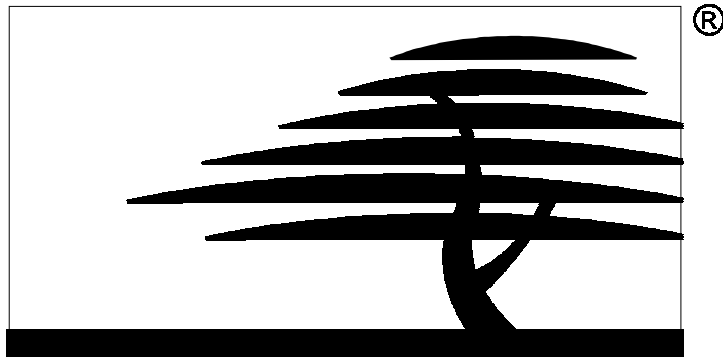




SCO

**SCO OpenServer
Release 5 and
UnixWare® 7
Administration I:
User Services**

Student Guide

Course Release: 2.0

For Software: SCO OpenServer Release 5
and UnixWare® 7

© 1987–1999 The Santa Cruz Operation, Inc. All rights reserved.

This publication is protected under copyright laws and international treaties.

Information in this document is subject to change without notice and does not represent a commitment on the part of The Santa Cruz Operation, Inc.

SCO, The Santa Cruz Operation, the SCO logos, VP/ix, SCO OpenServer, UnixWare, dbXtra, Panner, SCO Global Access, SCO MPX, SCO Doctor, SCO Doctor for Networks, SCO Doctor Lite, SCO Premier Motif, SCO Visual Tcl, SCO VisionFS, SCO ToolWare, SCO TermLite, SCO Vision 97, SCO CIFS Bridge, SCO WebTop, The Internet Way of Computing, and IWoC are trademarks or registered trademarks of The Santa Cruz Operation, Inc. in the USA and other countries. IXI, X.desktop, Deskworks, Wintif, IXI Panorama, Deskterm, and IXI Desktop are trademarks or registered trademarks of IXI Limited, a subsidiary of The Santa Cruz Operation, Inc. Visionware, Devkit.Vision, Esprit, Kodon, Super.Vision, Visionware Super.Vision, Term.Vision, Visionware Direction, the Visionware logo, Visionware SQL-Retriever, and XVision are registered trademarks of Visionware Limited, a subsidiary of The Santa Cruz Operation, Inc. SuperVision, TermVision, X-Visionware, SQL-Retriever, and Vision Builder are trademarks of Visionware Limited. UNIX is a registered trademark of The Open Group in the United States and other countries. Cheyenne and ARCServe are registered trademarks of Cheyenne Software, Inc. Netscape, Netscape Navigator, Netscape Communications Server, Netscape Commerce Server, Netscape Proxy Server, Netscape FastTrack Server, Netscape Enterprise Server, Netscape SuiteSpot, Netscape Catalog Server, Netscape News Server, Netscape Mail Server, and Netscape Navigator Gold are trademarks of Netscape Communications Corporation. NFS was developed by Legent Corporation (formerly Lachman Associates, Inc.) based on LACHMAN SYSTEM V NFS. LACHMAN is a trademark of Legent Corporation. NFS is a trademark of Sun Microsystems, Inc. TCP/IP was developed by Legent Corporation (formerly Lachman Associates, Inc.) based on LACHMAN SYSTEM V STREAMS TCP, a joint development of Lachman Associates and Convergent Technologies. MPX was developed by Corollary, Inc. VP/ix is a product developed and licensed by Phoenix Technologies, Ltd/INTERACTIVE Systems Corporation. XRemote is a registered trademark of Network Computing Devices, Inc. Oracle is a registered trademark of Oracle Corporation, Redwood City, California. Java is a trademark of Sun Microsystems, Inc. All other brand and product names are or may be trademarks of, and are used to identify products or services of, their respective owners.

The SCO software that accompanies this publication is commercial computer software and, together with any related documentation, is subject to the restrictions on US Government use as set forth below. If this procurement is for a DOD agency, the following DFAR Restricted Rights Legend applies:

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the US Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013. Contractor/Manufacturer is The Santa Cruz Operation, Inc., 400 Encinal Street, Santa Cruz, CA 95060.

If this procurement is for a civilian government agency, this FAR Restricted Rights Legend applies:

RESTRICTED RIGHTS LEGEND: This computer software is submitted with restricted rights under Government Contract No. _____ (and Subcontract No. _____, if appropriate). It may not be used, reproduced, or disclosed by the Government except as provided in paragraph (g)(3)(i) of FAR Clause 52.227-14 alt III or as otherwise expressly stated in the contract. Contractor/Manufacturer is The Santa Cruz Operation, Inc., 400 Encinal Street, Santa Cruz, CA 95060.

The copyrighted software that accompanies this publication is licensed to the End User only for use in strict accordance with the End User License Agreement, which should be read carefully before commencing use of the software.



The SCO[®] Authorized Education class you are about to take may form part of the recommended preparation for those seeking the SCO ACE[™] certification.

The SCO ACE certification program uses a series of exams to measure your ability to support and administer SCO installations with the SCO recommended skills that maximize system performance. As a certified SCO ACE graduate, you earn all of the respect that the industry accords to each certified member of the extended SCO support team. When professionals in the computer industry see your SCO ACE certification, they know that you have passed demanding SCO testing requirements. This certification is the foundation for mastering real-world UNIX[®] systems, including those from other vendors.

Important!

When registering for any SCO ACE exam at a Sylvan Prometric[™] testing center, please quote the course release number on the front of the ***Student Guide*** that the exam covers. This number corresponds to the version number of the SCO ACE exam for that course.

Ask your SCO Authorized Instructor, or call SCO for more information about SCO Certification.

Table of Contents

Lesson 1: Course Introduction	1-1
Introduction.....	1-3
Course goal and purpose	1-3
Target audience	1-3
Student Materials	1-4
The Student Guide	1-4
Job aids	1-4
SCO documentation	1-4
Typographical Conventions	1-5
Using scoadmin	1-7
Sample exam items	1-8
 Lesson 2: Using SCO Administration Tools	 2-1
Using SCOadmin	2-5
Accessing SCOadmin	2-5
Starting individual managers	2-7
Using the Character Motif (CHARM)	2-9
Describing the Screen Areas of SCOadmin	2-11
Menus.....	2-11
Control buttons	2-11
Display area	2-11
Status line.....	2-11
Graphical mode.....	2-11
Character mode	2-13
Locating Topics in SCO Documentation	2-15
Overview of printed documentation	2-15
Overview of online documentation.....	2-15
Using SCOhelp	2-16
Searching in SCOhelp.....	2-17
Using the manual pages	2-17
Accessing the SCO support technical articles	2-18
Using SCO's Online Support (SOS) System	2-18
Newsgroups	2-19
Other resource sites.....	2-19
Technical support	2-20
Using the root and System Owner Account.....	2-21
Maintaining a System Log	2-21
Lesson Summary.....	2-22
Relevant Commands and Files	2-23
Answers to Review Questions	2-24

Lesson 3: Administering User Accounts	3-1
Analyzing User Account Requirements	3-4
User ID number	3-4
Login name	3-4
Security profiles and password settings	3-5
Login Shell.....	3-7
Home directory	3-9
Groups.....	3-9
Creating a group.....	3-11
Changing the default login group	3-13
Reviewing permissions	3-15
Default Discretionary Access Control (DAC)	3-17
Login environment.....	3-19
System environment files.....	3-20
Creating User Accounts	3-21
Creating a default account	3-23
Creating an account using modified defaults.....	3-25
Creating user accounts with useradd	3-27
Deleting Users.....	3-29
Managing user files.....	3-29
Deleting users	3-30
Modifying Accounts	3-33
Modifying individual account information	3-33
Using usermod	3-34
Modifying an individual account environment.....	3-35
Modifying the umask of a single account	3-35
Modifying user passwords	3-35
Controlling password expiration.....	3-36
Controlling password options with default files	3-37
Managing User Groups	3-38
Modifying group membership	3-38
Deleting a group.....	3-38
Setting Login Attempt Restrictions	3-39
SCO OpenServer login restrictions for user accounts	3-41
UnixWare 7 login restrictions for user accounts.....	3-41
Locking and unlocking a user account	3-43
Using Account Activity Reports.....	3-44
Reporting password status	3-44
Reporting login activity	3-45
Troubleshooting User Accounts.....	3-46
Fixing forgotten or compromised passwords.....	3-46
Fixing an incorrect terminal type.....	3-46
Lesson Summary.....	3-47
Lab: Creating Groups.....	3-49
Overview.....	3-49

Procedure	3-49
Lab: Creating New Users.....	3-50
Overview.....	3-50
Procedure	3-51
Relevant Commands and Files	3-52
Answers to Review Questions	3-54
Lesson 4: Managing Processes	4-1

Describing a Process	4-4
Definitions	4-4
The process table	4-4
Viewing current processes	4-5
Process genealogy	4-7
Foreground and background processes	4-7
Daemon processes.....	4-7
Defunct processes	4-8
Terminating Processes	4-9
The kill command.....	4-9
Using the kill command.....	4-10
Managing Processes with the Process Manager	4-11
Viewing processes on a system.....	4-11
Deleting processes	4-14
Scheduling Tasks	4-15
Setting the system date and time.....	4-15
Specifying jobs and scheduled times	4-16
Listing cron jobs	4-16
Adding, modifying, and removing cron jobs	4-17
Removing all cron jobs for a user	4-17
Authorizing users to schedule tasks.....	4-17
UnixWare 7: Using the Task Scheduler	4-18
Modifying System Tasks.....	4-21
Lesson Summary.....	4-22
Lab: Scheduling Processes.....	4-23
Overview.....	4-23
Procedure	4-23
Relevant Commands and Files	4-24
Answers to Exercises and Review Questions	4-25
Lesson 5: Working with Filesystems	5-1

Describing Filesystem Basics	5-5
SCO OpenServer Filesystem Types.....	5-6
UnixWare 7 Filesystem Types	5-7
Advanced filesystem features	5-9
Mounting and Unmounting a Filesystem.....	5-11
Describing filesystem information.....	5-11

Filesystem creation	5-12
Adding the mount configuration	5-13
Mounting a filesystem	5-15
Why mounting can fail	5-15
Unmounting a filesystem	5-16
Repairing a Non-root Filesystem	5-17
Causes of filesystem corruption	5-17
The <i>lost+found</i> directory	5-17
Repairing a non-root filesystem manually	5-19
Monitoring Filesystems	5-21
Determining filesystem free space	5-21
Determining the size of a directory	5-23
Determining disk usage by user	5-25
Locating and Managing User Files	5-26
Finding large files	5-26
Finding files by owner	5-26
Finding and removing <i>core</i> files	5-27
Managing System Log Files and Temporary Directories	5-28
Clearing key system logs	5-28
Clearing key temporary directories	5-29
Lesson Summary	5-30
Lab: Working with a Floppy Disk Filesystem	5-31
Overview	5-31
Procedure	5-32
Relevant Commands and Files	5-33
Answers to Exercises and Review Questions	5-34

Lesson 6: Working with Disks and Tapes

6-1

Working with Floppy Disks	6-5
Identifying floppy device names	6-5
Formatting a floppy disk	6-5
UnixWare 7 <i>getdev</i> and <i>devattr</i> commands	6-7
SCO OpenServer <i>dtype</i> command	6-7
Working with Tape Drives	6-8
Identifying tape device names	6-8
Working with tapes	6-9
Transferring Data to Tape or Disk	6-10
Absolute versus relative pathnames	6-10
Transferring symbolically linked files	6-11
The tar command	6-13
The cpio command	6-19
Accessing DOS data	6-22
Formatting a DOS floppy	6-22
Using DOS utilities	6-22
Mounting a DOS filesystem	6-23

Lesson Summary	6-24
Lab: Creating an Archive and Extracting Selected Files	6-25
Overview	6-25
Procedure	6-26
Relevant Commands and Files	6-29
Answers to Exercises and Review Questions	6-30
 Lesson 7: Backup Administration	 7-1
Organizing and Maintaining Filesystem Backups	7-4
Deciding when to back up filesystems	7-4
Storing backup media	7-4
Maintaining backup logs	7-5
Using different backup levels	7-5
Describing backup products	7-6
SCO OpenServer SCOadmin Backup Manager	7-6
SCO ARCserve/Open Lite	7-7
SCO ARCserve/Open	7-7
ARCserveIT	7-9
Third party backup products	7-10
Lesson Summary	7-11
Relevant Commands and Files	7-12
Answers to Review Questions	7-13
 Lesson 8: Managing Printers	 8-1
Managing the Print Service	8-4
Describing the print service	8-4
Stopping the print service	8-5
Starting the print service	8-7
Accepting and Rejecting Print Jobs	8-9
Rejecting print jobs	8-9
Accepting print jobs	8-10
Disabling and Enabling Printers	8-11
Disabling a printer	8-11
Enabling a printer	8-13
Managing Print Jobs	8-15
Viewing print jobs	8-16
Deleting print jobs	8-17
Holding print jobs	8-18
Moving print jobs	8-20
Viewing and Changing Printer Settings	8-23
Setting the default printer	8-23
Setting a default printer for a user	8-23
Controlling access to a printer	8-25
Preventing banner pages	8-27
Troubleshooting Printer Problems	8-28

Lesson Summary	8-29
Lab: Managing Printers	8-30
Overview	8-30
Procedure	8-31
Relevant Commands and Files	8-32
Answers to Review Questions	8-33
 Lesson 9: Stopping and Starting the System	 9-1
Stopping SCO OpenServer and UnixWare 7 systems	9-5
Normal shut down	9-6
Other shut down circumstances	9-8
Starting the system	9-9
Autoboot to multiuser mode	9-9
Other boot options	9-10
Switching Between System States	9-13
Lesson Summary	9-14
Relevant Commands and Files	9-15
Answers to Exercises and Review Questions	9-16
 APPENDIX A: ARCserveIT Administration	 A-1
Describing ARCserveIT Products	A-4
System requirements	A-5
Licensing ARCserveIT	A-5
Installing ARCserveIT	A-5
Configuring ARCserveIT	A-5
Starting ARCserveIT	A-6
Stopping ARCserveIT services	A-7
Accessing ARCserveIT	A-9
Exiting ARCserveIT	A-10
Setting up user profiles	A-10
Creating a user	A-10
Accessing Tape Device(s)	A-13
Selecting devices and device views	A-13
Maintaining Media	A-15
Ejecting a tape	A-16
Using compression	A-17
Backing up files and filesystems	A-19
Expanding and object	A-19
Viewing filesystems and files	A-19
Performing a system backup	A-23
Backing up filesystems	A-24
Backing up selected files	A-24
Monitoring the status of a backup	A-27
Backup filter options	A-27

Backup method	A-27
Backup schedule	A-28
Placing backup jobs on hold	A-29
Saving backup jobs as scripts	A-29
Verifying the backup and checking logs	A-33
Scanning media.....	A-33
Viewing Reports and Logs.....	A-33
Restoring files and filesystems from backups	A-37
Browsing media	A-37
Performing a Basic Restore Job.....	A-38
Restore filter options.....	A-39
Lesson Summary.....	A-40
Relevant Commands and Files	A-41
Answers to Review Questions	A-42

1

Course Introduction

Lesson 1

Course Introduction

1.1 Introduction

The *SCO OpenServer Release 5 and UnixWare 7 Administration I: User Services* course is designed for individuals who need to perform routine system administration tasks on a daily basis. The target audience includes support technicians, developers, engineers, system administrators, and other technical users who require the ability to perform standard system administration tasks.

1.1.1 Course goal and purpose

At the conclusion of this course, you should be able to perform the tasks and duties associated with the routine operation of SCO OpenServer and UnixWare 7 systems. These tasks include technical procedures such as supporting user accounts and user processes, maintaining filesystems and backing up their data, managing printers, and performing system startups and shutdowns.

1.1.2 Target audience

This class is aimed at system operators, developers, engineers, support technicians, system administrators and other technical users who require an understanding of front line system administration. Students are likely to have a wide variety of backgrounds in XENIX® or UNIX®, but must have prerequisite skills equal to or exceeding those provided in the *Introduction to SCO UNIX Systems* course.

1.2 Student Materials

The contents of the box containing your student materials are yours to keep and write notes in. Your instructor may have other reference materials in the classroom for your use while attending this class. Please do not take or write in these materials as they will be needed again for later classes.

1.2.1 The *Student Guide*

The *Student Guide* contains:

- Outline of presentation
- Copies of all transparencies

1.2.2 Job aids

Job aids are designed to assist you in performing on-the-job administration and as references to other helpful materials. This course includes the **vi Quick Reference Guide**, a card that provides a list of the most commonly used **vi** commands and their functions.

1.2.3 SCO documentation

This course references the online and printed documentation included with SCO OpenServer and UnixWare 7.

1.3 Typographical Conventions

This publication presents commands, filenames, keystrokes, and other special elements as shown here:

lp or lp(1)	Commands, device drivers, programs, and utilities (names, icons, or windows); the letter in parentheses indicates the reference manual section in which the command, driver, program, or utility is documented
<i>/new/client.list</i>	Files, directories, and desktops (names, icons, or windows)
root	System, network, or user names
<i>arg</i> or <i>arg</i>	Placeholders (replace with appropriate name or value)
[<i>arg</i>]	Surrounds an <i>option</i> or <i>cmdarg</i> that is not mandatory
<i>arg</i> ...	Indicates multiple occurrences of the <i>option</i> or <i>cmdarg</i>
\$	Command line prompt for commands executable from the Bourne shell by any user
#	Command line prompt for commands only executable by the super user from the root account
	Keyboard keys
 	Press  and  keys simultaneously
  	Press all three keys,  ,  , and  , simultaneously
 d	Press  key. With  still depressed, also press  .
 0240	Press  key. With  still depressed, also press and release in sequence  ,  ,  , and  .
 Q + i	Press  key. With  still depressed, also press  . Release both keys and press  .
 Q + I	Press  key. With  still depressed, also press  . Release both keys and press   .
Exit program?	System output (prompts, messages)

tar tv	User input, such as on a command line
“Description”	Field names or column headings (on screen or in database)
Cancel	Button names
Edit	Menu names
Copy	Menu items
File ⇨ Find ⇨ Text	Sequences of menus and menu items
open or open(2)	Library routines, system calls, kernel functions, C keywords; the letter in parentheses indicates the reference manual section in which the file is documented
\$HOME	Environment or shell variables
SIGHUP	Named constants, signals, and kernel parameters
“adm3a”	Data values
<i>employees</i>	Database names and tables
buf	C program structures
b_b.errno	Structure members

1.4 Using **scoadmin**

Many procedures in this course indicate to “start the **Account Manager**” or “run **scoadmin**” when invoking administration utilities.

SCO OpenServer Release 5 and UnixWare 7 include both graphical and character-based versions of most administration utilities. When started from the Desktop, the graphical versions automatically run. When started from a character device (such as a terminal), the character versions run.

In this course, the Desktop is used to start the graphical versions of administration utilities. There are several ways to run **scoadmin** utilities.

