refer to the *Netra ct Server Service Manual* for more information.

▼ To Redirect the Console to RSC

To enable RSC as the system console device, you must connect to the console port, and either:

- Use the `eeeprom` command at a UNIX prompt to set the input and output devices:

```
# eeeprom input-device=rsc
# eeeprom output-device=rsc
```

or

- Go to the OBP environment, and use the `setenv` command at the `ok` prompt to set the input and output devices:

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

These commands take effect after the next server reset.

▼ To Redirect the Console From RSC to the CPU Serial Port

At any time, to remove RSC as the default console and redirect the console from RSC to the CPU console, you can connect to the alarm card's console port, and either:

- Use the `eeeprom` command at a UNIX prompt to set the input and output devices:

```
# eeeprom input-device=ttya
# eeeprom output-device=ttya
```

or

- Go to the OBP environment, and use the `setenv` command at the `ok` prompt to set the input and output devices:

```
ok setenv input-device ttya
ok setenv output-device ttya
```

These commands take effect after the next server reset.

Console Session Information

See “console” on page 21, “break” on page 21, and “escape_char” on page 42, for information about using these commands and variables in a console session.

Backing Up the 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. To restore the settings, use the `rscadm set` command. For example,

```
# rscadm set < remote-filename
#
```

Note that the `set` command restores configuration settings, but not user accounts.

Using the RSC Command Shell

The RSC command shell is a command-line interface that supports commands that enable you to administer or diagnose the server. It also has commands for configuring RSC. RSC supports a maximum of four concurrent telnet connections per server.

This chapter describes:

- “Logging In to Your RSC Account” on page 17
- “RSC Command Overview” on page 19
- “Server Status and Control Commands” on page 20
- “RSC View Log Commands” on page 22
- “RSC Configuration Commands” on page 25
- “Other RSC Commands” on page 30
- “RSC Configuration Variables” on page 30

Note – If you need help with RSC commands, type `rscadm help` in the console window or type `help` at the `hostname rsc>` prompt.

Logging In to Your RSC Account

After RSC software is installed and configured and an account has been set up for you (by using either the `rsc-config` script or the `rscadm useradd` command), you can connect to RSC and log in to your account using a Solaris workstation, standard ASCII character terminal, or a computer running ASCII terminal emulation software.

▼ To Log In to Your RSC Account

1. Connect to RSC using one of these methods:

- a. If you are already connected to your company Ethernet, use the `telnet` command to connect to RSC.
- b. Use PPP to connect to the RSC modem.
To use this option, PPP must be enabled on serial port COM2.
- c. If PPP is not enabled on the serial port, dial in to the RSC modem.
- d. Connect your client machine directly to the RSC serial port.

When connection is established, the following screen appears:

```
RSC version 2.0 (hostname)
Please login:
```

2. Enter your RSC login name.

After you enter your RSC login name, you are prompted for your password.

```
Please enter password:
```

3. Enter your RSC password.

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

```
hostname rsc>
```

You can enter RSC shell commands at the `hostname 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 10 minutes of inactivity; see “`serial2_hw_handshake`” on page 33. Serial port COM1 has an idle time-out of 10 minutes enabled; idle time-out is not enabled if you are using the `console` command.

RSC Command Overview

The following table summarizes RSC shell commands.

TABLE 3-1 RSC Shell Commands

Type of Command	Command	Description
Server status and control	<code>environment</code>	Displays current environmental information
	<code>shownetwork</code>	Displays the current network configuration
	<code>console</code>	Connects to the server console
	<code>break</code>	Puts the server in debug mode
	<code>xir</code>	Generates an externally initiated reset to the server
	<code>reset</code>	Resets the server immediately
	<code>poweroff</code>	Powers off the server
	<code>poweron</code>	Powers on the server
	<code>powersupply</code>	Deactivates or activates a power supply unit on the server
	<code>alarm</code>	Sets alarm relays to allow notification of events
View logs	<code>loghistory</code>	Displays the history of all events logged in the RSC event buffer
	<code>consolehistory</code>	Displays the history of all console messages logged in the buffer
	<code>consolerestart</code>	Makes the current <code>run</code> console log the <code>orun</code> log
Configuration	<code>set</code>	Sets a configuration variable
	<code>show</code>	Displays one or more configuration variables
	<code>date</code>	Displays or sets the current time and date
	<code>password</code>	Changes your RSC password
	<code>useradd</code>	Adds an RSC user account
	<code>userdel</code>	Deletes an RSC user account
	<code>usershow</code>	Shows characteristics of an RSC user account
	<code>userpassword</code>	Sets or changes a user's password
	<code>userperm</code>	Sets the authorization for a user
	<code>resetrsc</code>	Resets RSC, after prompting for confirmation

TABLE 3-1 RSC Shell Commands (Continued)

Type of Command	Command	Description
Other	help	Displays a list of RSC shell commands and a brief description of each
	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 28 for information about user permission levels.