For example, to run the **Account Manager**, a tool that allows you to create and manage users:

- From the **SCOadmin** icon, select ⇒ **Account Manager**.
- From a terminal window, enter **scoadmin Account Manager**.
- From the command line, enter **scoadmin acc**.

When you are directed to “run the **Account Manager**” (or “start the **Account Manager**” later in this guide, you can use any of these methods.

Be sure you are logged in as **root**.

1.5 Sample exam items

The following questions are representative exam items on the SCO ACE certification exam for this course. Being able to correctly answer these questions does not guarantee that you will be able to pass the actual exam.

1. Which login shell is assigned to root by default?
 - A. The Bourne shell
 - B. The C shell
 - C. The Korn shell
 - D. The SCO shell
2. What types of processes perform system tasks on a periodic basis and sleep between those tasks?
 - A. Child processes
 - B. Daemon processes
 - C. Foreground processes
 - D. Zombie processes
3. What filesystem feature can reduce the system downtime incurred when cleaning a large filesystem?
 - A. Access control lists
 - B. Data compression
 - C. Intent logging
 - D. Extent based allocation

4. Which system log file contains account records?
- A. `/etc/wtmp`
 - B. `/etc/syslogd`
 - C. `/etc/motd`
 - D. `/usr/adm/syslog`
5. Which device file represents a floppy device?
- A. `/dev/xct0`
 - B. `/dev/vt02`
 - C. `/dev/rfd0`
 - D. `/dev/c1s0`
6. Which command can you use to send a job to a printer?
- A. **accept**
 - B. **enable**
 - C. **lp**
 - D. **lpsched**

Answers to sample exam items

1. A
2. B
3. C
4. A
5. C
6. C

2

Using SCO Administration Tools

AV 2-1

Using SCO Administration Tools

- Use **scoadmin**(1M)
- Locate topics in SCO documentation
- Identify other sources of information
- Describe the importance of the superuser and owner accounts
- Maintain a system log

sa1g-0201© The Santa Cruz Operation, Inc.

Lesson 2

Using SCO Administration Tools

Introduction

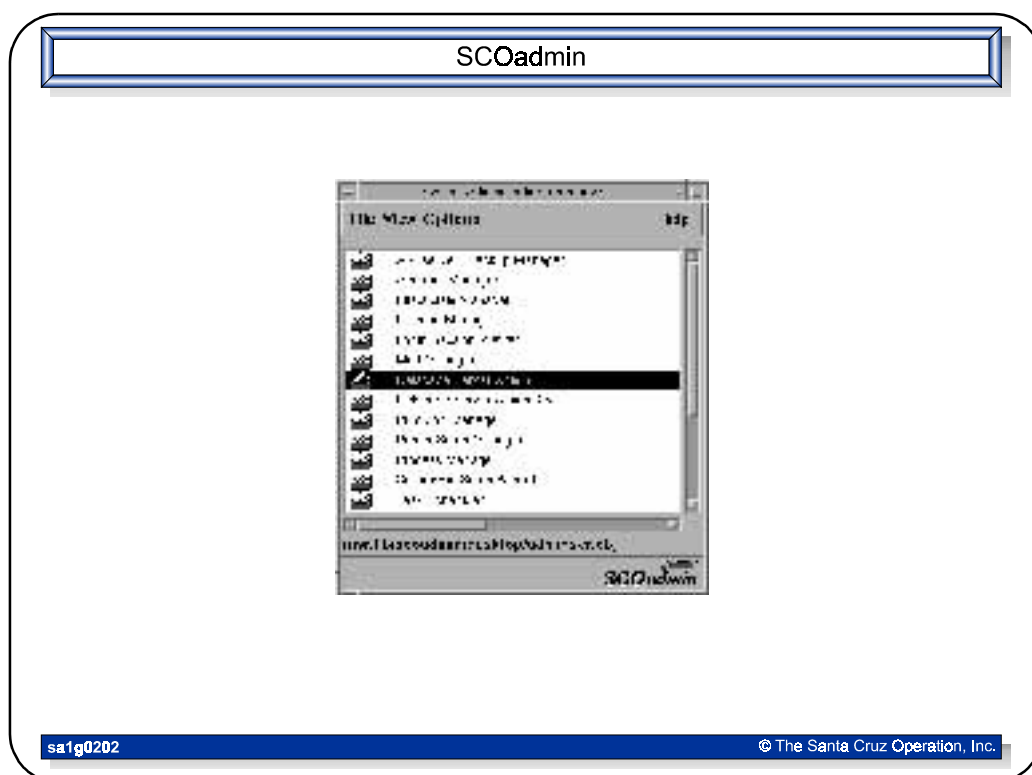
A system administrator is responsible for performing a wide variety of tasks. This course focuses on the most common tasks involved in the routine administration of SCO OpenServer Release 5 and UnixWare 7 systems.

Objectives

This lesson introduces a system administrator's responsibilities and the available tools and resources. By the end of this lesson you will be able to:

- Use **scoadmin**
- Locate topics in SCO documentation
- Identify other sources of information
- Describe the importance of the **root** account
- Maintain a system log

AV 2-2



2.1 Using SCOadmin

Most of the system administration tasks described in this course can be accomplished using **scoadmin**, SCO's system administration and configuration tool. This tool consists of a series of managers that support a graphical interface. Many of the **SCOadmin** managers are available in a character interface, but the methods of navigation differ.

2.1.1 Accessing SCOadmin

You can use either one of these methods to access the **SCOadmin** managers from your graphical desktop:

- Step 1. Log in as **root** using the graphical login.
- Step 2. **SCO OpenServer:** Double-click the **System Administration** folder on the desktop

UnixWare 7: Click on the arrow above the **SCO logo** icon from the **Main Panel** of your desktop. Click on **SCOadmin**.
- Step 3. Select a manager. Double-click on the manager name.
- Step 4. Exit the manager. Select **Host** ⇒ **Exit**.
- Step 5. Exit **SCOadmin**. Select **File** ⇒ **Exit**.

Exercise: Accessing a SCOadmin manager.

Access the **SCOadmin Account Manager** by following the steps under Section 2.1.1, "Accessing SCOadmin," on page 2-5.

From the command line while in the desktop, use the second method:

- Step 1. Open a graphical terminal window.
- Step 2. Enter **scoadmin&**. The **&** allows you to continue to use your terminal window for other commands.
- Step 3. Select a manager.

2.1.2 Starting individual managers

You can also start individual managers by using the **scoadmin** command with the application name. For example, to start the **Account Manager** from a desktop **Terminal** window, enter:

```
scoadmin account manager&
```

List the available managers, enter:

```
scoadmin -t
```

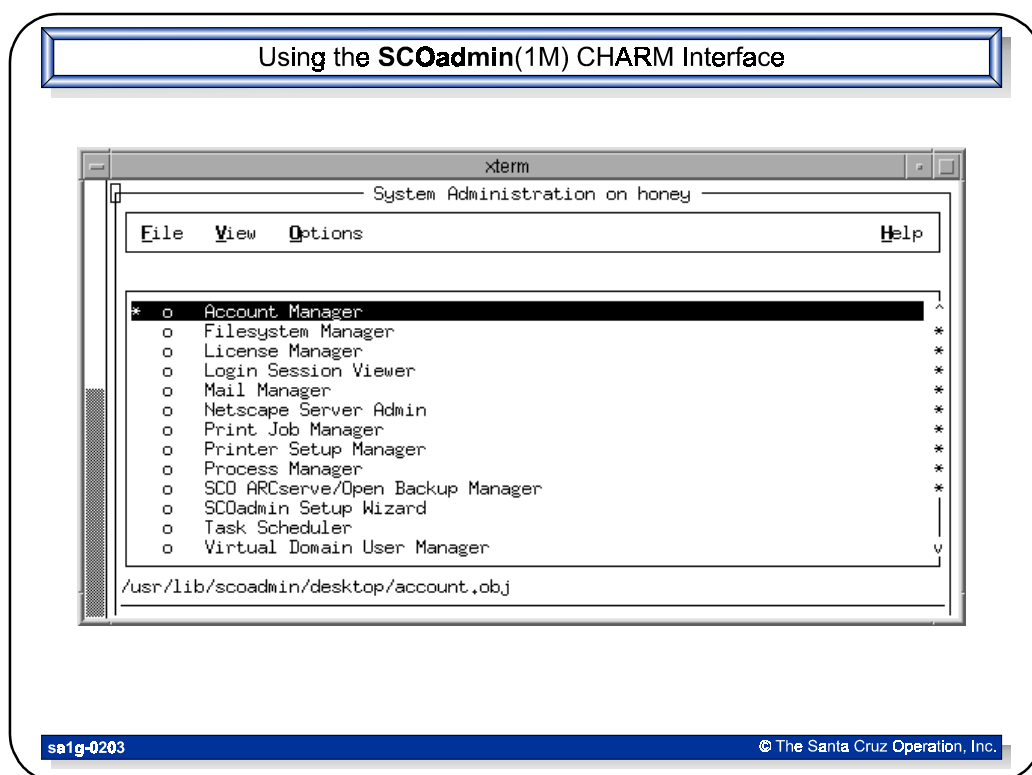
Note that you do not have to enter the full manager name. A minimum number of characters is accepted. For example, to run the **Account Manager**, enter:

```
scoadmin ac&
```

Exercise: Starting an individual manager

Start the **SCOadmin Process Manager** from a **Terminal** window. Refer to the second method under Section 2.1.2, “Starting individual managers,” on page 2-7.

AV 2-3



2.1.3 Using the Character Motif (CHARM) ---

You can also access **SCOadmin** using the Character Motif (CHARM) interface by switching from your desktop to a console screen. This is also useful if you are accessing **SCOadmin** from a terminal. Follow these steps to access the **SCOadmin** managers using the CHARM interface:

Step 1. Switch from the desktop to a character screen. For example, press:

Step 2. Log in as **root**. Enter the password.

Step 3. Enter **scoadmin**. Select the manager.

Step 4. Press  ⇒ **Host** ⇒ **Exit** to exit the manager.

Step 5. Press  ⇒ **File** ⇒ **Exit** to exit the **SCOadmin** manager.

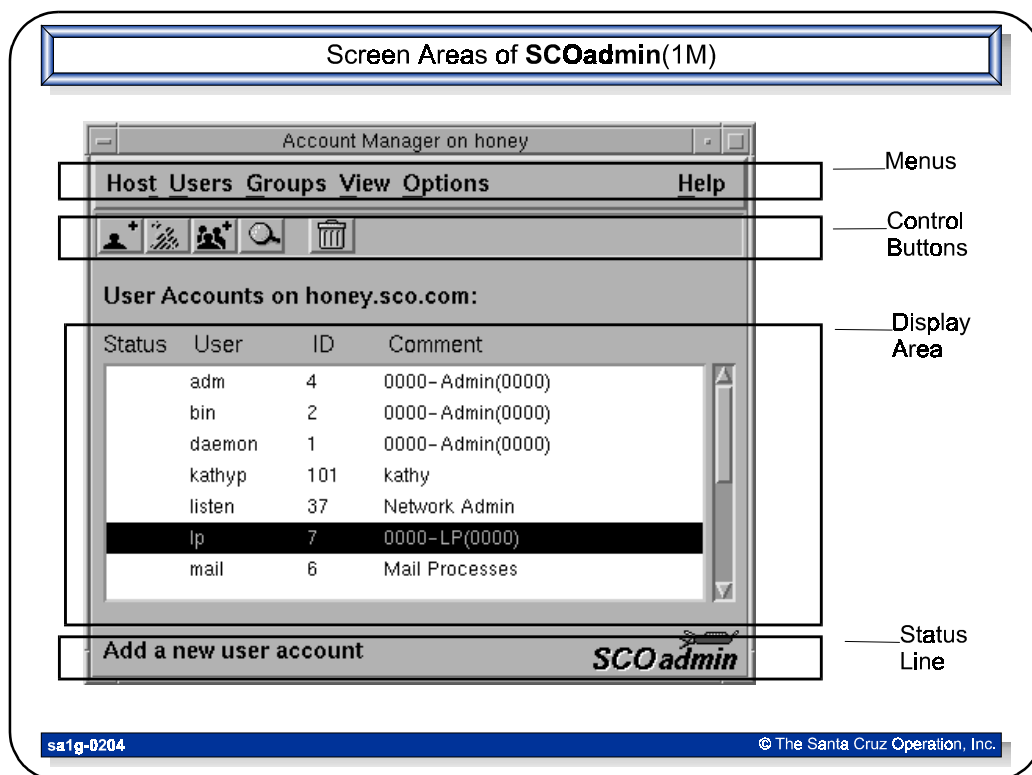
Step 6. **SCO OpenServer:** Press    to return to the desktop.

UnixWare 7: Press    to return to the desktop.

Exercise: Starting the SCOadmin character interface

Start and stop the **SCOadmin** character interface. Follow the steps under Section 2.1.3, “Using the Character Motif (CHARM),” on page 2-9.

AV 2-4



2.2 Describing the Screen Areas of SCOadmin

In each **SCOadmin** manager, there are four distinct areas of the window.

2.2.1 Menus

Located at the top of the window, the menu provides access to the various administration tasks supported by that manager. For example, the **Host** menu lets you select a remote system to control and the **Help** menu allows you to access context-sensitive online help.

2.2.2 Control buttons

Control buttons appear in the data area, typically at the top. They provide shortcuts to the menu options or allow you to configure additional options for a particular manager.

2.2.3 Display area

This section contains the system elements that each manager supports. For example, the display area of the **Account Manager** lists user accounts, and the display area of the **Filesystem Manager** lists mounted filesystems.

2.2.4 Status line

At the bottom of the screen is a status line. The default behavior of the manager is to list a brief explanation of the currently selected menu option or button. If you find the default behavior distracting, select **Options** ⇒ **Point Help** to disable the display.

2.2.5 Graphical mode

These are general guidelines for using the administration managers in graphical mode:

- Use the mouse to switch between the various screen areas.
- Select a menu by clicking on it. A drop down menu appears. Execute an option by clicking on it.



- Switch between options within a particular screen area by using the arrow keys.
- Activate buttons by clicking on them.
- Select a particular element in the display area by clicking on it.

2.2.6 Character mode

The character mode version of **scoadmin** is based upon the Character Motif (CHARM) interface. It executes some of the same managers as the graphical version, so the layout of the data and menu options are similar. Instead of using a mouse, you can use these keys to navigate through the interface:

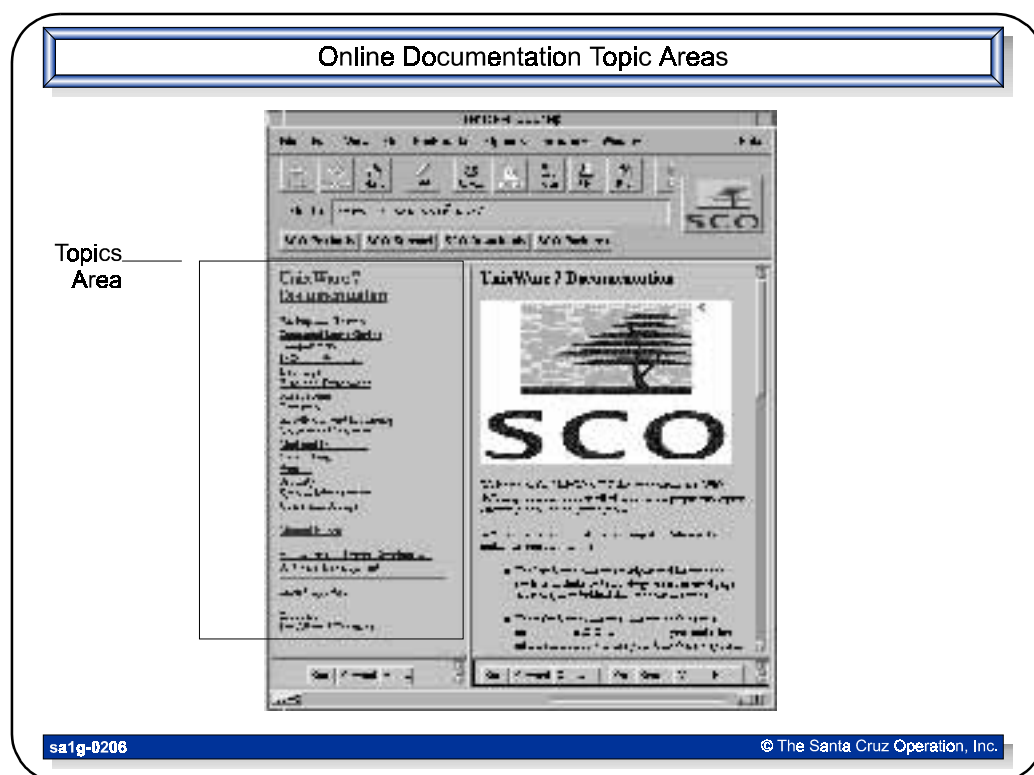
- Use  to switch between the various screen areas. You can also press  to go to the menu.
- Use  on a menu to open the drop-down menu. Use  and  to select a menu option. Press  to execute a menu option.
- Activate a button by selecting it, then pressing  or .
- Select a particular element in the display area by using the arrow keys to highlight it, then press  or .
- Use the  key to cancel a drop down menu.

While in any character based **SCOadmin** manager, use the  to list the “Charm Navigation and Select Keys.” help screen.

Exercise: Using SCOadmin to change the system time.

Follow the steps under Section 2.1.1, “Accessing SCOadmin,” on page 2-5 and access the **System** ⇒ **System Time Manager**. If necessary, change the time.

AV 2-6



2.3 Locating Topics in SCO Documentation

SCO provides comprehensive documentation covering the use of its systems. This documentation is packaged online as part of the product. Some of the documentation is available in printed form. This section provides an overview of this documentation so that you can easily locate topics to find the answers to your questions.

2.3.1 Overview of printed documentation

The printed UnixWare 7 documentation includes:

- *UnixWare 7 Installation Guide*
- *UnixWare 7 Runtime Release Notes*
- *UnixWare 7 System Handbook*

More extensive printed documentation is available for SCO OpenServer. Among the more popular printed SCO OpenServer books are the following:

- *SCO OpenServer Handbook*
- *SCO OpenServer Operating System Administration Guide*
- *SCO OpenServer Operating System User's Guide*
- *SCO OpenServer Networking Guide*

The topics in the printed documentation are also included with the online documentation.

2.3.2 Overview of online documentation

SCO systems provide extensive online help to both graphical and character-based users.

- **SCOhelp** provides graphical online help
- The **man** command provides character-based online help

2.3.3 Using SCOhelp

Use **SCOhelp** to access the online documentation set. **SCOhelp** is a browser-based help system complete with hypertext links and a search engine.

From the desktop, you can start **SCOhelp** in the following ways:

1. Open a terminal window from the Desktop. Enter:

scohelp&

Note: scohelp cannot be accessed from a character terminal. Use the online **man** pages.

2. **SCO OpenServer:** Double-click on the **SCOhelp** icon. This is the icon that appears as a life preserver on the desktop.

UnixWare 7: Click on the **SCOhelp** icon. This icon is available from the CDE front panel. It is the icon with a question mark in the middle.

The enhanced features of **SCOhelp** for UnixWare 7 include:

- A new frames interface
- New organization of topics (or subject areas)
- Enhanced print functionality
- Buttons for navigation
- The ability to search for both words and phrases using the Verity Search Engine

2.3.4 Searching in SCOhelp

SCOhelp includes a search engine that may assist you in locating specific topics and information. SCO OpenServer allows you to search the entire online support library, but for only a single keyword. UnixWare 7 includes a more powerful search engine from Verity.

Exercise: Using SCOhelp

From the desktop, run **SCOhelp**.

Search for the word *printer*. Which topic provides the information to install a printer?

2.3.5 Using the manual pages

You can access the manual pages either through **SCOhelp** or from the command line:

Exercise: Locating the shutdown manual page

Use **man** from the command line to locate the manual page for the **shutdown** command.

Defaults for the **man** command are stored in */etc/default/man*.

2.3.6 Accessing the SCO support technical articles

SCO's Technical Support group keeps technical articles (TAs) describing how to fix various problems the group has encountered. You can access the TA database via a Web browser at:

<http://www.sco.com/ta/>

If you do not have Web access, you may also purchase this collection of articles for local reference. These articles are grouped by major product functions, and you can conduct a key word search to locate help on specific problems.

2.3.7 Using SCO's Online Support (SOS) System

SOS is a menu driven BBS program that provides a character-based, front-end for ease of use for those who do not have a Web browser or Internet access.

SCO's Online Support (SOS) system contains similar information to that available on WWW.

The required software or hardware to access SOS, include:

- Terminal and modem
- Computer, modem, and third-party communications package
- **cu** command (for SCO XENIX® and SCO UNIX® systems)

Note: To download supplemental patches, your third-party communications package must support XMODEM, YMODEM, ZMODEM, or kermit transfer protocols.

2.3.8 Newsgroups

If you have access to the Internet, you can find many technical newsgroups where users share tips and information. Communicate with a wide range of users and find someone who has encountered, or solved, a problem similar to yours.

SCO-specific newsgroups include:

- comp.unix.sco.announce
- comp.unix.sco.misc
- comp.unix.sco.programmer
- comp.unix.unixware.announce
- comp.unix.sco.announce
- comp.unix.unixware.misc

For information about these newsgroups check the technical support area on the SCO WWW Page:

<http://www.sco.com/company/netnews.html>

2.3.9 Other resource sites

In addition to newsgroups, SCO also sponsors:

SCO Home Page	Provides a graphical interface to browse SCO's technical scripts and download patches. Access <i>http://www.sco.com/</i> .
SCO FTP sites	Download patches using File Transfer Protocol (FTP). If you do not have a Web browser, you can download patches from these sites. Access <i>ftp.sco.com</i> and <i>sosco.sco.com</i> .
SCO Documentation site	Access online documentation via the web at <i>http://doc.sco.com/</i>
CompuServe	SCOFORUM

2.3.10 Technical support

You can also receive one-on-one help through the Technical Support services offered by SCO or its authorized partners. These professionals can help you solve problems you might encounter while administering the system.

2.4 Using the root and System Owner Account

Every UNIX system has an account for system administration. It is known as the **root** account and:

- Can read any file
- Can write or delete any file
- Can execute and terminate any user program

Because there are no restrictions to this account, you can potentially delete major portions of the system. Use the **root** account only when necessary and be careful when you are using it. Use your regular login account when you are not performing system duties.

2.5 Maintaining a System Log

When troubleshooting a system, one of the most valuable tools available is a current system log. This book should be kept next to the system and should contain the following types of information:

- All system modifications, including hardware and software configuration information
- All system problems encountered, including detailed summaries of the causes and solutions
- Statistics describing system use, performance, and growth

2.6 Lesson Summary

There are many resources available to you as an administrator. Administration tools and documentation come with the system to support your job. For the most current information and more extensive support, you can turn to technical libraries, other users, or SCO Technical Support.

Review questions

1. List the steps to start the **SCOadmin Process Manager** from a **Terminal** window.
2. Which option to the **scoadmin** command would you use to display the available **SCOadmin** managers?
3. Which menu selection in each of the **scoadmin** managers allows you to turn off the Status Line?
4. List two methods to access SCOhelp.
5. Should the system log be kept online? Why or why not?

2.7 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

man	Displays manual pages
scoadmin	Invokes SCOadmin managers
scohelp	Displays documents, manual pages, and context sensitive help

Relevant files

<i>/etc/default/man</i>	Default configuration file for the use of man
-------------------------	--

2.8 Answers to Review Questions

This section provides the answers to review questions presented in this lesson.

Review questions

1. The steps to start the **SCOadmin Process Manager** from a terminal window are: Login to a terminal window as **root**. Enter: **scoadmin pro**.
2. The option to list the available **SCOadmin** managers is **scoadmin -t**.
3. The menu selection in each of the **scoadmin** managers that allows you to turn off the Status Line is the **Options** ⇒ **Point Help** menu selection.
4. Two methods to access **SCOhelp** are: click on the **SCOhelp** icon, or enter **scohelp** from a terminal window.
5. The system log should not be kept online. If the system crashes to the point where you cannot boot, you need to be able to access a hard copy log.

3

Administering User Accounts

AV 3-1

A slide titled "Administering User Accounts" with a blue header bar. The slide contains a bulleted list of tasks. At the bottom, there is a blue footer bar with the text "sa1g-0301" on the left and "© The Santa Cruz Operation, Inc." on the right.

Administering User Accounts

- Analyze user account requirements
- Create, modify, and delete user accounts
- Manage user groups
- Set, change, and remove login and password restrictions
- Create and use account activity reports
- Troubleshoot user accounts

sa1g-0301 © The Santa Cruz Operation, Inc.

Lesson 3

Administering User Accounts

Introduction

The main task of any system administrator is to support the use of computer resources by users. Before a user can use an SCO OpenServer Release 5 or UnixWare 7 system, the user must have an account on the system. The system administrator's job includes creating accounts for users and maintaining these accounts by changing user environments, passwords, and group membership, as necessary.

Objectives

This lesson demonstrates the creation and management of user accounts and groups. By the end of this lesson, you will be able to:

- Analyze user account requirements
- Create, modify, and delete user accounts
- Manage user groups
- Set, change, and remove login and password restrictions
- Create and use account activity reports
- Troubleshoot user accounts

3.1 Analyzing User Account Requirements

Before a user can log into and use the system, you must create an account for that user. A user account is more than a login ID; it sets up the user's basic working environment and access privileges for the system.

A little planning can simplify account management. Most users probably share similar requirements with regards to access privileges. Others might have special needs. Setting up system-wide account defaults saves time when creating and maintaining accounts. This section addresses issues to consider before creating user accounts.

3.1.1 User ID number

The User ID number, or UID, is a unique number that identifies each user. It can be assigned by the system or by the administrator. The suggested minimum UID for individual accounts is 200 for SCO OpenServer and 100 for UnixWare 7. The suggested maximum UID is 60000.

Use the same UID to identify a specific person on all systems to which the person has access.

The **id** command displays the user name and user ID and the group name and group ID for a user. It also displays any other group names and IDs to which a user belongs.

3.1.2 Login name

The login name is the account name entered at the **login:** prompt. It must be unique for each user on the system. The maximum login length is 8 characters for SCO OpenServer and 254 characters for UnixWare 7. For compatibility with all system utilities and other UNIX platforms, the suggested length is between 3-8 characters.

The login name:

- Should not contain upper case letters (Cannot for SCO OpenServer)
- Cannot contain a colon character (:)
- Cannot start with a number

As with the User ID, use the same login to identify a specific person on all systems.

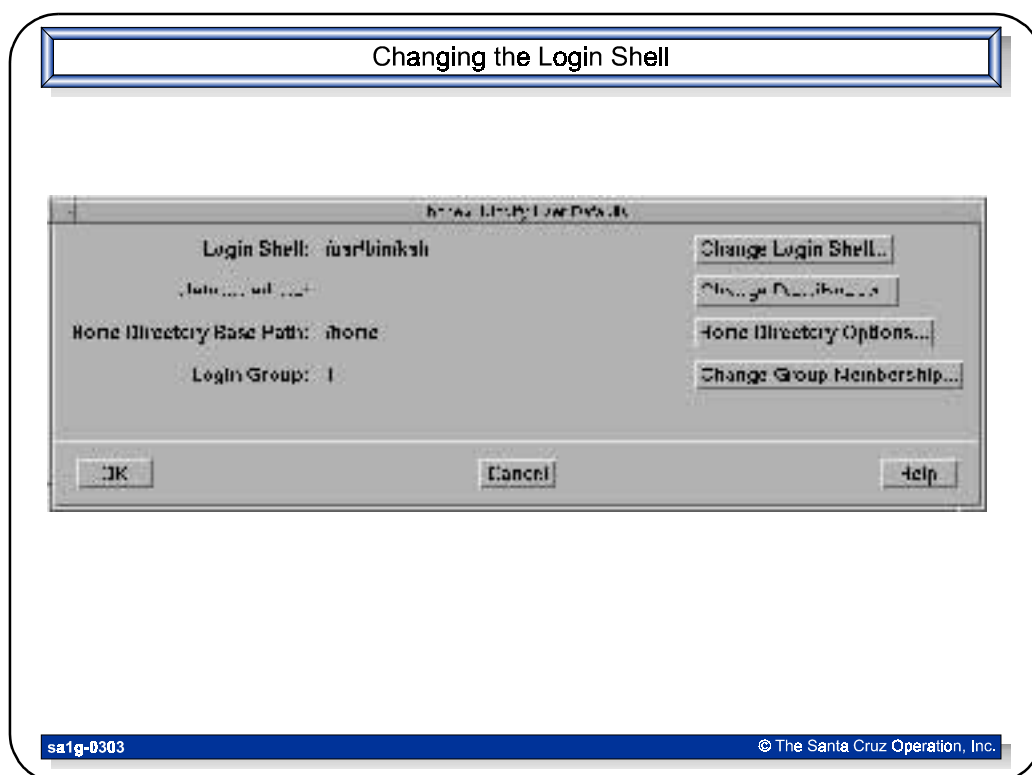
3.1.3 Security profiles and password settings

Before a user can log into the system, the user account should have an assigned password. Password restrictions are based upon the security profile of the system, but you can also modify password expiration parameters for an individual user.

The security profile is chosen during the installation of your system. The four security profiles are:

Low	Passwords are not required.
Traditional	SCO OpenServer: Passwords are not required. The default minimum password length is three (3) characters. UnixWare 7: Passwords are required (however, a carriage return may be considered a valid password). The default minimum password length is three (3) characters.
Improved	Passwords are required. The minimum length is six (6) characters.
High	Passwords are required. The minimum length is eight (8) characters.

AV 3-2



3.1.4 Login Shell

The *login shell* is the command shell the user sees immediately after login. The default shell for SCO OpenServer is the Bourne Shell (sh) and for UnixWare 7 it is the Korn shell (ksh), however the default can be set to one of many available shells, including:

- Bourne or Restricted Bourne Shell
- Korn or Restricted Korn Shell
- C Shell

Use these steps to change the login shell for account creation:

Step 1. Start **SCOadmin** ⇒ **Account Manager**.

Step 2. Select **Options** ⇒ **User Defaults**.

Note the login shell, home directory, and login group.

Step 3. Select **Change Login Shell**, select the new login shell, then click on **OK** to return to the main defaults window.

Step 4. Click on **OK** twice to exit the **User Defaults** window and proceed with the change.

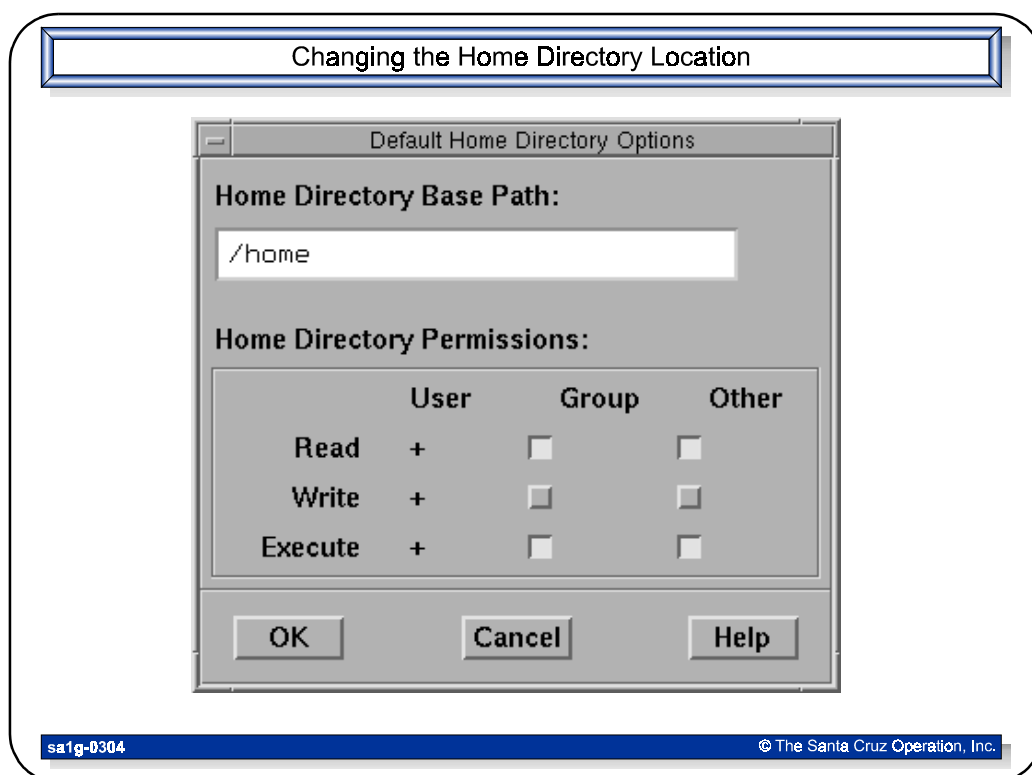
Step 5. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

Any defaults changed affect only the creation of new user accounts. Existing accounts are not affected.

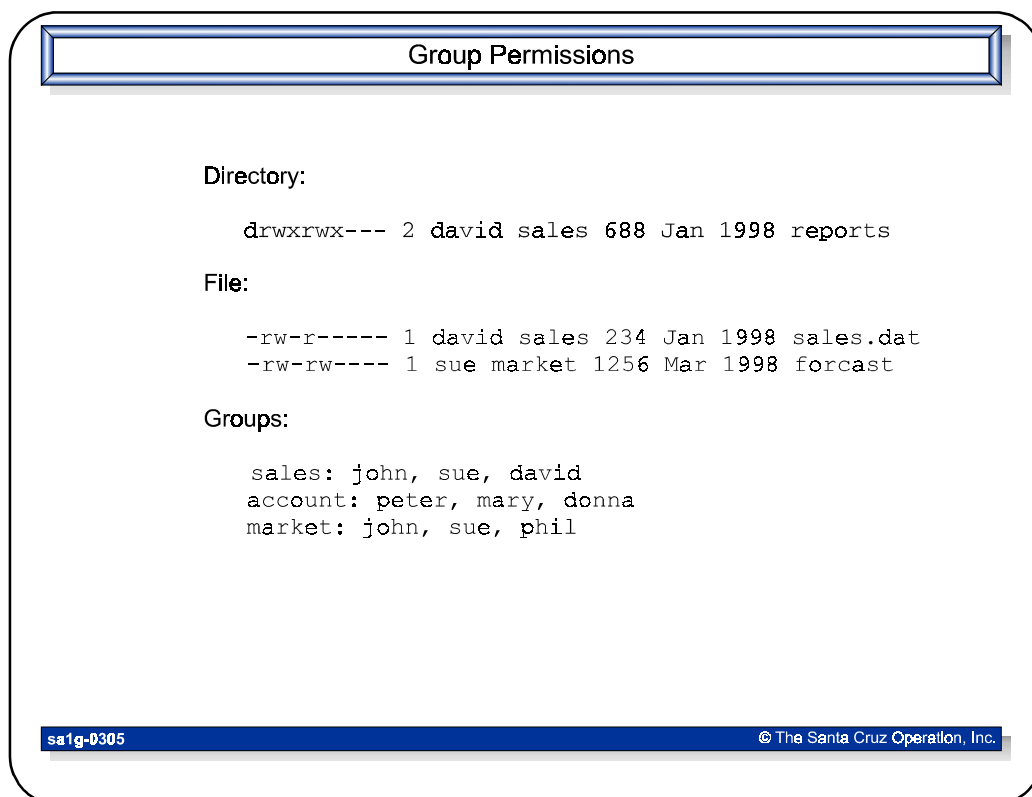
Exercise: Setting the default login shell

On your classroom system, change the default login shell to the C shell. If necessary, refer to the steps under Section 3.1.4, “Login Shell,” on page 3-7.

AV 3-3



AV 3-4



3.1.5 Home directory

The *home directory* is the user's "work space" on the system and the directory where most of the user's files are stored. By default, for SCO OpenServer this directory is */usr/login_name*, and for UnixWare 7 it is */home/login_name*.

In some environments, groups of users share a single home directory. In that case, the directory name should reflect the group, not an individual user.

To change the default home directory location from the **SCOadmin Account Manager**, select **Options** ⇒ **Users Defaults** ⇒ **Home Directory Options**.

3.1.6 Groups

Groups allow sets of users to share files. The group permissions on a file or directory determine whether members of a group can access them. This allows work groups to share files and still restrict access to sensitive data.

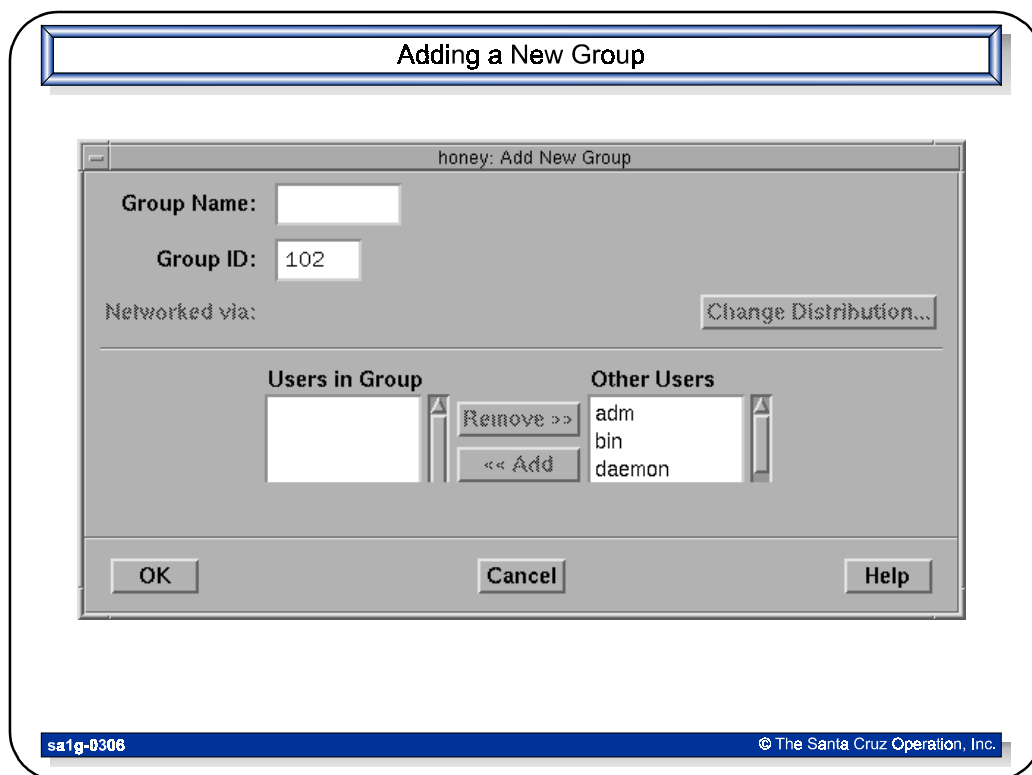
A user can be a member of several groups simultaneously, and can access any file or directory that is within that user's group set, if group permissions allow. By default, files created by a user have the group ID of that user's login group.