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

Server Status and Control Commands

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

environment

Use the `environment` command to display a snapshot of server environmental status, such as temperatures, power supply status, active alarms, and so forth.

shownetwork

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

```
hostname 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
hostname 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. You use an escape character sequence to return to the RSC prompt. The default escape sequence is `~.` (tilde period). For more information, see “`escape_char`” on page 42.

break

Use the `break` command to put the server into debug mode. You must have `C` level user permission to use this command. Debug mode can use either `kadb` or `OBP`. Issuing a `:c` in `kadb` or `go` in `OBP` causes the server to resume execution (if possible); otherwise, you would normally reboot the server after issuing the `break` command.

Use the `break` command, then the `console` command, to take control of the server.

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.

Use the `xir` command, then the `console` command, to take control of the server.

reset

This command forcibly resets the server. You must have `R` level user permission to use this command. If the `panic_dump` variable is set to `true`, RSC attempts to create a panic core dump on the 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 the server is already powered on.

powersupply *n* [on|off]

The `powersupply` command enables you to deactivate a power supply on the server. Power supply 1 or power supply 2 can be specified. You must have `R` level user permission to use this command.

alarm *n* [on|off]

Use the `alarm` command to set external alarm relays 0, 1, 2, or 3 on the alarm card to on or off. This command is useful for testing purposes. You must have `R` level user permission to use this command. You can view the current status of alarms with the `environment` command.

If `envmond` is running, alarm 0 is immediately cleared by the heartbeat. See “Alarm Variables” on page 38 for more information on alarms.

RSC View Log Commands

The following RSC commands work with RSC and console log files.

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:

```
hostname 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:

```
JAN 01 07:33:03 sst4828: 00060003: "RSC System booted"
```

```
consolehistory [run|orun]
[index [+|-]n] [pause 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 two console logs:

- The `run` buffer contains the most recent data received from the server operating system.
- If the system panics and resets, the `orun` buffer will contain messages printed to the console before the reboot, which are the panic messages.

The `run` and `orun` buffers can contain up to 16 Kbytes of information.

RSC always writes to the `run` (`run`) log. When the `run` log fills up, it overwrites old data in the `run` log.

When RSC senses a server reset or when the `consolerestart` command is issued, RSC stores the contents of the current `run` log in the `orun` log. The `run` log is cleared and all further server operating system messages are stored in the `run` log.

```
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 23 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 47.

consolerestart

Use the `consolerestart` command to copy the current run log into the old log buffer (designated `orun`). This command copies the current `run` buffer to the `orun` buffer, overwriting the previous contents. Then it clears the current `run` buffer. You must have A level user permission to use this command.

RSC Configuration Commands

RSC configuration commands set or show characteristics of the RSC or server configuration:

`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 30 for descriptions of these variables.

Changes to some variables do not take effect until the alarm card 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:

```
hostname rsc> set page_info2 ""
hostname 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 30 for descriptions of these variables.

`date [[mmdd]HHMM | mdddHHMM[cc]yy] [.SS]`

Use the `date` command without arguments to show RSC's current date and time. For example:

```
hostname rsc> date
TUE FEB 22 10:29:58 2000
hostname rsc>
```

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 3-2 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 can omit the month, day, and year; the current values are applied as defaults.

Examples:

```
hostname rsc> date 031521452000
hostname rsc> date 03152145
hostname rsc> date 2145
```

The first example sets the time to March 15, 9:45 p.m., 2000. The second example sets the time to March 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. To keep RSC time in sync with server time, 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:

```
hostname rsc> password
password: Changing password for username
Enter login password:
Enter new password:
Re-enter new password:
hostname 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 16 characters, it 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 user name, permissions, and whether a password is assigned. For example:

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

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

```
hostname rsc> version
RSC Bootmon version: 1.x
RSC version: 2.x
hostname rsc> version -v
RSC Bootmon version: 1.x
RSC Bootmon checksum: A91695A9
RSC version: 2.x
RSC firmware checksum: 4FF151DA
RSC firmware built Feb 15 2000, 16:14:30
...
hostname rsc>
```

logout

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

RSC Configuration Variables

RSC has nonvolatile configuration variables you can use to change RSC behavior. The `rsc-config` script sets most configuration variables during installation. After installation, you can use the RSC shell or the `rscadm` utility (see Chapter 4) to set or change configuration variables.

This section describes these types of RSC variables:

- Serial port variables
- Alert variables
- Alarm variables
- Ethernet port variables
- Console session variable
- Server variable

▼ To Set a Configuration Variable in the RSC Command Shell

You must have level A user permission to set configuration variables in the RSC shell.

1. **Log in to your RSC account.**
2. **Use the `set` command with the variable and its value that you want to set:**

```
hostname rsc> set variable value
```

▼ To Set a Configuration Variable in the `rscadm` Utility

1. **Log into the server as root.**
2. **Use the `rscadm set` command with the variable and its value that you want to set:**

```
# rscadm set variable value
```

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

The alarm card has two serial ports. Serial port 1 is locked at 9600 baud, 8 data bits, 1 stop bit, no parity; it cannot be changed. Serial port 2 is configurable, and has two modes that can be set using the `serial2_mode` variable.

`serial2_mode`

When the `serial2_mode` variable is set to `modem` or `tty` (terminal mode), you can specify values for the variables described in this section. The default setting is `disabled`.

`serial2_baud`

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

- 300
- 1200
- 2400
- 4800
- 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 Appendix A).

`serial2_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 Appendix A).

`serial2_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 Appendix A).

`serial2_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 Appendix A).

`serial2_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`, and if the hardware supports hardware flow control, hardware flow control and modem control are enabled. If set to `false`, or if the hardware does not support hardware flow control, RSC disconnects a session connected to the serial port after 10 minutes of inactivity.

The following PPP variables apply to serial port 2.

`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

SNMP Alerts

Simple Network Management Protocol (SNMP) allows you to control RSC from a remote host. SNMP is insecure; there are no user permissions associated with it. Use SNMP only if you have RSC on a secure LAN.

`snmp_enabled`

Use this variable to specify whether SNMP will be used for RSC configuration and alerts. Valid values are `true` or `false`; the default is `false`.

`snmp_host`

This variable contains the Internet address to which RSC sends SNMP trap alerts. Specify the address using standard dot notation. For example:

```
hostname rsc> set snmp_host 139.143.4.2
```

If this variable is empty, or if the `snmp_enabled` variable is set to `false`, no alert is sent. The default setting is empty.

Paging and Mail Alerts

`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 40 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 and in the `rsc>` prompt. The hostname is set with `rscadm set hostname`. If the hostname is changed, current sessions keep the old hostname prompt until the user logs out and logs back in. This string can be up to 40 characters, including alphanumeric characters and hyphen (-). The default setting is empty.

`page_enabled`

Use this variable to specify whether paging 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)

For example:

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

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.

`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 `serial2_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 `serial2_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 `serial2_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 `serial2_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)

For example:

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

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.

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 `serial2_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 `serial2_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 `serial2_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 `serial2_stop` variable.