As with User ID, and login names, each group should have the same group name and group ID on all systems in the organization.

Exercise: Reviewing group permissions

Refer to the lower graphic on the adjoining page. Who can read and write to the file *sales.dat*? Who can read and write to the file *forecast*?

AV 3-5



3.1.7 Creating a group

These steps describe how to create a new group:

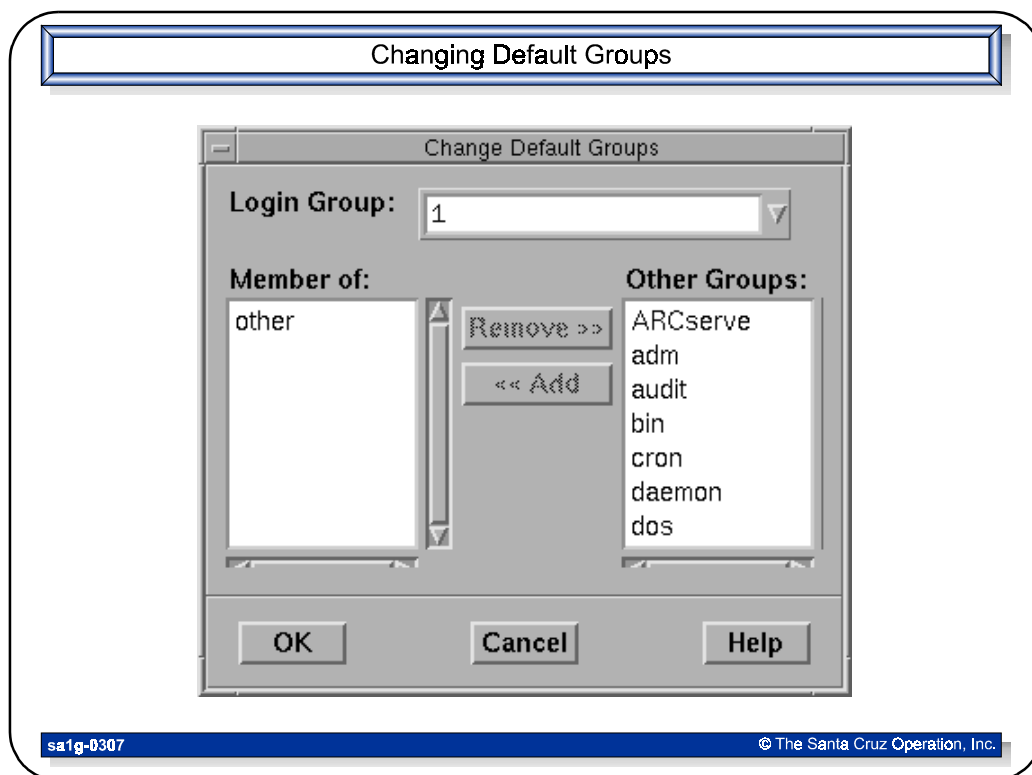
- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select **Groups** ⇒ **Add New Group**.
- Step 3. Enter a group name of 3-8 characters.
- Step 4. Accept the default Group ID or enter a new one.
- Step 5. Add users to the group. Select them from the **Other Users** list and select **Add**. Remove users from the group. Select them from the **Users in Group** list and select **Remove**.
- Step 6. Click on **OK** to create the group.
- Step 7. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

Group information is stored in */etc/group*.

Exercise: Creating a group

Using the steps under Section 3.1.6, “Groups,” on page 3-9, create a new group called *class*. Add **user1** and **user2** to the *class* group.

AV 3-6



3.1.8 Changing the default login group

If the majority of users belong to a specific group, you might want to set that group to be the default login group. On a newly installed SCO OpenServer system, *group* is the default login group, for UnixWare 7, *other* is the default login group.

These steps describe how to change the default login group for new users:

- Step 1. Start SCOadmin. Select **Account Manager**.
- Step 2. Select **Options** ⇒ **User Defaults**.
- Step 3. Click on **Change Group Membership**.
- Step 4. Enter or Select the login group. When entering a new group, you are prompted to create the new group.
- Step 5. Click on **OK** to accept changes.
- Step 6. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

Exercise: Modifying the default login group

Change the default login group to *class*. Will this change an existing account's login group?

AV 3-7

Changing Permissions

Each permission is assigned a number:

read = 4 write = 2 execute = 1

Add these numbers for each user category:

	OWNER	GROUP	OTHER
-	r w -	r - -	r - -
	4 2	4	4
	6	4	4

Use with chmod(1):

\$ chmod 644 filename

3.1.9 Reviewing permissions

Each file and directory has permissions for three categories of users:

User	The owner of the file
Group	Members of a designated group
Other	Other users

Users can change the permissions of files and directories that they own. Permissions are set with the **chmod** command. This command includes numeric options for setting permissions:

Read	4
Write	2
Execute	1

Add these numbers to determine the permission for the owner, group, or others. This number sequence is known as the file's *mode*.

For example, to set the permissions for the */etc/motd* file, enter:

```
chmod 664 /etc/motd
```

The resulting file permissions are:

```
-rw-rw-r--
```

In this example:

- Owner permissions = read (4) + write (2) = 6
- Group permissions = read (4) + write (2) = 6
- Other permissions = read (4) = 4

AV 3-8

Default Discretionary Access Control		
Directories:	(7) (7) (7)	
Unrestricted permissions:	drwxrwxrwx	
— umask 022	0 2 2	
Permissions of new directories	drwxr-xr-x	
	(7) (5) (5)	
Ordinary files:	(7) (7) (7)	
Unrestricted permissions:	-rwxrwxrwx	
— umask 022	0 2 2	
— No execute permission	---x---x---x	
Permissions of new files	-rw-r--r--	
	(6) (4) (4)	

sa1g-0309 © The Santa Cruz Operation, Inc.

3.1.10 Default Discretionary Access Control (DAC)

When a file or directory is created, it is set with default permissions defined by the system. Typical permissions are:

For files `-rw-r--r--`

For directories `drwxr-xr-x`

If a group of users shares a common set of directories and files, you can change the default permissions for users within that group. For example, you might want the default permissions to grant write permissions to all members within the group.

3.1.10.1 Review of `umask`

Default DAC is set with the **`umask`** command. This command specifies which permissions to *withhold*.

For example, to set the default permissions:

Owner All permissions (none excluded)

Group All permissions (none excluded)

Others Read (r) and search (x) permissions (write excluded)

Use this command:

`umask 002`

Given these defaults, new files are set to the mode 664 (`-rw-rw-r--`), and new directories are set to mode 775 (`drwxrwxr-x`).

3.1.10.2 Modifying system-wide umask defaults

Without any **umask** setting, permissions are completely open on both new files (-rw-rw-rw-) and new directories (drwxrwxrwx). An initial **umask** is defined in */etc/profile* and */etc/cshrc*. These files are discussed in Section 3.1.12, “System environment files,” on page 3-20.

The default **umask** in these files is determined by the security profile of the system:

Low and Traditional	022
Improved, under SCO OpenServer	027
Improved, under UnixWare 7	077
High	077

To modify this setting, edit the appropriate environment file.

Exercise: Setting umasks

Situation 1: Set up the Bourne and Korn shell system-wide **umask** so that only the owner has permissions to view or modify a file, or change into a directory. What are the permissions for any new file created? For any new directory?

Situation 2: Set up the C shell system-wide **umask** so that only the owner can modify a file or directory, but everyone can read or execute. What are the permissions for any new file created? For any new directory?

Note that many third-party applications set their own **umask**, so the permissions on files created with those applications may not reflect the system or user’s **umask**.

3.1.11 Login environment

When the system creates a user's home directory, it populates the directory with the appropriate startup files for the login shell. These files set the user's working environment, including the system prompt, the command search path, and other environment variables.

You can modify the system-wide default startup files to match the needs of your users. Listed below are some of the changes commonly made:

- Adding a variable identifying a default printer for the user
- Modifying the user's search path
- Modifying the user's default terminal settings

3.1.11.1 Default account startup files

When you create an account using the **SCOadmin Account Manager**, default account startup files are copied from the following directories:

SCO OpenServer: */usr/lib/mkuser/**

UnixWare 7: */usr/lib/scoadmin/account/skel*

These files can contain default environment settings for the:

- Terminal type
- Search path
- Mail folder

3.1.12 System environment files

In addition to the default account files, these system environment files are executed before executing the user startup files:

/etc/profile Executed for both the Bourne and Korn shells

/etc/cshrc Executed for C shells

Among other things, these files are configured by default to:

- Set the user's initial umask value
- Inform the user if there is unread mail waiting in the mailbox
- Display the "message of the day" file, */etc/motd*

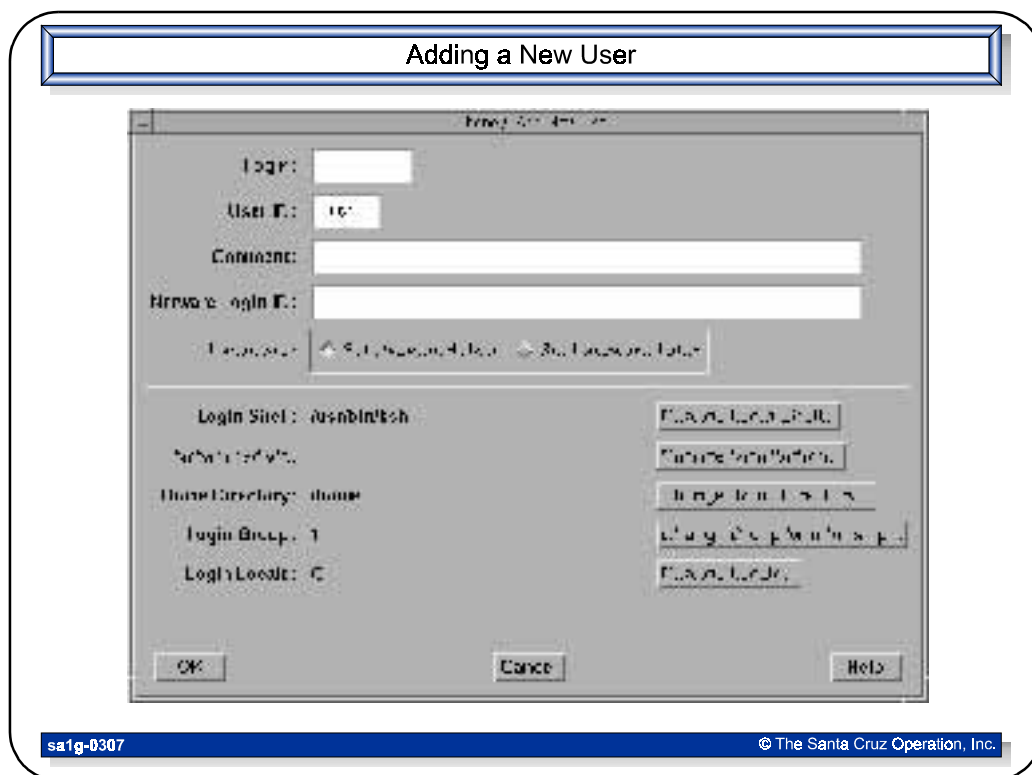
3.2 Creating User Accounts

When a new person requires access to the system, you must create a user account. You can either accept the default account, or modify some of the account parameters. You can also set the password for the new account to avoid unauthorized access.

After using the system for a while, a user might want to use a different login shell or belong to different groups. Some users might forget their account passwords. In those cases, you can modify the current account settings.

User accounts can be managed from either the **SCOadmin Account Manager** or the command-line interfaces.

AV 3-9



3.2.1 Creating a default account

These steps describe how to create a user account using system defaults:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select **Users** ⇒ **Add New User**.
- Step 3. Enter a login name. Accept the UID.
- Step 4. Enter information about the user into the **Comment** field. You can enter information about this user, such as full name, department, and extension.
- Step 5. Accept the defaults for the remaining fields.
- Step 6. Select **Set password now**.
- Note that, if you select **Set password later**, the user cannot log in until you set the initial password.
- Step 7. Click on **OK** to continue.
- The system prompts you to set the initial password for the account.
- Step 8. Do one of the following:
- Set the password; the system prompts you to enter the password twice.
 - Remove a password if the security profile allows. This lets a user login without a password. It is not recommended.
- Step 9. Select whether or not you want to force the user to change the password at the first login.
- Step 10. Click on **OK**. The system creates the user account.
- Step 11. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

3.2.2 Creating an account using modified defaults

These steps describe how to create a user account using modified defaults:

Step 1. Start **SCOadmin**. Select **Account Manager**.

Step 2. Select **Users** ⇒ **Add New User**.

Step 3. Enter the login name.

Step 4. Enter a different **User ID**.

Step 5. Enter information about the user into the **Comment** field.

Step 6. Select **Set password now**.

Note that, if you select **Set password later**, the user cannot log in until you set the initial password.

Step 7. Click on **Change Login Shell** and select the appropriate shell.

Click on **Add Shell Environment Files to Home Directory** and set up the default user environment for that shell. Click on **OK** to continue.

Step 8. Click on **Change Distribution** and select one of the following: local, Advanced Server, or NIS network.

Step 9. Click on **Change Home Directory**, enter the new directory location, and set the permissions for the home directory. Click on **OK** to continue.

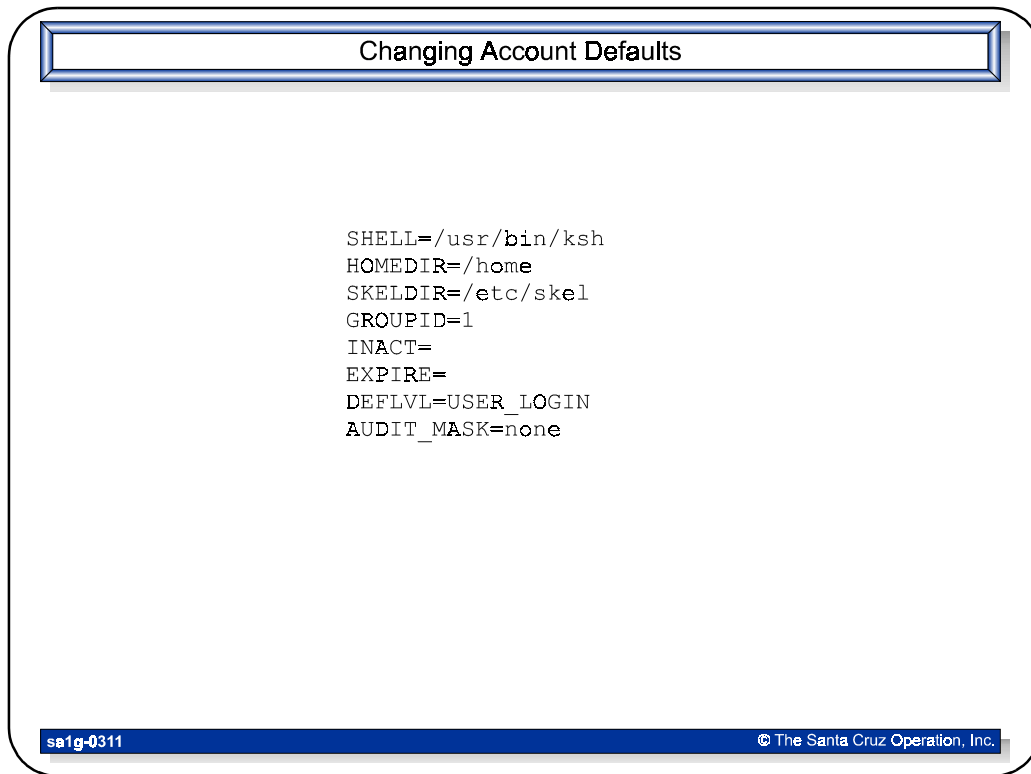
Step 10. Click on **Change Group Membership** and set the login group. A list of groups and a list of groups to which the account belongs displays. **Add** and **Remove** groups between those two lists. Click on **OK** to continue.

Step 11. Click on **Change Locale** and select a new locale. Click on **OK** to continue. Click on **OK** to create the account (UnixWare 7 only).

Step 12. Set the password for this account.

Step 13. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

AV 3-10



Exercise: Creating a user account

In an earlier exercise, the default login shell was changed to the C shell. Change the default login shell to the Korn shell before creating this user. Refer to the steps under Section 3.1.4, “Login Shell,” on page 3-7.

Create an additional user account for yourself. Select your login name. Add *adm* to your group membership.

3.2.3 Creating user accounts with **useradd**

You can also use the **useradd** command to create user accounts. Some administrators find it faster using command-line options as opposed to a graphical interface.

3.2.3.1 Changing account defaults

Recall when creating accounts with the **Account Manager**, defaults are established by accessing **Options** ⇒ **User Defaults**. When you create accounts using the **useradd** command, for SCO OpenServer the defaults are stored in */etc/default/accounts*. For UnixWare 7 the defaults are stored in */etc/default/useradd*.

There are several ways that you can change these defaults:

- Use the **SCOadmin Account Manager**
- Use a text editor to edit the */etc/default/accounts* or */etc/default/useradd* file
- Use the **defadm** command (UnixWare 7 only)

For example, on UnixWare 7, to change the default login shell to the C shell, enter:

```
defadm useradd SHELL=/bin/csh
```

3.2.3.2 Creating user accounts

For example, to create an account using the defaults, with home directory creation, and a user ID of 250, enter:

```
useradd -m -u 250 login_name
```

The user is not able to log in until a password is assigned.

Exercise: Creating a user account with useradd

Create a user account with a UID of 150 and a comment about the user. Refer to **useradd** for the option that is used to add the comment. Enclose the comment with quotes.

3.3 Deleting Users

When a user leaves the company, you should disable that user's account to prevent any unauthorized access. To disable a user account:

- Deactivate the account. No one is able to log into the system using the login name, and the login name is unavailable for reuse.
- Delete the account so that the login name and user ID can be reused

3.3.1 Managing user files

Before you deactivate or delete a user account, you must decide what to do with the user's files and directories. Leaving the files unchanged can mean that disk space is used unnecessarily. Also, any files that belong to that user might be inaccessible except by the **root** account.