mail_enabled

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

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:

```
hostname 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.

Alarm Variables

Alarm variables work with the environmental monitoring daemon (`envmond`) to provide alarming.

Environmental Monitoring and Alarms

The `envmond` daemon runs on the server connected to RSC; it is responsible for the main interaction between the Solaris operating environment and the alarm card. `envmond` monitors the server and sends the alarm card:

- Status messages (the `alarmn_prefix` variables determine whether an alarm should be turned on or off as a result of the messages)
- A “heartbeat” every second (the `watchdog` variables determine the alarm card’s reaction to missed heartbeats)

The alarm card, based on values set in the `alarmn_prefix` variables, determines whether to turn on an alarm; if an alarm is turned on, the alarm card sends a notification, for example, by email or pager, depending on the alert settings.

There are four relays on the external alarm port: 0, 1, 2, and 3. Relay 0 is reserved for the system; you can set the `alarmn_prefix` values for relays 1, 2, and 3. The *n* in an `alarmn_prefix` variable corresponds to the same alarm port number.

Use the `environment` command to view the status of alarms.

Messages

The messages generated by `envmond` have six digits, in the format shown in FIGURE 3-1.

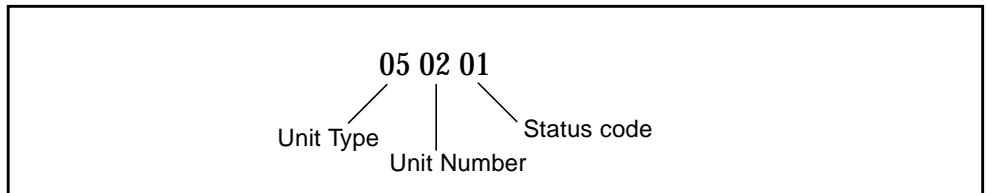


FIGURE 3-1 Message Format

Valid values for the unit type are:

Unit Type	Description
03	Power Supply
04	Temperature Sensor
05	Fan
90-99	User-specified unit

Use the `alarmn_prefix` variables to specify the unit type.

Values for the unit number can be 01 or 02, for example, Power Supply 01.

Values for the status codes are:

Status Code	Description
00	OK
01	Failed
02	Present, but status unknown
10	Not present
99	Informational message only, no status update

Thus, the message depicted in FIGURE 3-1 means, "Fan 2 failed."

`alarm1_prefix`

Use this variable to specify the unit type for messages that will activate alarm relay 1. The default value is 03 (power supply). Other valid values are 04 (temperature), 05 (fan), and a number in the range 90 through 99.

`alarm2_prefix`

Use this variable to specify the unit type for messages that will activate alarm relay 2. The default value is 04 (temperature). Other valid values are 03 (power supply), 05 (fan), and a number in the range 90 through 99.

`alarm3_prefix`

Use this variable to specify the unit type for messages that will activate alarm relay 3. The default value is 05 (fan). Other valid values are 03 (power supply), 04 (temperature), and a number in the range 90 through 99.

`solaris_watchdog_timeout`

The number of seconds before the alarm card determines that the system is hung or has failed. If `envmond` is running, the daemon sets this variable to 10 seconds by default. If this variable is set to 0, the alarm card ignores any heartbeats that arrive from `envmond`. If this variable is set to a positive value, the alarm card notices if a

heartbeat from `envmond` has not arrived in time. A time-out indicates a system failure, and an alarm relay is activated; a system failure message is also sent through event notification to email or a pager.

`solaris_watchdog_reboot`

Use this variable to specify whether the alarm card should attempt to reset a system that is hung or has failed. Valid values are `true` or `false`. The default is `false`. See also “`panic_dump`” on page 43.

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.

When the alarm card is first installed in the system, the Ethernet interface is disabled (the `ip_mode` is set to the default of `none`). When you run the `rsc-config` script, the script prompts you to configure the Ethernet interface by displaying the default for `ip_mode` as `config`. Changes to this variable take effect after the next RSC reset.

`ip_addr`

This variable is used only 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 used only 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.

`tpe_link_test`

Use this variable to specify whether 10BASE-T Ethernet Link Integrity tests are enabled. Valid values are `true` or `false`; the default is `true`. The `false` value should be used for installations where RSC is located in a hub that does not support Link Integrity tests or that has the Link Integrity tests disabled. Link Integrity tests should have consistent settings for both RSC and the local hub; if the settings are not consistent, communication may not be possible.

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).

When you change the escape sequence character, your current console session is not impacted; the new escape sequence character is valid only for future console sessions. Use the `show` command to display the current escape sequence character.

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, followed by a period (`.`), to end a console session.

If you are using `tip` to administer your system, the system will not recognize the default escape sequence of `~.` (tilde period). Use the following command to change the escape sequence to another character if you are using `tip` to administer your system:

```
# rscadm set escape_char new_character
```

where `new_character` is the new escape sequence character that you want to use. For example, to change the escape sequence character to `#` (pound), enter:

```
# rscadm set escape_char #
```

Server Variable

`panic_dump`

Use this variable to specify whether the server will generate a core dump when it is reset by RSC. RSC resets the server when the `reset` RSC shell command is issued or when the `solaris_watchdog_reboot` variable is set to `true` and heartbeats are not received in `solaris_watchdog_timeout` seconds. Valid values are `true` or `false`. The default is `false`.

Using the `rscadm` Utility

This chapter describes the `rscadm` utility. It contains these sections:

- “Overview of the Utility” on page 45
- “`rscadm` Subcommands” on page 46
- “Turning on an Alert at a UNIX Prompt” on page 49

Overview of the Utility

The `rscadm` utility and its subcommands enable 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 is installed in the directory
`/usr/platform/SUNW,UltraSPARC-III-Netractor/rsc.`

The `rscadm` utility uses the following subcommands.

TABLE 4-1 `rscadm` Utility Subcommands

Subcommand	Description
<code>help</code>	Displays a list of <code>rscadm</code> commands and brief descriptions for each
<code>version</code>	Displays information about the firmware running on RSC
<code>date</code>	Displays or sets the current time and date
<code>set</code>	Sets a configuration variable
<code>show</code>	Displays one or more configuration variables
<code>resetrsc</code>	Resets RSC immediately
<code>download</code>	Downloads firmware to the RSC flash PROM
<code>send_event</code>	Logs an event; can also send an alert message
<code>modem_setup</code>	Changes configuration of the modem connected to the RSC serial port
<code>useradd</code>	Adds an RSC user account
<code>userdel</code>	Deletes an RSC user account
<code>usershow</code>	Shows characteristics of an RSC user account
<code>userpassword</code>	Sets or changes a user's password
<code>userperm</code>	Sets the authorization for a user

`rscadm` Subcommands

Use the `rscadm` subcommands with the `rscadm` utility on the server. For example:

```
# rscadm show
```

`help`

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

version

The `version` subcommand displays information about the firmware running on RSC. For example:

```
# rscadm version
RSC Version v2.0
RSC Bootmon v2.0
RSC Main    v2.0.15
RSC Post Status = 0xFFFF
#
```

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 26 for a description of date formats.

set *variable value*

Use the `set` subcommand to set an RSC configuration variable. See “RSC Configuration Variables” on page 30 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 30 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* are installed in the boot section of RSC non-volatile memory. When the transfer is completed, the alarm card performs a self-test; `envmond` reinitializes itself after the alarm card reports the self-test completed successfully.

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 B for an example of a Perl script that uses this subcommand to send an alert, and see “Turning on an Alert at a UNIX Prompt” on page 49 for an example of using this subcommand at a UNIX prompt.

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. If you are using `tip` to administer your system, and you run `rscadm modem_setup` over this connection, the `tip` connection recognizes the `~` escape sequence and breaks the `tip` connection. The `rscadm modem_setup` process continues to run.

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 27)
- `userdel` (see “`userdel username`” on page 28)
- `usershow` (see “`usershow [username]`” on page 28)
- `userpassword` (see “`userpassword username`” on page 28)
- `userperm` (see “`userperm username [c][u][a][r]`” on page 28)

Turning on an Alert at a UNIX Prompt

This section gives an example of how to turn on an alert at a UNIX prompt using `rscadm` subcommands. This example configures the alarm card to turn on an alarm and send a notification in email.

1. Configure the alarm card for email notification:

```
# rscadm set mail_enabled true
# rscadm set mailhost 139.143.4.2
# rscadm set mailuser joe@company.com
```

2. Configure an alarm relay, in this example, alarm 3:

```
# rscadm set alarm3_prefix 91
```

3. Send alert messages:

```
# rscadm send_event -c "910100 Process is now being monitored"  
# rscadm send_event -c "910101 Process failed"  
# rscadm send_event -c "910100 Process OK"  
# rscadm send_event -c "910110 Process no longer monitored"
```

When the “failed” event is received, the alarm card:

1. Turns on alarm relay 3.
2. Logs the failure to the alarm card console, logs, and environment command.
3. Sends email notification to the mailuser specified.
4. Sends an SNMP trap (if `snmp_enabled` and `snmp_host` variables have been set).

When the “OK” event is received, the alarm card:

1. Turns off alarm relay 3.
2. Logs the OK event to the alarm card console, logs, and environment command.
3. Sends email notification to the mailuser specified.
4. Sends an SNMP trap (if `snmp_enabled` and `snmp_host` variables have been set).

Using OpenBoot PROM Features that Support RSC

This chapter contains information about OpenBoot PROM features that support RSC.

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
ok setenv output-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. `ttya` output and input are typically still set to be the default, and `keyboard` and `ttyb` continue to be other valid options.

Troubleshooting

This chapter includes information about troubleshooting these problems:

- “Troubleshooting RSC Problems” on page 53
 - “Troubleshooting Server Problems Using RSC” on page 56
-

Troubleshooting RSC Problems

Cannot log in to RSC

- Check the RSC device name you are connecting to (normally *servername-rsc*).
- Check that you are using your correct RSC user name; it might not be the same as your system user name.
- Check that you are using your correct RSC password.

Cannot connect to RSC using telnet

RSC supports a maximum of four simultaneous telnet connections. When all 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.
```

Cannot connect to RSC through the Ethernet connection

First, log in to the server as root and see if the command `rscadm version` reports a good status. If it does, 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 `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 alarm card hardware.

Cannot connect to RSC through the serial port

First, log in to the server as root and see if the command `rscadm version` reports a good status. If it does, RSC is working. 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.
- Run SunVTS diagnostics to check the serial port; the external serial port test requires a serial loopback connector.
- Run SunVTS diagnostics to check the alarm card hardware.

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.

If SNMP traps are not received:

- Check that the `snmp_enabled` variable is set to true.
- Try issuing the `ping` command for the IP address in the `snmp_host` variable.
- Check the SNMP management station to be sure it can receive traps from other hosts.