You have two options:

- Back up the files and directories, then delete any files or directories owned by the user. This frees up disk space, but can remove files that other users require. If at a later date, you need a file, you can restore it from the backup.
- Assign ownership of the files to another user. That user can then decide which files to keep and which to remove.

Use the **find** command to locate any files owned by a particular user.

Exercise: Locating user files

Use **find** to locate any files owned by your new user account. For example, if the new user login is **user1**, enter:

```
find / -user user1 -print | pg
```

3.3.2 Deleting users

As a general rule, if you add users using the **Account Manager**, use the **Account Manager** to delete your users. If you add your users using the **useradd** command, use the **userdel** command to delete your users.

3.3.2.1 Using the SCOadmin Account Manager

WARNING: In UnixWare 7 the user's home directory and its contents is removed!

These steps describe how to delete a user account:

- Step 1. Back up the user's files.
- Step 2. Start **SCOadmin**. Select **Account Manager**.
- Step 3. Select the user account.
- Step 4. Select **Users** ⇒ **Delete**. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

3.3.2.2 Using the userdel command

To delete a user from the command line, use **userdel** as follows:

userdel login_name

SCO OpenServer: In SCO OpenServer, an additional command, **rmuser**, offers functionality similar to **userdel** for deleting user accounts. Refer to the OpenServer on-line documentation for additional information on the **rmuser** command.

UnixWare 7: In UnixWare 7, the **userdel** command offers additional flexibility when deleting user accounts. For example, you can specify whether the home directory and its' contents are to be removed and whether the UID should be reused in a number of months, reused immediately, or never reused. Refer to the UnixWare on-line documentation for additional information on the **userdel** command

Exercise: Using the userdel command

Use the **userdel** command to remove one of the accounts you created earlier?

What happened to the files in the user's home directory?

3.3.2.3 Re-using UIDs under UnixWare 7

In UnixWare 7, when a user is deleted from the system, the UID of the user is stored in the */etc/security/ia/ageduid* file.

By default, the UID is kept in this file for twelve (12) months. You can change this parameter by changing the UIDAGE parameter in the */etc/default/userdel* file.

If the UID is:

- Never to be reused, enter UIDAGE=-1
- Reused immediately, enter UIDAGE=0
- Reused in six (6) months, enter UIDAGE=6

3.3.2.4 Retiring user accounts under SCO OpenServer

For SCO OpenServer systems configured at Improved or High security levels, accounts cannot be deleted. Instead they must be retired to disable them. Once retired, the login name and UID are not available for reassignment.

These steps describe how to retire an SCO OpenServer account:

- Step 6. Start **SCOadmin**. Select **Account Manager**.
- Step 7. Select the user account.
- Step 8. Select **Users** ⇒ **Retire**. Click on **OK**.
- Step 9. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

OpenServer accounts can be retired regardless of the system's security profile, but accounts cannot be unretired when the system is configured at the High (C2) security profile.

To reactivate a retired account on non-high security profile systems, use the **unretire** command as follows:

unretire login_name

You must change the password after using **unretire**

3.4 Modifying Accounts

You can make changes to individual accounts and change system-wide defaults that determine the settings for new accounts. Modifications for individual accounts or system-wide can include:

- Environment variables and location of home directories
- **umask** value
- Passwords or password expiration

Changes on individual accounts take place upon the next user log in.

Changing system-wide defaults affect the default settings when creating new accounts.

3.4.1 Modifying individual account information

When modifying a user account using the **SCOadmin Account Manager**, you can change any of the values except the login name and user ID.

Follow these steps to modify an existing account:

Step 1. Start **SCOadmin**. Select **Account Manager**.

Step 2. Select an account to modify.

Step 3. Select **Users** ⇒ **Modify**.

The screen is the same as the screen used to create a new user.

Step 4. Select a **Change** option(s) and make the necessary changes. For example, to change the login shell, select **Change Login Shell**.

Step 5. Click on **OK** after each change.

Step 6. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

3.4.2 Using usermod

The **usermod** command offers more flexibility than the **SCOadmin Account Manager** when modifying user accounts. You can change the:

- UID number
- Login name

To change the UID number, enter:

```
usermod -u uid login_name
```

After changing the UID number, files and directories owned by the user still reflect the old UID. Use the **chown** and **chgrp** commands to change ownership of all files and directories owned by the user. Don't forget the user's home directory.

3.4.2.1 UnixWare 7: Enhanced usermod functionality

UnixWare 7 provides enhanced **usermod** functionality, including:

- The ability to set an expiration date on an account. For example, it might be necessary to set up a temporary user account so that the login expires when the account is no longer in use.
- The ability to deactivate an account by setting the “inactive” field of the login.

For more information on these and other options to **usermod**, refer to the on-line documentation.

Note that both SCO OpenServer and Unixware 7 allow the configuration of account expiration and locking through the **SCOadmin Account Manager**.

3.4.3 Modifying an individual account environment

When the home directory (*\$HOME*) is created for the user, copies of the default startup files are placed there. Listed below are the shells and the names of their respective startup files:

Bourne shell	<i>\$HOME/.profile</i>
C shell	<i>\$HOME/.cshrc</i> and <i>\$HOME/.login</i>
Korn shell	<i>\$HOME/.kshrc</i> and <i>\$HOME/.profile</i>

3.4.4 Modifying the umask of a single account

You can set up a user's environment to use a specific default by including a **umask** statement in the shell's startup file, such as *\$HOME/.profile* or *\$HOME/.login*. This overrides the system-wide default. Note that this **umask** could create a potential security breach because the permissions on all new files will differ from the system defaults.

If you change an individual's **umask**, you might have to use the **chmod** command to set the permissions on existing files to match the new environment.

3.4.5 Modifying user passwords

These steps describe how to modify a user's password:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select the user.
- Step 3. Select **Users** ⇒ **Change Password**.
- Step 4. Set the password.
- Step 5. Click on **OK**.
- Step 6. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

To modify a user's password from the command line, use the **passwd** command.

3.4.6 Controlling password expiration

You can control password expiration for an individual user. These steps describe how to change the password expiration values:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select the user.
- Step 3. **SCO OpenServer**: Select **Users** ⇒ **Password Restrictions** ⇒ **Expiration**.
UnixWare 7: Select **Users** ⇒ **Expiration**
- Step 4. Select the default or enter the values for:
- **Days required between changes**. This is the number of days until the password can be changed.
 - **Days before password expires**. This is the maximum number of days a password is valid. This number must be larger than the number of days required between changes.
 - **Days until account is warned**. This is the number of days before the password expires that a warning is issued to change your password. This number should be less than the number of days until the password expires.
- Step 5. Click on **OK**.
- Step 6. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

Password values can also be changed using the **passwd** command.

3.4.7 Controlling password options with default files

By assigning values to a set of parameters in the */etc/default/passwd* file, an administrator can control the aging and length of passwords. The following are some of the parameters available.

MINWEEKS	Minimum number of weeks before a password can be changed.
----------	---

MAXWEEKS	Maximum number of weeks a password can be unchanged.
----------	--

PASSLENGTH	Minimum number of characters in a password.
------------	---

Two parameters located in */etc/default/login* control the existence of passwords.

PASSREQ	If set to YES, all users must have a password. Users who do not have a password are requested for one at the next opportunity permitted by password aging.
---------	--

MANDPASS	(UnixWare 7 only) If set to YES, all users must have a password. The PASSREQ parameter is overwritten and users who have a NULL password are denied access.
----------	---

Exercise: Changing the password length

Edit the */etc/default/passwd* file and change the minimum password length to five (5).

3.5 Managing User Groups

You can add and remove users from groups, as well as remove groups.

3.5.1 Modifying group membership

These steps describe how to modify a group's membership:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select **View** ⇒ **By Groups**. Select the appropriate group.
- Step 3. Select **Groups** ⇒ **Modify**. Add and remove users. Click on **OK**.
- Step 4. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

Use the **chgrp** command to change group ownership on the files.

3.5.2 Deleting a group

These steps describe how to delete a group:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select **View** ⇒ **By Groups**. Select the appropriate group.
- Step 3. Select **Groups** ⇒ **Delete**. Click on **OK**.
- Step 4. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

After you remove a group, you should also:

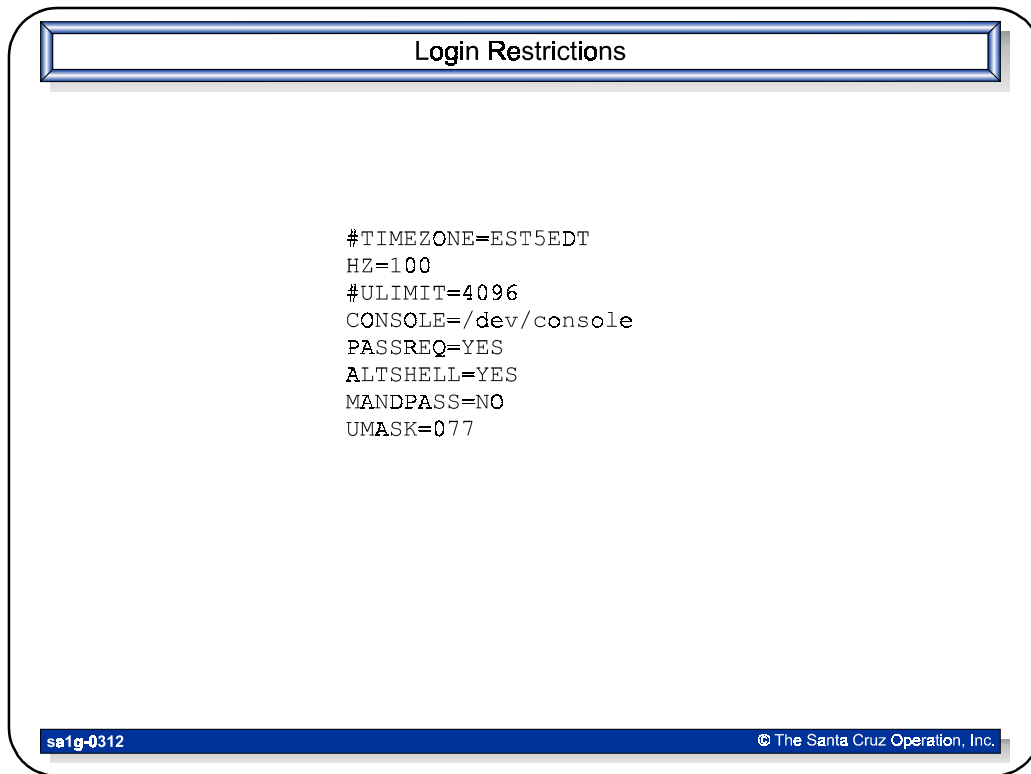
- Set the default login group to a different group if this group was the default login group.
- Use the **chgrp** command to change group ownership of the files still assigned to the deleted group.

3.6 Setting Login Attempt Restrictions

One way that a person can try to break into a system is to repeatedly attempt to log into a terminal by guessing the password for specific accounts. You can protect both terminals and user accounts by limiting the number of consecutive unsuccessful login attempts. The system locks the account or terminal if the unsuccessful attempts exceed the specified limit.

Of course, if legitimate users forget their passwords, their accounts or terminals can be locked. In that case, unlock accounts and terminals so that those users can access the system again.

AV 3-11



3.6.1 SCO OpenServer login restrictions for user accounts

Increase the difficulty of users logging into your SCO OpenServer system using the **SCOadmin Account Manager**:

Step 1. Start **SCOadmin**. Select **Account Manager**.

Step 2. Select the user.

Step 3. Select **Users** ⇒ **Login Controls**

The window displays the last successful login, the last failed login, the last logout, and the number of failed login attempts since the last successful login.

Step 4. Deselect the default or enter a new limit:

Step 5. Click on **OK**.

Step 6. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

3.6.2 UnixWare 7 login restrictions for user accounts

Increase the difficulty of users logging into your UnixWare 7 system by setting these parameters in the */etc/default/login* file.

DISABLETIME	Interval between login prompts after an unsuccessful login attempt. The default is ten (10) seconds.
MAXTRYs	Maximum number of login attempts allowed. The default is 99 attempts.
TIMEOUT	Maximum interval for a login attempt. When this period expires, the login is considered unsuccessful and the session is closed. The default is 60 seconds.

3.6.2.1 UnixWare 7: Recording unsuccessful login attempts

If you suspect that someone is trying to break into your UnixWare 7 system, you can record unsuccessful login attempts. Record unsuccessful login attempts by adding the **LOGFAILURES** parameter with the number of unsuccessful attempts to the */etc/default/login* file.

The default number of login attempts is 99.

You must also create a */var/adm/loginlog* file. The file owner must be **root**, a member of the group **sys**, and the permissions read and write by the owner.

Exercise: Recording unsuccessful login attempts

Using the steps under Section 3.6.2, “UnixWare 7 login restrictions for user accounts,” on page 3-41 as a guide, change the **LOGFAILURES** parameter to three (3).

Create the */var/adm/loginlog* file, enter:

```
> /var/adm/loginlog
```

Remember to change ownership to **root** and group ownership to **sys**.

Using your student account, log into your system and enter an incorrect password. Log in as **root** and view the */var/adm/loginlog* file.

What information is recorded about your login attempt? If the login attempt is not recorded, what is your problem?

3.6.2.2 SCO OpenServer: Reporting unsuccessful login attempts

If you suspect that someone is trying to break into your SCO OpenServer system, you can run a report using the **SCOadmin Reports Manager**. This manager will be covered in more detail shortly.

3.6.3 Locking and unlocking a user account

If a user is going to be away for an extended period of time, you can lock the account to prevent unauthorized logins.

Follow these steps to lock an account:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select the user account.
- Step 3. **SCO OpenServer**: Select **Users** ⇒ **Login Controls** ⇒ **Lock Account**.
UnixWare 7: Select **Users** ⇒ **Lock**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

Follow these steps to unlock an account:

- Step 1. Start **SCOadmin**. Select **Account Manager**.
- Step 2. Select the user account.
- Step 3. **SCO OpenServer**: Select **Users** ⇒ **Login Controls** ⇒ **Unlock Account**.
UnixWare 7: Select **Users** ⇒ **Unlock**. Click on “Enter a New Password.” Enter the new password.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

To lock an account from the command line use the **passwd** command:

```
passwd -l login_name
```

To unlock an account from the command line use the **passwd** command:

```
SCO OpenServer: passwd -u login_name
```

```
UnixWare 7: passwd login_name (assign a new password)
```

3.7 Using Account Activity Reports

Part of maintaining system security is watching for potential problems. Expired passwords, locked terminals and accounts, and repeated unsuccessful logins can all indicate either a security problem or an education problem for your users.

Because a system often has a large number of users, it is impractical to use the **Account Manager** to examine each individual account for potential problems. **SCOadmin** provides a **Reports Manager** that generates system-wide reports for account, password, and terminal status.

3.7.1 Reporting password status

You can display a system-wide password status report that includes all users or individual users. This report shows the:

- Status of accounts (locked or unlocked)
- Date the password was changed
- Password expiration controls

To display password status:

- Step 1. Enter **scoadmin reports**. Select **Reports** ⇒ **Password Status**.
- Step 2. **SCO OpenServer**: Select **Accounts with password soon to expire**
UnixWare 7: Select **Systemwide** to display a report for all users.
- Step 3. Select **Host** ⇒ **Exit** to exit the **Reports Manager**.

Exercise: Running the password status report

Use the **Reports Manager** and run a password status report for your student account. What is the status of your password expiration controls?

3.7.2 Reporting login activity

The user login status report can address a single user, terminal, all users, or the current user. In a networking environment, you can also display and print a status report by each system. This report provides the:

- User name
- Terminal associated with the login
- Time of login
- Status of the login
- Last logout time
- System logged on

To display login activity:

Step 1. Enter **scoadmin reports**.

Step 2. **SCO OpenServer**: Select **Reports** ⇒ **User Login Status**
UnixWare 7: Select **Reports** ⇒ **User Logins**.

Step 3. Select the type of report.

Step 4. Select **Host** ⇒ **Exit** to exit the **Reports Manager**.

To print the **Password Status** or **User Logins** report while in the **Reports Manager**, select **Host** ⇒ **Print report**.

Exercise: Running the Report Manager

Print the **User Logins** report for all users connected to your system. Save this information to a file in the */tmp* directory.

3.8 Troubleshooting User Accounts

From time to time users may run into problems using their accounts. Listed below are some of the common problems and the steps to correct them.

3.8.1 Fixing forgotten or compromised passwords

These steps describe what you should do if you need to change a user's password:

- Step 1. Confirm the user's identity.
- Step 2. Unlock the user's account, if necessary.
- Step 3. Change the password. Force the password to be changed at the next login.
- Step 4. Remind the user to keep the password private.

3.8.2 Fixing an incorrect terminal type

Sometimes a user incorrectly sets the terminal type during login. As a result, some applications might not display or interpret keystrokes correctly. To reset the terminal type, you must modify the `TERM` variable.

In either the Bourne or Korn shell, at the user prompt, enter:

```
TERM=terminal_type  
export TERM
```

In the C shell, at the user prompt, enter:

```
setenv TERM terminal_type
```

3.9 Lesson Summary

Before users can access a system, they must have been designated accounts. These accounts set up the users' working environments. In addition to creating and removing accounts, there are a number of elements that you control:

- Which accounts are members of which groups
- How strict are the parameters defining acceptable passwords
- What type of login restrictions are enforced for both accounts and terminals

Review questions

1. Under the Traditional security profile, what is the default value of the **PASSLENGTH** field in */etc/default/passwd*?
2. Which group is the default login group for all users? If you change this group, is the default group automatically changed for existing users?
3. What **umask** value grants the owner and members of the group full access, but denies any access to other users?
4. Under UnixWare 7 which command can you use to change account defaults used by **useradd** in the */etc/default/useradd* file?
5. Which two tasks can you perform using the **usermod** command that you cannot perform using the **SCOadmin Account Manager**?

6. Which file sets an individual account environment using the Bourne shell?

7. Which report in the **SCOadmin Reports Manager** can you generate that gives you the status of the account, the date the password was changed, and the password expiration controls?

8. List the steps to fix a forgotten or compromised password.

3.10 Lab: Creating Groups

In the next two labs, you are going to create user accounts that will belong to different groups. In this first lab, you will set up groups for the organization.

3.10.1 Overview

Create two new groups called **admin** and **sales**.

3.10.2 Procedure

Create the groups.

- Step 1. Start **SCOadmin** ⇒ **Account Manager**.
- Step 2. Select **Groups** ⇒ **Add New Group**.
- Step 3. Enter **admin** as the **Group Name**.
- Step 4. Click on **OK**.
- Step 5. Repeat Steps 2 through 4 to create the **sales** group (enter the appropriate **Group Name** in Step 3).
- Step 6. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

3.11 Lab: Creating New Users

Set up new user accounts and modify them as specified. In addition, assign them to the appropriate groups.