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.

Some RSC functions work, but others do not

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

- The `poweroff` command has no effect: The server is already powered off.
- The `poweron` command has no effect: The server is already powered on.

RSC is not working after you swap the alarm card hardware

After replacing an alarm card, you must reconfigure RSC. Use the `rscadm` utility to restore configuration settings from a backup file and manually add users.

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.

Configuring Alarm Card Modems

This appendix provides information on alarm card modems. It is organized as follows:

- “External Modems” on page 57
- “Troubleshooting Modem Problems” on page 60

External Modems

You can access RSC using an Ethernet connection, and you can also install a dedicated modem connected to the RSC serial port 2 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. To enable and set up the RSC modem, use the following configuration settings unless otherwise indicated. 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 2.

TABLE A-1 Serial Port 2 General Settings

GUI Setting	Configuration Variable Setting
Serial Port 2 Settings	
Baud Rate: 9600	<code>serial2_baud 9600</code>
Parity: None	<code>serial2_parity none</code>
Stop Bits: One	<code>serial2_stop 1</code>
Data Bits: 8	<code>serial2_data 8</code>
Enable hardware handshaking	<code>serial2_hw_handshake true</code>
Enable PPP on serial port	<code>ppp_enabled true</code>

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 2 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.

TABLE A-2 Pager Alert Settings

Setting	Configuration Variable Setting
Send RSC alerts to pagers	<code>page_enabled true</code>
Number: <i>phone number to dial</i> PIN: <i>pager user's PIN</i>	<code>page_info1 <i>number@ID</i></code>

TABLE A-2 Pager Alert Settings (Continued)

Setting	Configuration Variable Setting
Baud Rate: 9600	<code>page_baud1 9600</code>
Parity: even	<code>page_parity1 even</code>
Stop Bits: 1	<code>page_stop1 1</code>
Data Bits: 7	<code>page_data1 7</code>
Password: <i>password</i>	<code>page_password1 <i>password</i></code>
Modem Init. String: see "Modem Initialization Strings" on page 60	<code>page_init1</code>
Number: <i>phone number to dial</i>	
PIN: <i>pager user's PIN</i>	<code>page_info2 <i>number@ID</i></code>
Baud Rate: 9600	<code>page_baud2 9600</code>
Parity: even	<code>page_parity2 even</code>
Stop Bits: 1	<code>page_stop2 1</code>
Data Bits: 7	<code>page_data2 7</code>
Password: <i>password</i>	<code>page_password2 <i>password</i></code>
Modem Init. String: see "Modem Initialization Strings" on page 60	<code>page_init2</code>

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 consist of attention (AT) commands. 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 2 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.

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.

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 2 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.

Sample Script for 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.

CODE EXAMPLE B-1

```
#!/usr/bin/perl
# Disk Monitor
# USAGE: dmon <mount> <percent>
# e.g.: dmon /usr 80
@notify_cmd = '/usr/platform/SUNW,UltraSPARC-IIi-Netrtract/rsc/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($fields);
    ($fs, $size, $used, $avail, $capacity, $mount) = split(' ', $fields);
    if ($ARGV[0] eq $mount)
    {
        $found = 1;
    }
}
```

CODE EXAMPLE B-1 (Continued)

```
if ($capacity > $ARGV[1])
{
    print STDERR "ALERT: \"", $mount, "\" is at ", $capacity,
        " of capacity, sending notification\n";
    $notify_msg = 'mount point "'. $mount. "' is at '.
        $capacity.' of capacity';
    exec (@notify_cmd, 'send_event', '-c', $notify_msg)
        || die "ERROR: $!\n";
}
}
}
if ($found != 1)
{
    print STDERR "ERROR: \"", $ARGV[0],
        "\" is not a valid mount point\n\n";
}
close(DF);
```

Error Messages

This appendix provides information on error messages, and it is organized as follows:

- “Error Messages at the `rsc>` Prompt” on page 63
- “Error Messages From the `rscadm` Utility” on page 65

Error Messages at the `rsc>` Prompt

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 user name.

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].

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 user name you specified is not associated with an RSC account on this server.