3.11.1 Overview

Create user accounts for two (2) individuals. Assign a password to each account. Modify the default settings according to the individual's requirements:

sales1 Uses the C shell

sysadm Uses the Korn shell; also belongs to the **lp** group.

The **sysadm** account should also belong to the **admin** group. The **sales1** account should belong to the **sales** group.

3.11.2 Procedure

These steps describe how to set up and modify user accounts:

Step 1. Start **SCOadmin** ⇒ **Account Manager**.

Perform these steps for each account.

Step 2. Select **Users** ⇒ **Add New User**.

Step 3. Enter the login name.

Step 4. Enter a comment to describe the account.

Step 5. Select **Set password now**.

Step 6. Select **Change Login Shell**.

Step 7. Select the correct shell for the user.

Step 8. Click on **OK** to set the login shell.

Step 9. Select **Change Group Membership**.

Step 10. Select the group to add from the **Other Groups** list, then select **Add**.

Step 11. Change the login group.

Step 12. Assign other group memberships.

Step 13. Click on **OK** to set the group membership.

Next, you create the account and set the new user password.

Step 14. Click on **OK** from the **Add New User** screen to create the user account. The system sets up the account, including the populated home directory.

Step 15. Enter the initial password when prompted in both the **Enter Password** and **Confirm Password** fields.

Step 16. Select **Force password change at login**, then click on **OK** to set the password.

Step 17. Select **Host** ⇒ **Exit** to exit the **Account Manager**.

3.12 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

defadm	Displays and modifies account defaults in the <i>/etc/default/useradd</i> file (UnixWare 7)
find	Finds user files
passwd	Sets a password. Also locks and unlocks user accounts.
rmuser	Deletes user accounts using a command line interface (SCO OpenServer)
scoadmin	Invokes SCOadmin applications (in this lesson, the Account Manager and the Reports Manager)
setenv	Sets environment for command execution in the C shell
umask	Displays or sets file-creation mode mask
unretire	Unretire a previously retired account (SCO OpenServer)
useradd	Creates user accounts using a command line interface
userdel	Deletes user accounts using a command line interface
usermod	Modifies user accounts using a command line interface

Relevant files

<i>/etc/cshrc</i>	Is the system environment file for the C shell
<i>/etc/default/login</i>	Contains system-wide log in parameters
<i>/etc/default/passwd</i>	Contains password aging parameters
<i>/etc/default/accounts</i>	Contains account defaults used by the useradd command (SCO OpenServer)
<i>/etc/default/useradd</i>	Contains account defaults used by the useradd command (UnixWare 7)
<i>/etc/default/userdel</i>	Contains the UIDAGE parameter (UnixWare 7)
<i>/etc/profile</i>	Is the system environment file for the Bourne or Korn shell
<i>/etc/default/passwd</i>	Is the file specifying default password information
<i>/usr/lib/mkuser/*</i>	Contains default account startup files used by the SCO Admin Account Manager (SCO OpenServer)
<i>/usr/lib/scoadmin/account/skel/*</i>	Contains default account startup files used by the SCO Admin Account Manager (UnixWare 7)
<i>/etc/security/ia/ageduid</i>	Contains used UIDs (UnixWare 7)
<i>\$HOME/.chsrc</i>	Is the account environment file for the C shell
<i>\$HOME/.login</i>	Is the account environment file for the C shell
<i>\$HOME/.kshrc</i>	Is the account environment file for the Korn shell
<i>\$HOME/.profile</i>	Is the account environment file for the Bourne or Korn shell

3.13 Answers to Review Questions

This section provides the answers to review questions presented in this lesson.

Review questions

1. The Traditional security profile sets the **PASSLENGTH** field to three (3) in */etc/default/passwd*.
2. The default login group for UnixWare 7 users is *other*. The default login group for SCO OpenServer users is *group*. When you change the default group, the group is not automatically changed for existing users.
3. The umask value **007** grants the owner and members of the group full access, and denies any access to others.
4. Under UnixWare 7 you can use the **defadm** command to change account defaults in the */etc/default/useradd* file.
5. Using the **usermod** command, you can modify a user's login name and UID.
6. The **\$HOME/.profile** sets an individual account environment using the Bourne shell.
7. You can generate the **Password Status Report** in the **SCOadmin Reports Manager** to give you the status of the account, the date the password was changed, and the password expiration controls.
8. The steps to fix a forgotten or compromised password are: 1. Confirm the user's identify. 2. Unlock the user's account if necessary. 3. Change the password. Force the password to be changed at the next login. 4. Remind the user to keep the password private.

4

Managing Processes

Lesson 4

Managing Processes

Introduction

Every program that runs on SCO OpenServer and UnixWare 7 systems runs as a process. As the system administrator, support your users by ensuring that processes are running efficiently.

You also can automate jobs that clean or monitor the system. You can schedule these maintenance tasks to occur during periods of low system use without administrator or operator intervention.

Objectives

This lesson covers process management and job scheduling. By the end of this lesson, you will be able to:

- Describe a process
- Terminate processes
- Manage processes using the **SCOadmin Process Manager**
- Automate system tasks using **cron**

4.1 Describing a Process

Every event that occurs on SCO OpenServer and UnixWare 7 systems is the result of a process. Both users and the system can generate processes, which can affect the performance of the system. Understanding how processes work can help you identify problems and decide how best to fix them.

4.1.1 Definitions

A process is a single instance of a program in execution. This can be a login shell or an operating system command, but not a built-in shell command. If a command is built into the shell, a separate process is not created on its invocation; the built-in command is issued within the context of the shell process.

Processes differ from programs in that they are actively present in memory, and their state changes with time. A program, on the other hand, does nothing until it is executed and gives rise to a process. Each process is assigned a unique numerical ID. This number is called the process ID or PID.

SCO OpenServer and UnixWare 7 systems support many processes running simultaneously. This is called *multitasking*.

Various system calls enable processes to create other processes, communicate with other processes, and terminate other processes.

4.1.2 The process table

The kernel maintains a table of all running processes. This table can grow dynamically to a maximum size determined by the kernel and available system resources.

Every time a new process is created, the kernel assigns it a unique process identification number (PID) and adds information describing it to the process table.

4.1.3 Viewing current processes

Use the **ps** command to view processes currently in the table. Common options for the **ps** command include:

- f** Generates a full listing of a process
- u** Generates a list of processes owned by the specified user(s)
- t** Generates a list of processes associated with the specified terminal
- e** Displays the entire list of processes

Exercise: Viewing user processes

Use **ps** to view all of the processes associated with your current login.

Use **ps -f** to view more information about the processes associated with your current login.

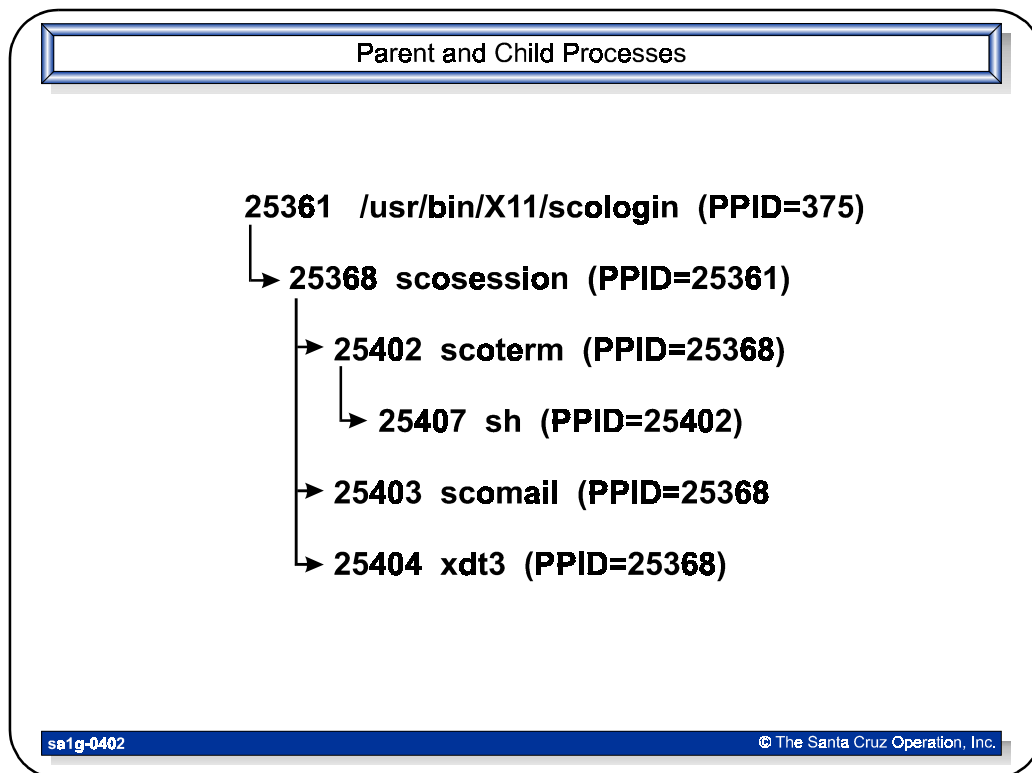
Use **ps -fu root | pg** to view information about all of **root**'s current processes.

SCO OpenServer: Use **ps -ft tty01** to view information about all processes associated with the console.

UnixWare 7: Use **ps -ft console** to view information about all processes associated with the console.

Use **ps -ef | pg** to view information about all processes.

AV 4-1



4.1.4 Process genealogy

A process that creates a new process is called the *parent* process. The new process is called the *child*.

The shell takes most commands and creates a child process to run them. Often, a parent process waits for the child process to complete before continuing.

By starting child processes in the background, one parent process can have many child processes.

A child process can have only one parent process.

4.1.5 Foreground and background processes

When a command is entered at the shell prompt, the shell creates a child process to run the command. The shell waits for the command to exit, then returns the prompt to the user. This command is said to run in the *foreground* because the user cannot execute another command until it completes.

When a command is entered at the shell prompt followed by an ampersand (&), the shell creates a child process to run the command, but does not wait for the command to exit. Instead, it immediately returns the prompt to the user. This command is said to run in the *background*.

Background processes must be non-interactive. Different shells handle foreground and background processes differently.

4.1.6 Daemon processes

A daemon is a background process that performs a regular routine or provides a service without human intervention. A daemon might be started as a response to some event, or it can be executed at regular intervals. One of the most important daemons on the system is **cron**, which regularly runs other programs.

Daemon processes are processes that are not associated with a terminal or an active user session. They perform system tasks on a periodic basis, and sleep between those tasks. The **ps** command indicates daemons with a question mark (?) in the *tty* field.

4.1.7 Defunct processes

A defunct process is a process that has terminated but has not been removed from the process table because its parent did not “wait” for it. While the parent is still alive and not waiting for the child, defunct processes can remain in the process table. The slot in the process table is the only system resource consumed by it.

Large numbers of defunct processes can interfere with the creation of new processes, as the total number of processes that can exist on the system is limited.

Usually, a defunct process is removed when its parent terminates. If it is not removed, **init** inherits and removes it.

Should a defunct process remain, consider rebooting the system to remove it.

4.2 Terminating Processes

Sometimes processes stop functioning. For example, a program might encounter an unexpected error and hang. The process is still in the process table, and sometimes a window still displays, but the process is not running. You must manually terminate such a process.

You might also need to terminate runaway processes. These have not stopped working; they are caught in some form of infinite loop. These usually occur as a result of a user script error. Runaway processes continue to use system time and memory until you terminate them.

4.2.1 The kill command

You can terminate processes using the **kill** command:

kill [-signal] PID

The **kill** command sends a signal to a process which usually results in the termination of that process. Some processes can ignore signals or respond to them without terminating. This is called *trapping* the signal.

To terminate a process with the **kill** command, you must know the process ID number (PID) of the process to terminate. Non-root users can kill only their own processes.

4.2.2 Using the kill command

You send various signals to a process when using the **kill** command. Send them in the following order until the process terminates:

- Press  (**kill -2**) at the terminal with the hung foreground process.
- Press  (**kill -3**) at the terminal with the hung foreground process. This is likely to create a *core* file which contains an image of the program code.
- Run **ps -u account_name** from another terminal or console to get the PID of the hung process.
- Run **kill PID** from another terminal or console. This is the same as **kill -15 PID**. Some programs trap this signal, terminate child processes, and clean up any temporary files before exiting.
- Run **kill PID** on the parent process.

If none of these signals work, run **kill -9 PID**. Processes cannot trap or ignore this signal. Because this signal cannot be trapped, the process cannot clean up temporary files and child processes.

Note that high-priority processes cannot be terminated without rebooting the system.

4.3 Managing Processes with the Process Manager

You can use the **SCOadmin Process Manager** to manage processes on a local or remote system. The **Process Manager** lets you:

- View processes
- Delete (kill) processes

4.3.1 Viewing processes on a system

By default, the **Process Manager** uses these attributes to display all processes on a system:

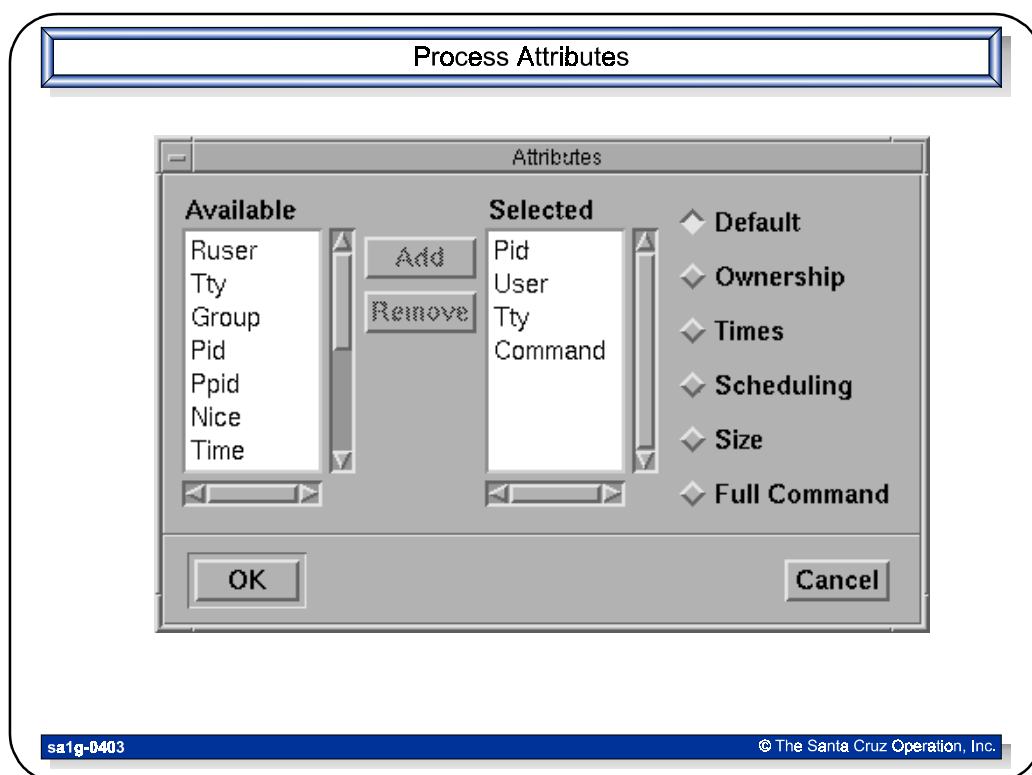
PID	The number that identifies the process.
User	The owner of the process.
Tty	The terminal that is associated with the process.
Command	The command that is executed.

You can also view processes associated with a specific User, Tty, or PID.

These steps describe how to access the **Process Manager** and view processes associated with an individual user:

- Step 1. Start **SCOadmin**. Select **Process Manager**.
- Step 2. Select **View By User**.
- Step 3. Select the user from the **Available** column. Click on **Add**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Process Manager**.

AV 4-2



4.3.1.1 Changing displayed process attributes

You can change the viewing of processes by adding or removing attributes. For example, to add the accumulated time that a process has been running to the default view, perform these steps:

- Step 1. Start **SCOadmin**. Select **Process Manager**.
- Step 2. Select **View ⇒ Attributes**.
- Step 3. **SCO OpenServer**: Select **Cumulative CPU Time** from the **Available** column. Click on **Add**.
UnixWare 7: Select **Time** from the **Available** column. Click on **Add**.
- Step 4. Click on **OK**.

4.3.1.2 Sorting by attributes

You can change the attributes by which processes are sorted. To determine which processes are using the most CPU (accumulated) time, perform these steps:

- Step 1. Start **SCOadmin**. Select **Process Manager**.
- Step 2. Select **View ⇒ Sort**.
- Step 3. Select **Time**. Click on **Decreasing**.
- Step 4. Click on **OK**.

The processes are sorted by the **Time** attribute and display in descending order.

4.3.2 Deleting processes

In Section 4.2.2 on page 4-10 you learned how to use the **kill** command with different signals to terminate processes. You can also use the **Process Manager** to delete (kill) processes.

These terms describe some of the signals used when deleting processes using the **Process Manager**.

INT	Signal 2
QUIT	Signal 3
TERM	Signal 15
KILL	Signal 9

For a complete description of all the signals used by the **Process Manager**, refer to the **signal** manual page.

These steps describe how to send a **TERM** signal to a process using the **Process Manager**:

- Step 1. Start **SCOadmin**. Select **Process Manager**.
- Step 2. Select a process to delete.
- Step 3. Select **Process** ⇒ **Signal**.
- Step 4. Select **TERM**
- Step 5. Click on **OK**.
- Step 6. Select **Host** ⇒ **Exit** to exit the **Process Manager**.

4.4 Scheduling Tasks

One way to make your job easier is to automate system tasks such as unattended backups and system clean ups. The **cron** daemon runs these automated tasks at designated hours on designated days. You can schedule non-interactive tasks to run during off-hours when the system load is lighter.

A scheduled task can be either a single command or a shell script containing many commands. Output of a command is mailed to you unless redirected to a file or terminal.

4.4.1 Setting the system date and time

Before you schedule tasks, verify the system date and time are correct.

These steps describe how to set the correct date and time using the **System Time Manager**:

- Step 1. Start **SCOadmin**. Select **System** ⇒ **System Time Manager**.
- Step 2. **Select Host** ⇒ **Set Time**.
- Step 3. Enter the correct date and time. Note that hours are expressed in a 24-hour format. For example, enter **14** to indicate 2:00 p.m.
- Step 4. Click on **OK** to set the system clock to the specified date and time.
- Step 5. Click on **OK** to accept the date and time.
- Step 6. Select **Host** ⇒ **Exit** to exit the **System Time Manager**.

Use the **date** command to set the system date and time from the command line. SCO OpenServer administrators can also use the **asktime** command.

Exercise: Setting today's date and time

Use the **System Time Manager** to set the system clock to the correct date and time.

4.4.2 Specifying jobs and scheduled times

Jobs and their scheduled times are stored in a text file. Each user can submit only one request file, however the file can contain multiple job requests. A new submission overwrites previous submissions. Each request in the file must contain six fields, separated by spaces or tabs. The file may contain comments (lines beginning with #), but must not contain blank lines.

The six fields, and their permissible values, are as follows:

- Minutes (0-59)
- Hours (0-23)
- Day of Month (0-31)
- Month (1-12)
- Day of Week (0-6, where 0=Sunday)
- Command Text (commands, executable binary, or shell script)

Each of the first five fields can consist of a single integer value, a range of values, or a list of values separated by commas. An asterisk selects all allowable values. For example, the following entry specifies that command `/usr/bin/doit` should be executed at 10 minutes and 40 minutes past the hours of 2am through 6am, Monday through Friday:

```
10,40      2-6   *    *    1-5   /usr/bin/doit
```

4.4.3 Listing cron jobs

To list the current user's cron jobs, use the **crontab** command as follows:

```
crontab -l
```

To save the user's current cron information to a file, redirect the output:

```
crontab -l > cronfile
```

4.4.4 Adding, modifying, and removing cron jobs

To add, modify, or remove individual job requests, follow these steps:

Step 1. Save the user's current cron information to a file, or create a new empty file:

```
crontab -l > cronfile
```

Note: the name of the file is not important, here we use *cronfile*.

Step 2. Use a text editor to modify the file *cronfile*. Add new entries, modify existing entries, or remove entries as needed.

Step 3. Submit the requests using the **crontab** command and the name of the file:

```
crontab cronfile
```

The user's cron requests will be placed in */var/spool/cron/crontabs/<user_name>*

4.4.5 Removing all cron jobs for a user

To cancel all cron activities for a user, follow these steps:

Step 1. Login as the user or **su** to the user's account

Step 2. Run the command **crontab -r**

WARNING! This command should not be run by root as it would remove cron jobs used for regular system maintenance.

4.4.6 Authorizing users to schedule tasks

The following files control which users can submit cron jobs

<i>/usr/lib/cron/cron.allow</i>	An explicit list of users that can schedule cron jobs. An empty file indicates that no user can schedule jobs.
---------------------------------	--

<i>/usr/lib/cron/cron.deny</i>	An explicit list of users that cannot schedule cron jobs. An empty file indicates that any user can schedule jobs.
--------------------------------	--

If neither of these files exists, any user can schedule cron jobs. If both these files exist, *cron.allow* takes precedence.

4.4.7 UnixWare 7: Using the Task Scheduler

UnixWare 7 provides a graphical interface, the **SCOadmin Task Scheduler**, to help simplify the creation of cron jobs. With the **Task Scheduler**, you can:

- Authorize users to schedule tasks
- Add or modify tasks
- View scheduled tasks
- Delete scheduled tasks

4.4.7.1 Authorizing users to schedule tasks

Follow these steps to authorize users on your system to schedule tasks:

- Step 1. Start **SCOadmin** ⇒ Select **Task Scheduler**.
- Step 2. Select **Task** ⇒ **Authorization**.
- Step 3. Select a name of the user from the “Denied” column. Click on **Add**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Task Scheduler**.

In Step 3, to remove a user’s authorization, select the name of the user from the “Allowed” column and click on **Deny**.

4.4.7.2 Adding scheduled tasks

Recall that a task can be a single command, shell script, or any application program that can run unattended. When you or a user adds tasks, the **Task Scheduler** creates a single file for each user containing the scheduled times and tasks. This file is:

- Named for the user scheduling the task
- Placed in the `/var/spool/cron/crontabs` directory

Before adding a scheduled task, decide:

- The name of the user running the task.
- The full pathname of the command or program. If the full pathname is not entered, the **Task Scheduler** examines the user's home directory.
- The day of the week or the date of the month that the task is scheduled.
- The time. The task can be scheduled to run at a specific time or every hour. Hours are expressed in a 24-hour format. Task can also be scheduled to run every minute.

For example, to add a task that runs at 1:00 p.m. on Wednesday and Friday, perform these steps:

Step 1. Start **SCOadmin**. Select **Task Scheduler**

The next step is optional. The task is created for the user who is running the **Task Scheduler**. If you are logged in as **root**, and want to create a task for a different user, you must perform Step 2.

Step 2. Select **View ⇒ Tasks by Users**. Select the authorized user.

Step 3. Click on **Task ⇒ Add Task** and enter the full pathname of the command or program.

Step 4. Click on **Day of Week**. Select **Wednesday** and **Friday** as the days of the week.

Step 5. Click on "Every Hour" and "Every Minute." This "turns off" this option.

Step 6. Click on **Hour**. Select 13. Click on **Minute**. Select 0. Click on **OK**.

Step 7. Select **Host ⇒ Exit** to exit the **Task Scheduler**.

4.4.7.3 Modifying scheduled tasks

When you are logged in as **root** and enter the **Task Scheduler**, **root**'s scheduled tasks display. From the **Task Scheduler**, to modify a task:

- Step 1. Select the task to modify.
- Step 2. Click on **Task ⇒ Modify Task**.
- Step 3. Modify the task by following Steps 4 through 7 under Section 4.4.7.2, "Adding scheduled tasks," on page 4-19.

The **root** user can modify any user's tasks. Other users can only modify their own tasks.

When you are logged in as **root** and want to modify another user's scheduled tasks, perform these steps. From the **Task Scheduler**:

- Step 1. Select **View ⇒ Tasks by Users**.
- Step 2. Select the name of the user. Click on **OK**.
- Step 3. Click on **Task ⇒ Modify Task**.
- Step 4. Modify the task by following Steps 4 through 7 under Section 4.4.7.2, "Adding scheduled tasks," on page 4-19.

4.4.7.4 Deleting scheduled tasks

To delete a task, perform these steps. From the **Task Scheduler**:

- Step 1. Select the task.
- Step 2. Click on **Delete**.
- Step 3. Click on **OK**.

The **root** user can delete any user's task. Other users can only delete their own tasks.

If you are logged in as **root** and want to delete another user's task, first list the user's tasks. Click on **View ⇒ Tasks by Users**.

4.5 Modifying System Tasks

In addition to creating your own scheduled tasks, some tasks are automated by default. You can modify these tasks or add tasks to each file. These files are stored in the */var/spool/cron/crontabs* directory:

<i>adm</i>	You can use this file to clear login logfiles and to collect system startup and performance information.
<i>lp</i>	Removes log information about the print service that is recorded in the <i>/var/lp/logs</i> directory (UnixWare 7)
<i>root</i>	Performs various administrative functions.
<i>sys</i>	Collects performance information using the sar command
<i>uucp</i>	Automates and performs maintenance tasks for UUCP

It is recommended that you not remove these files. If you do not want these tasks to run, precede the file entries with a pound (#) sign.

4.6 Lesson Summary

Processes comprise a large portion of system operation. From time to time you will need to terminate hung processes so that a user can resume working. You can also schedule jobs to automate routine administration tasks.

Review questions

1. What is the difference between a program and a process?
2. Why is it preferable to use **kill -15** instead of **kill -9**?
3. List the steps to add the Accumulated CPU Time field for viewing processes using the **SCOadmin Process Manager**.
4. Which factors should you consider before scheduling a cron job?

4.7 Lab: Scheduling Processes

In this lab you will schedule tasks to run automatically.

4.7.1 Overview

Using your student account, create a scheduled task to mail the last 50 lines of */var/adm/syslog* to **root** at 8:00am on Mondays.

4.7.2 Procedure

Create the task:

Step 1. Log in using your student account.

Step 2. Create an executable file in your home directory. Name the file *mailroot*. Enter this line into the *mailroot* file:

```
tail -50 /var/adm/syslog | mail root
```

Step 3. Save the user's current cron information to a file, or create a new empty file:

```
crontab -l > cronfile
```

Step 4. Use a text editor to modify *cronfile*. Add the appropriate entry to run the **mailroot** script, created above, at the specified day and time.

Step 5. Submit the requests using the **crontab** command and the name of the file:

```
crontab cronfile
```

The user's cron requests will be place in */var/spool/cron/crontabs/<user_name>*

4.8 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

asktime	Sets the system time (SCO OpenServer only)
cron	Executes commands scheduled by crontab
crontab	Schedules commands to be executed at regular intervals
kill	Sends a kill signal to a process
ps	Displays information about active processes
sar	Reports usage information for individual processors, as well as summary information for average processor usage
scoadmin	Invokes SCOadmin applications (in this lesson, the Cron Manager and the System Time Manager)

Relevant files

<i>\$HOME/cronfile</i>	Suggested file containing a user's cron requests
<i>/usr/lib/cron/cron.allow</i>	A list of users allowed to submit cron jobs
<i>/usr/lib/cron/cron.deny</i>	A list of users denied the ability to submit cron jobs
<i>/var/spool/cron/crontabs/*</i>	Cron files for users who have submitted cron jobs

4.9 Answers to Exercises and Review Questions

This section provides the answers to exercises and review questions presented in this lesson.

Review questions

1. The difference between a program and process is that a program is an executable file in the filesystem, while a process is an instance of a program that is currently executing.
2. It is preferable to use kill -15 instead of kill -9 because many processes will trap a signal 15 and remove any temporary files before exiting. A signal 9 cannot be trapped, and any temporary files created would not be deleted.
3. The steps to add the Accumulated Time field to the Process Manager are: run **scoadmin** Double click on **Process Manager**. Select **View**⇒ **Attributes**. Select **Time**. Click on **Add**. Click on **OK**.
4. The factors that you should consider before scheduling a cron job are the user name of the person running the task, full pathname of the program, the day of the week or the date of the month that the task is scheduled, and the time.

5

Working with Filesystems

AV 5-1

Working with Filesystems

- Describe filesystem basics
- Use different filesystem types
- Mount and unmount a filesystem
- Repair a non-root filesystem
- Monitor filesystems
- Locate and manage user files
- Manage system logs and temporary directories

sa1g-0501 © The Santa Cruz Operation, Inc.

Lesson 5

Working with Filesystems

Introduction

A filesystem is a structure that allows the UNIX operating system to store and access files and directories. Unless you make a filesystem available to your users, they cannot access any of the stored information.

If a filesystem becomes damaged, you are unable to access any of the data until it is repaired.

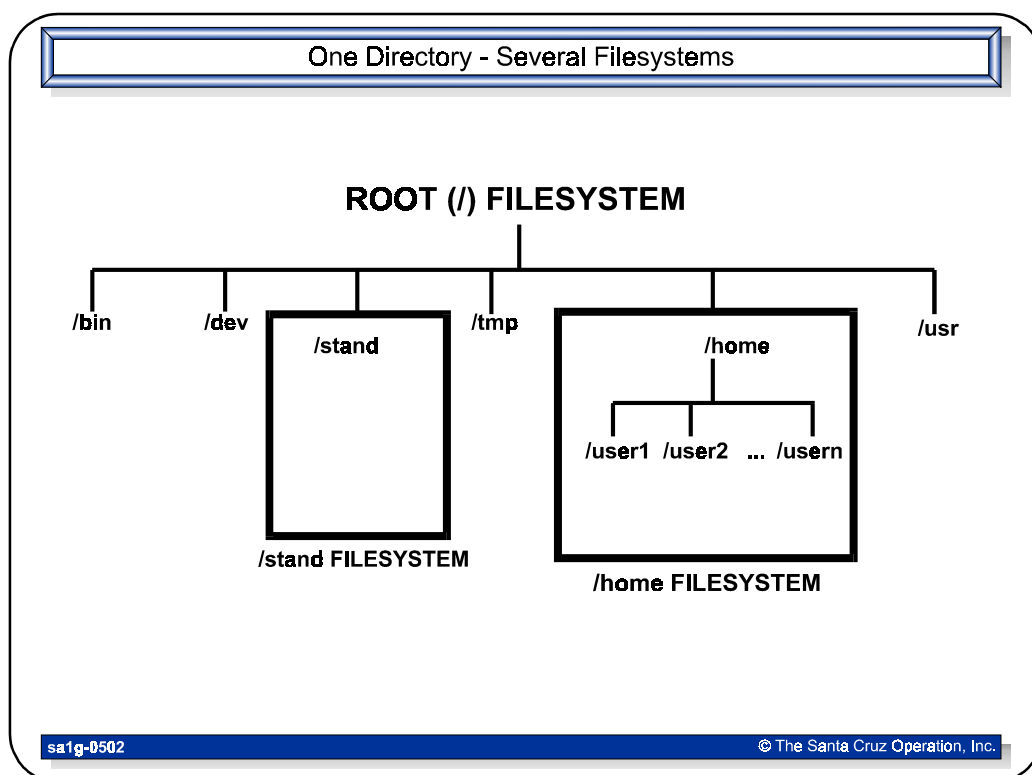
If a filesystem becomes full, performance degrades and users may not be able to save their work. Maintaining filesystems ensures that users can access the programs and data needed.

Objectives

This lesson shows you how to manage and repair filesystems. By the end of this lesson, you will be able to:

- Describe filesystem basics
- Use different filesystem types
- Mount and unmount a filesystem
- Repair a non-root filesystem
- Monitor filesystems
- Locate and manage user files
- Manage system log files and temporary directories

AV 5-2



5.1 Describing Filesystem Basics

Understanding what a filesystem is will help you manage user data efficiently. Filesystems must be handled properly to prevent data corruption. The capabilities of each filesystem type differ.

Although a user sees all data organized as a single hierarchical directory structure, that directory structure can consist of more than one filesystem.

A UNIX filesystem is a data structure that stores files. All files in the UNIX hierarchical directory structure are stored in filesystems. Filesystems hold all regular, special, and directory files, as well as the information needed to locate and access them.

A filesystem contains files much as a library bookshelf holds books. Like bookshelves, filesystems can fill up—and when a filesystem is full, it cannot hold any more data.

Filesystems can reside on hard disks, floppy disks, CD-ROMs, and in RAM. RAM filesystems are not permanent; they disappear along with their contents when you turn off the system.

Filesystems must be *mounted* to be available for use.

Filesystems are *unmounted* when they are no longer available for use.

5.2 SCO OpenServer Filesystem Types

Filesystem types differ in capabilities that can affect how they are used. The most frequently used SCO OpenServer filesystem types are:

<i>HTFS</i>	High Throughput Filesystem. The usual default for root and user filesystems. It provides enhanced performance and reliability.
<i>DTFS</i>	Desktop Filesystem. Provides data compression and a variable block size.
<i>EAFS</i>	Extended Acer Fast Filesystem. The default filesystem used for booting the operating system.
<i>DOS</i>	Lets users access DOS-formatted floppies and hard disk partitions using UNIX system commands and calls.
<i>HS, ISO9660, Rockridge</i>	Used for accessing CDROM devices: High Sierra, ISO-9660, and Rockridge format specifications.
<i>NUCFS</i>	NetWare® UNIX client filesystem. Allows mounting of volumes on NetWare servers.

The *S51K* filesystem is available for compatibility with other UNIX systems. The *HTFS* is the most feature-rich filesystem available with SCO OpenServer.

5.3 UnixWare 7 Filesystem Types

The most frequently used UnixWare 7 filesystem types are:

<i>vxfs</i>	The Veritas filesystem. This is the default type created for root and user filesystems. It provides enhanced performance and reliability.
<i>bfs</i> (boot filesystem)	Is a special-purpose filesystem that contains all the stand-alone programs, such as unix (located in <i>/stand</i>), and the text files necessary for boot procedures.
<i>dosfs</i> (DOS filesystem)	Lets users access DOS-formatted floppies and hard disk partitions using UNIX system commands and calls.
<i>cdfs</i> (CD-ROM filesystem)	Supports ISO-9660 and the High Sierra file system format specifications.
<i>memfs</i> (memory filesystem)	Is a high-performance, volatile memory filesystem type.
<i>nucfs</i> (NetWare® UNIX client filesystem)	Provides NetWare client file services on a UnixWare 7 platform.

The *s5*, *ufs*, *sfs* are standard UNIX filesystems that are available for compatibility with other UNIX systems. The *vxfs* is the most feature-rich filesystem available with UnixWare 7.

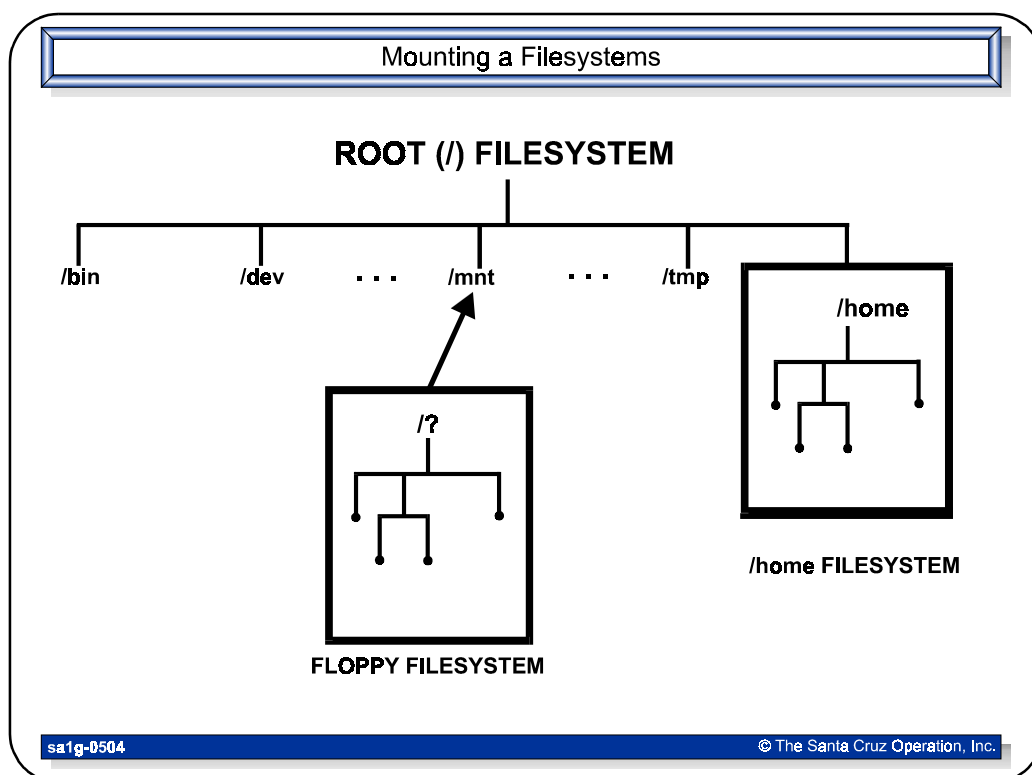
5.4 Advanced filesystem features

The SCO OpenServer *HTFS* and the UnixWare 7 *vxfs* filesystems provide enhancements that increase usability and make the system more viable for use in the commercial marketplace. They are particularly useful in environments that require high performance and availability and deal with large volumes of data.

The key features of these filesystems include:

- Large file and filesystem support. *HTFS* supports a maximum file size of 2GB and filesystem size of 1TB, while *vxfs* supports a maximum file and filesystem size of 1TB.
- Intent logging. The filesystem transactions are recorded in a log and then committed to disk. This increases system recovery speed with a very small performance penalty.
- Enhanced data integrity. If the system crashes or is shutdown improperly, only files that are currently being written to media can lose data.
- Default logical block size of 1KB (*vxfs* can support block sizes of 1024, 2048, 4096, and 8192 bytes).
- They are not bootable filesystems.