Error Messages From the `rscadm` Utility

Passwords didn't match, try again

During execution of the `userpassword` subcommand, you enter the password twice. If the 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 EEPROM 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: the RSC lock file was found. Only one instance of rscadm can run at a given time.
As stated.

rscadm: the RSC hardware was not detected
As stated.

rscadm: the RSC hardware could not be initialized
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.

rscadm: invalid variable or value

During execution of the `set` subcommand, you entered an invalid variable or value.

rscadm: malformed password

You entered an invalid password. A valid password has between six and eight 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 user name.

rscadm: maximum username length is - 16

The user name you entered exceeded 16 characters; the maximum length for a user name is 16 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 user name format when trying to add an RSC user account.

rscadm: username does not exist

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

rsc-config cannot be run while envmond is running

envmond must be stopped before you can run the rsc-config program.

This program **MUST** be run as root

As stated.

Index

A

- alarm card, 7
- alarm command, 22
- alarming, 38
- alert messages, 4
- alerts
 - configuring, 13, 34
 - creating a script to send, 61
 - format, 13
 - troubleshooting, 54
 - variables, 34

B

- backing up configuration, 15
- break command, 21

C

- command shell
 - configuration commands, 25
 - configuration variables, 30
 - alert, 34
 - console, 42
 - Ethernet port, 41
 - serial port, 32
 - logging in to, 17
 - other commands, 30
 - server status and control commands, 20
 - view log commands, 22

- commands, shell, 17
 - alarm, 22
 - break, 21
 - configuration commands, 25
 - configuration variables, 30
 - alert, 34
 - console, 42
 - Ethernet port, 41
 - serial port, 32
 - console, 21
 - consolehistory (chist), 24
 - consolerestart, 25
 - date, 26
 - environment, 20
 - help, 30
 - loghistory (lhist), 23
 - logout, 6, 30
 - other commands, 30
 - password, 27
 - poweroff, 22
 - poweron, 22
 - powersupply, 22
 - reset, 21
 - resetrsc, 29
 - server status and control, 20
 - set, 25
 - show, 25
 - shownetwork, 20
 - table, 19
 - useradd, 27
 - userdel, 28
 - userpassword, 28
 - userperm, 28
 - usershow, 28

- version, 30
- view log commands, 22
- xir, 21
- configuration
 - backing up, 15
 - point-to-point protocol (PPP), 12
- configuration script, 9
- configuration variables, 30
 - alarm
 - alarm1_prefix, 40
 - alarm2_prefix, 40
 - alarm3_prefix, 40
 - solaris_watchdog_reboot, 41
 - solaris_watchdog_timeout, 40
 - alert, 34
 - customerinfo, 35
 - hostname, 35
 - mail_enabled, 38
 - mailhost, 38
 - mailuser, 38
 - page_baud1, 36
 - page_baud2, 37
 - page_data1, 36
 - page_data2, 37
 - page_enabled, 35
 - page_info1, 35
 - page_info2, 37
 - page_init1, 36
 - page_init2, 37
 - page_parity1, 36
 - page_parity2, 38
 - page_password1, 36
 - page_password2, 37
 - page_stop1, 36
 - page_stop2, 38
 - snmp_enabled, 34
 - snmp_host, 34
- console, 42
 - escape_char, 42
- Ethernet port, 41
 - ip_addr, 41
 - ip_gateway, 42
 - ip_mode, 41
 - ip_netmask, 41
 - tpe_link_test, 42
- serial port, 32
 - ppp_enabled, 34
 - ppp_local_ip_addr, 33
 - ppp_remote_ip_addr, 34
 - serial2_baud, 32
 - serial2_data, 33
 - serial2_hw_handshake, 33
 - serial2_mode, 32
 - serial2_parity, 32
 - serial2_stop, 33
 - server
 - panic_dump, 43
- console, 21, 42, 51
 - redirecting, 13 to 15
- console command, 21
- console log, 3, 22
- consolehistory (chist) command, 24
- consolerestart command, 25
- customerinfo variable, 13

D

- date command, 26
- DHCP, 9
- documentation, xiii, xiv
- drivers, for alarm card, 7

E

- eprom command, 14
- environment command, 20
- environmental monitoring daemon, *see* envmond
- envmond, 11 to 12, 22, 38 to 41, 48
- error messages, 63
- escape character, 42, 49
- event log, 3
 - creating a script to write to, 61

F

- fan, 40

G

- gateway, 10

H

help command, 17, 30
hostname, 9
hostname variable, 13, 35
hot swap, 7, 14

I

inactivity time-out, 6, 33
input-device variable, 13, 51
installation directory, 8
IP address, 10, 20, 41

L

link integrity tests, 42
loghistory (lhist) command, 23
login failures, 18
logout command, 6, 30

M

mail_enabled variable, 13
modem
 example of general settings, 58
 multiple sessions, 12
 pager variable settings, 58
 setup, 48 to 49, 57 to 60

N

netmask, 10

O

OpenBoot PROM (OBP), 14, 15
 environment variable properties for RSC
 rsc, 51
output-device variable, 13, 51

P

page_enabled variable, 13
page_infol variable, 13
password, 11, 55
password command, 27
point-to-point protocol (PPP)
 configuration, 12
 ppp_enabled variable, 12
power supply, 20, 40
poweroff command, 22
poweron command, 22
powersupply command, 22
ppp_enabled variable, 12
PROM, 10

R

Remote System Control
 capabilities, 3
 configuration, backing up, 15
 configuring, 7, 9
 flash PROM, 10
 installing, 7
 login failures, 18
 overview, 1
 security, 6
 using, 4
reset command, 21
resetrsc command, 29
root account, 10
rscadm date, 47
rscadm download, 48
rscadm help, 46
rscadm modem_setup, 48
rscadm resetrsc, 48
rscadm send_event, 48
rscadm set, 15, 47
rscadm show, 47
rscadm utility, 9, 11, 13, 15, 45
rscadm version, 47
rscadm, and user account administration, 49
rsc-config script, 9, 9 to 11

S

- script, sample, 61
- security, 6
- serial port, 4, 8, 13, 32
- session time-out, 6
- sessions, multiple
 - Ethernet port, 12
 - serial port modem, 12
- set command, 25
- setenv command, 14
- shell commands, 17
 - alarm, 22
 - break, 21
 - configuration commands, 25
 - configuration variables, 30
 - alert, 34
 - console, 42
 - Ethernet port, 41
 - serial port, 32
 - console, 21
 - consolehistory (chist), 24
 - consolerestart, 25
 - date, 26
 - environment, 20
 - help, 30
 - loghistory (lhist), 23
 - login, 17
 - logout, 6, 30
 - other commands, 30
 - password, 27
 - poweroff, 22
 - poweron, 22
 - powersupply, 22
 - reset, 21
 - resetrsc, 29
 - server status and control, 20
 - set, 25
 - show, 25
 - shownetwork, 20
- table, 19
- useradd, 27
- userdel, 28
- userpassword, 28
- userperm, 28
- usershow, 28
- version, 30
- view log commands, 22
- xir, 21

- show command, 25
- shownetwork command, 20
- SMTP, 10, 38
- SNMP, 34
- subnet mask, 10
- superuser account, 10

T

- telnet, 4, 8, 17
- temperature, 20, 40
- time
 - differences in event and console logs, 24
 - synchronizing to server's, 47
 - troubleshooting, 55
- time-out, 6, 18, 33
- tip program, 8, 43, 49
- troubleshooting RSC problems, 53
 - cannot connect through Ethernet, 54
 - cannot connect through serial port, 54
 - cannot log in, 53
 - no alerts received, 54
 - not working after alarm card swap replacement, 56
 - passwords unknown, 55
 - some functions work, others do not, 55
 - times don't match, 55
- troubleshooting server problems using RSC, 56

U

- user account, 10, 15, 17, 27
- user permissions, 20, 28
- useradd command, 27
- userdel command, 28
- userpassword command, 28
- userperm command, 28
- usershow command, 28

V

- variables, configuration, 30
 - alarm
 - alarm1_prefix, 40

- alarm2_prefix, 40
- alarm3_prefix, 40
- solaris_watchdog_reboot, 41
- solaris_watchdog_timeout, 40
- alert, 34**
 - customerinfo, 35
 - hostname, 35
 - mail_enabled, 38
 - mailhost, 38
 - mailuser, 38
 - page_baud1, 36
 - page_baud2, 37
 - page_data1, 36
 - page_data2, 37
 - page_enabled, 35
 - page_info1, 35
 - page_info2, 37
 - page_init1, 36
 - page_init2, 37
 - page_parity1, 36
 - page_parity2, 38
 - page_password1, 36
 - page_password2, 37
 - page_stop1, 36
 - page_stop2, 38
 - snmp_enabled, 34
 - snmp_host, 34
- console, 42**
 - escape_char, 42
- Ethernet port, 41**
 - ip_addr, 41
 - ip_gateway, 42
 - ip_mode, 41
 - ip_netmask, 41
 - tpe_link_test, 42
- serial port, 32**
 - ppp_enabled, 34
 - ppp_local_ip_addr, 33
 - ppp_remote_ip_addr, 34
 - serial2_baud, 32
 - serial2_data, 33
 - serial2_hw_handshake, 33
 - serial2_mode, 32
 - serial2_parity, 32
 - serial2_stop, 33
- server**
 - panic_dump, 43
- version command, 30

X

- xir command, 21