AV 5-3



5.5 Mounting and Unmounting a Filesystem

Filesystems allow easy expansion of the hierarchical directory structure because storage can be added to the system in a completely seamless manner. For example, if you make a floppy disk available to users, the filesystem on it appears within the existing directory structure. The user cannot distinguish between data on the floppy disk and data on the hard disk.

When the filesystem data is no longer needed, unmount the filesystem.

5.5.1 Describing filesystem information

Most permanent filesystems are mounted at boot time. Floppy and CD-ROM filesystems are mounted on occasion and only when the media is in the proper drive. When you mount a filesystem, you are attaching it to the current directory structure. You must have these two pieces of information:

- The device file that is associated with the filesystem.
- The existing directory on which to mount the filesystem. This is known as the *mount point*.

5.5.1.1 Filesystem device files

Each filesystem has an associated device file. Common filesystems with their respective device files include:

The <i>root</i> filesystem	<code>/dev/root</code>
Primary floppy disk	<code>/dev/fd0</code>
Alternate floppy names	SCO OpenServer: <code>/dev/fd0135ds18</code> UnixWare 7: <code>/dev/dsk/f03ht</code>
Primary CD-ROM	SCO OpenServer: <code>/dev/cd0</code> UnixWare 7: <code>/dev/cdrom/cdrom1</code>

5.5.1.2 Mount point

A mount point is an existing (normally empty) directory. When a filesystem is mounted, the directory entry for the mount point is redirected to the top level directory of the filesystem.

5.5.2 Filesystem creation

The **mkfs** command is used to create filesystems. Typically, this command is run automatically as the result of using other commands, for example, during the system installation and when adding additional hard disks. It can also be run from the command line to create filesystems on floppies and hard disks.

The basic use of the **mkfs** command is as follows:

mkfs <fs_type> <filesystem_device> <size>

Refer to the on-line documentation for additional information on the **mkfs** command

Exercise: Creating a floppy disk filesystem

Use **mkfs** to create a floppy filesystem:

Step 1. Insert a diskette in the floppy drive

Step 2. Create the floppy filesystem

SCO OpenServer: mkfs -f HTFS /dev/fd0 2880

UnixWare 7: mkfs -F vxfs /dev/fd0 2880

5.5.3 Adding the mount configuration

Before a floppy filesystem can be mounted using the **Filesystem Manager** you must first add the mount configuration.

Mount configuration information includes:

- If it is a local device or one that is installed on a remote system
- The device file
- The mount point. If the mount point directory contains any files, you are not able to access those files as long as a filesystem is mounted to that point. Specify the mount point as an absolute pathname. The mount point should always be an empty directory.
- Filesystem type
- Access mode
- When to mount it. Mount it now or during system startup.

For example, to add the mount configuration and mount a floppy filesystem:

Step 1. Enter **scoadmin filesystem** to start the **SCOadmin Filesystem Manager**.

Step 2. Select **Mount** ⇒ **Add Mount Configuration** ⇒ **Local**

Step 3. Select the device file. For example, to mount a 3.5 inch, 1.44MB floppy, enter:

SCO OpenServer: */dev/fd0*
UnixWare 7: */dev/dsk/f03ht*

Step 4. Insert the floppy disk into the floppy drive. The filesystem type is determined. Click on **Continue**.

Step 5. Specify the mount point, enter:

/floppy

Step 6. Enter a description of the floppy. This step is optional.

Step 7. Specify the access mode. The filesystem can be *read-only* or *read-write*.

Select **read-write**.

- Step 8. Specify when to mount the filesystem:
- **Now** mounts the filesystem immediately.
 - **At System Startup** mounts the filesystem when the system enters multiuser mode. This option is not recommended for a floppy disk filesystem.

Deselect **At System Startup**.

- Step 9. Click on **OK**. The filesystem is mounted.

To unmount the filesystem:

- Step 10. Select the filesystem

- Step 11. Click on **Mount** ⇒ **Unmount** ⇒ **Yes**.

- Step 12. Select **Host** ⇒ **Exit** to exit the **Filesystem Manager**.

Exercise: Mounting and unmounting a floppy disk filesystem

Following the steps in Section 5.5.3, “Adding the mount configuration,” on page 5-13 to add the mount configuration for a floppy disk filesystem. Mount and unmount the filesystem.

5.5.4 Mounting a filesystem

You can also use the **mount** command to mount a filesystem. For example to mount a filesystem floppy on the */mnt* directory:

```
SCO OpenServer: mount /dev/fd0 /mnt
UnixWare 7: mount -F vxfs /dev/fd0 /mnt
```

The *mount_point* directory does not need to exist immediately under the root (/) directory. Any subdirectory can be a mount point.

A CD-ROM filesystem should be mounted as *read-only*. For example, to mount a CD-ROM using the primary drive on the */mnt* directory, use this command:

```
SCO OpenServer: mount -r /dev/cd0 /mnt
UnixWare 7: mount -r -F cdfs /dev/cdrom/cdrom1 /mnt
```

Exercise: Mounting a floppy disk filesystem

Mount the floppy disk filesystem on the */mnt* directory using the **mount** command. Change to the */mnt* directory. Copy the */etc/default/tar* file to the filesystem floppy. List the contents. Do not umount the floppy.

5.5.5 Why mounting can fail

If a mount operation fails, check for these situations:

- Have you specified a non-existent device file or mount directory?
- If mounting a floppy disk or CD-ROM, is the media in the drive?
- Are you or someone else in the mount directory using a file? Use **fuser** to determine if anyone is using the directory:

```
fuser -u /dev/filesystem_device
```

This command normally displays the process ID of any process accessing that directory. The **-u** option also displays the user who owns the process. Use the **-ku** option to display the user and to send a kill signal to the process.

5.5.6 Unmounting a filesystem

To unmount a filesystem, users, including root, cannot be using the directory as their present working directory. Also, no one should be using any of the files in the filesystem.

Use **fuser -u** to determine if anyone is using the filesystem:

```
fuser -u /dev/filesystem_device
```

To unmount a filesystem from the command line, use the **umount** command:

```
umount /dev/filesystem_device
```

Exercise: Unmounting a floppy disk filesystem

Unmount the floppy disk filesystem, then mount it again on the */mnt* directory.

5.6 Repairing a Non-root Filesystem

If a filesystem becomes corrupt, you must repair it before you can use it safely. When a filesystem mounts during startup, the system usually checks and repairs it transparently without intervention before mounting it.

Check the filesystem manually under the following conditions:

- The filesystem is not mounted during startup.
- The filesystem is configured to be mounted during startup, but the system is not configured to check it.

5.6.1 Causes of filesystem corruption

When a filesystem becomes corrupted, either the actual data or the structures used to locate the data become damaged. Several situations can cause filesystem corruption:

- Hardware failure
- Program failures; for example, an error occurs that causes a program to fail, possibly resulting in data loss
- System crash

If filesystem corruption occurs, when you reboot the system, the root filesystem is automatically cleaned and repaired. Any user filesystems (if configured to do so) are also cleaned and repaired.

5.6.2 The *lost+found* directory

While repairing a filesystem, the system will attempt to recover any lost files and put them in the *lost+found* directory located in each filesystem. After a filesystem repair, check this directory for lost files.

5.6.3 Repairing a non-root filesystem manually

The **fsck** command is used to manually check and repair a filesystem. Before running **fsck**, you may need to determine the filesystem type using the **fstyp** command. The filesystem must be unmounted before running **fsck** on it.

One of the features of the *HTFS* and *vxfs* filesystems is to record pending filesystem updates in an *intent log*. This reduces the amount of time that **fsck** uses to repair the filesystem as only transactions in the intent log are checked. This is the default.

If a full filesystem check is required, run **fsck** with the **-ofull** option.

WARNING: Do not perform this procedure on the *root* filesystem.

For example, follow these steps to repair the floppy filesystem:

Step 1. Determine the filesystem type.

```
fstyp /dev/fd0
```

Step 2. Umount the filesystem.

```
umount /dev/fd0
```

Step 3. Run **fsck**.

```
SCO OpenServer: fsck -ofull /dev/fd0  
UnixWare 7: fsck -F vxfs -ofull /dev/fd0
```

Step 4. Mount the filesystem.

```
mount /dev/fd0 /mnt
```

Exercise: Repairing a floppy disk filesystem

Check and repair the floppy filesystem.

5.7 Monitoring Filesystems

When a filesystem comes close to being full, performance starts to suffer. If the filesystem actually fills up, users and programs cannot save any data to the filesystem. Never allow a filesystem to completely fill up.

In general, filesystem performance begins to degrade when filesystem free space drops below 15%. As the system administrator, it is your responsibility to keep track of filesystem usage.

5.7.1 Determining filesystem free space

The first statistic you should check is overall usage. These steps describe how to check the free space on all mounted filesystems:

Step 1. Enter **scoadmin filesystem** to start the **SCOadmin Filesystem Manager**

Step 2. Select **View** ⇒ **Disk Usage**.

Note the list displays the total size of each filesystem, the number of megabytes used, the number of megabytes free, and the percentage of the filesystem used.

Step 3. Select **Host** ⇒ **Exit** to exit the **Filesystem Manager**.

Use the **dfspace** command to check the free space on all filesystems. From the command line, enter:

dfspace

To check free space on a single filesystem, enter:

dfspace /dev/filesystem_name

Exercise: Determining free space on a filesystem

Use the **Filesystem Manager** to determine how much free space is in each of the filesystems.

AV 5-5.

Determining the Size of a Directory

```
# du -ks /etc
16352      /etc
```

SIZE IN 1K BLOCKS

-s: DISPLAY GRAND TOTAL
-k: DISPLAY SIZE IN 1K BLOCKS

sa1g-0506 © The Santa Cruz Operation, Inc.

5.7.2 Determining the size of a directory

If a filesystem is close to being full, you can determine if a particular directory seems inordinately large. This can help determine if there are files that you can remove to free disk space.

Use the **du** command to locate the directories using the most disk space:

```
du / | sort -nr | head -20
```

The **-n** option sorts by numeric value. The **-r** option reverses the order of the sort so that the directories using the most disk space are first. The **head -20** command displays the twenty largest directories

Exercise: Identifying the largest directories

Determine the twenty largest subdirectories on the system.

AV 5-6

Determining Disk Usage by a User

#quot	/dev/u
312348	john
86298	mary
5024	sue
⋮	

SIZE IN 512 BYTE BLOCKS

sa1g-0507

© The Santa Cruz Operation, Inc.

5.7.3 Determining disk usage by user

If a filesystem is close to full, you can check how much space each user is occupying. If a few are using an inordinate amount of space, you might request that they clean up their files.

Use the **quot** command to determine space used by each user in a given filesystem. It is reported in 512-byte blocks in SCO OpenServer and 1024-byte blocks in UnixWare 7.

For example, to determine the space used by each user in the *root* filesystem, enter:

SCO OpenServer: `quot /`
UnixWare 7: `quot -F vxfs /`

Exercise: Determining disk usage

Determine disk usage by user for the *root* filesystem. Remember to specify the filesystem type in UnixWare 7.

5.8 Locating and Managing User Files

Throughout normal operation, both users and programs save files to a filesystem. Most of these files are normal data files, but some can be *core* files. A core file can be created when a program fails. Maintain free disk space by periodically deleting these files.

Additionally, when you remove a user account, you need to decide what to do with files owned by that user. You can transfer ownership to another user or delete the files. Ignoring them can use valuable free space.

You can also track large files as a way to manage free space. If an individual file is growing, or if there is an increasing number of large files, adjust your filesystem organization to compensate for usage.

In any of these situations, the **find** command will help you manage user files.

5.8.1 Finding large files

Use the **find** command with the **-size +n** option (the size of the file in 512-byte blocks) to find files larger than a specific size:

```
find / -size +n
```

For example, to find all files larger than 1MB:

```
find / -size +2048
```

5.8.2 Finding files by owner

Use the **find** command with the **-user** option to find files owned by a specific user:

```
find / -user username
```

5.8.3 Finding and removing *core* files

A *core* file is created when a process aborts either from an error or from receiving a “quit” signal from a user ({CTRL} \ or **kill -3**). It contains a memory image of the process that was executing. In SCO OpenServer, the file is always named *core*. In UnixWare 7, the file is named *core.pid* where *pid* is the process id of the aborted process.

Use the **find** command to locate *core* files:

```
find / -name "core*"
```

WARNING! Do not automatically delete all files found. There may be important system files found (for example *core.h*). Instead, use the **find -ok** option if you would like to be prompted before deleting each file:

```
find / -name "core*" -ok rm {} \;
```

Exercise: Deleting *core* files

First, create a *core* file using the **sort** command:

```
sort /etc/termcap
```

While the command is running, press  \. Now use the **find** command to locate all *core* files on the system.

5.9 Managing System Log Files and Temporary Directories

The SCO OpenServer and UnixWare 7 systems maintain several log files that contain information about system usage. When new information is generated, the system appends it to the appropriate log file, preserving the previous contents of the file.

Because some log files can rapidly become quite large, check these files periodically, and if necessary, clear them by deleting their contents.

A temporary file contains data created as an intermediate step during execution of a program. This file can be left behind if a program encounters an error or a user stops it prematurely. In most cases, the user has no use for temporary files, and you can safely remove them.

5.9.1 Clearing key system logs

You might want to print or archive a log before clearing it. Do not delete a log file because the file ownership and permissions can change when it is recreated. Instead, enter the following:

> *log_filename*

Many of the system log files are stored in the */var/adm* directory. Some of the key log files are:

<i>/var/adm/messages</i>	System messages log (SCO OpenServer)
<i>/var/adm/log/osmlog</i>	System messages log (UnixWare 7)
<i>/var/adm/sulog</i>	Contains su attempts
<i>/var/adm/syslog</i>	Contains system startup information and errors
<i>/var/adm/wtmp</i>	Contains a history of all logins

5.9.2 Clearing key temporary directories

These directories include:

- *lost+found* (for each filesystem)
- */tmp*
- */var/tmp*

Check these directories periodically and remove older files.

5.10 Lesson Summary

Filesystems store data and programs used on the system. As an administrator, mount filesystems to make the data available. When a filesystem becomes corrupt, it must be cleaned before using it again. Monitoring and managing disk space assures that the system can continue to support your users' work. Monitoring can provide you with information that helps you decide when disk upgrades and additions are necessary.

Review questions

1. What must occur before a user can access the files on a filesystem?
2. Which filesystem type is the default used by UnixWare 7? SCO OpenServer?
3. List three situations that might cause a mount command to fail.
4. If the system is cleaning the *root* filesystem, in which directory will recovered files be stored?
5. Which command shows how much space each user is utilizing on a filesystem?
6. How would you find every file on the system that is larger than 3MB?
7. Why should you clear, rather than delete, a log file?

5.11 Lab: Working with a Floppy Disk Filesystem

This lab emphasizes the difference in a directory on which a filesystem is mounted.

5.11.1 Overview

If the floppy filesystem is still mounted, unmount it. Copy `/etc/default/passwd` to `/mnt` and write down the contents of `/mnt`. Mount a floppy disk filesystem on `/mnt`. Copy `/etc/default/login` to that filesystem and write down the contents of `/mnt`. Unmount the floppy filesystem and record the contents of `/mnt`. Compare the contents of `/mnt` when the floppy filesystem is mounted and when it is not.

5.11.2 Procedure

In this procedure, unmount the floppy disk filesystem (if it is mounted), copy a file to */mnt*, and record the original contents of that directory. Next, mount the floppy disk filesystem, copy a different file to */mnt*, and record the contents. Then, unmount the floppy disk filesystem, examine the contents of */mnt*, and compare the directory listings.

Step 1. Use the **umount** command to unmount the floppy disk:

```
umount /dev/fd0
```

Step 2. Copy */etc/default/passwd* to */mnt*:

```
cp /etc/default/passwd /mnt
```

Step 3. Obtain a directory listing of */mnt* and record the output.

```
ls -l /mnt
```

Step 4. Mount the floppy.

```
SCO OpenServer: mount /dev/fd0 /mnt  
UnixWare 7: mount -F vxfs /dev/fd0 /mnt
```

Step 5. Copy */etc/default/login* to the floppy disk filesystem:

```
cp /etc/default/login /mnt
```

Step 6. Obtain a directory listing of */mnt* and record the output:

```
ls -l /mnt
```

Step 7. Unmount the floppy.

```
umount /mnt
```

Step 8. Obtain a directory listing of */mnt* and record the output:

```
ls -l /mnt
```

Step 9. Compare the directory listings from Step 3, Step 6, and Step 8.

5.12 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

dfspace	Reports free disk space
du	Summarizes disk usage
find	Finds files
fsck	Checks and repairs filesystems
fstyp	Determines filesystem type
fuser	Identifies processes using a filesystem or directory
mkfs	Creates a filesystem
mount	Mounts a filesystem
quot	Summarizes filesystem ownership
scoadmin	Invokes SCOadmin applications (in this lesson, the Filesystem Manager)
umount	Unmounts a filesystem

Relevant files

<i>/etc/wtmp</i>	Accounts record file
<i>/var/adm/messages</i>	System messages log (SCO OpenServer)
<i>/var/adm/log/osmlog</i>	System messages log (UnixWare 7)
<i>/var/adm/sulog</i>	Contains su attempts
<i>/var/adm/syslog</i>	Contains system startup information and errors
<i>/var/adm/wtmp</i>	Contains a history of all logins

5.13 Answers to Exercises and Review Questions

This section contains answers to exercises and review questions presented in this lesson.

Review questions

1. Before a user can access the files on a filesystem, the filesystem must be mounted.
2. The default filesystem type used by UnixWare 7 is *vxfs*. The default filesystem type used by SCO OpenServer is *HTFS*.
3. Three situations that can cause a mount to fail are: specifying a non-existent device file or mount directory; mounting a floppy disk or CD-ROM without the media in the drive; or someone is either in the mount directory or using a file in the mount directory.
4. If the system is cleaning the *root* filesystem, recovered files are stored in */lost+found*.
5. The **quot** command shows how much space each user is using in the filesystem.
6. To find every file on the filesystem that is larger than 3MB, enter:

```
find / -size +6144
```

7. It is better to clear a log file rather than delete it because file permissions and ownership are also deleted.

6

Working with Disks and Tapes

AV 6-1

Working with Disks and Tapes

- Work with floppy disks
- Work with tape drives
- Transfer data to tape or disk using the **tar(1)** and **cpio(1)** commands

sa1g-0801 © The Santa Cruz Operation, Inc.

Lesson 6

Working with Disks and Tapes

Introduction

There are times when you will copy one or more files to another system. If the two systems are not networked together, copying the files to a floppy disk is the easiest way to do this.

You might also need to duplicate one or more directories on another system. If the amount of data is large, it can require many diskettes. Transferring the data by tape is the quickest way to accomplish this.

Objectives

This lesson covers all of the common ways to transfer data using both floppy disks and tapes. By the end of this lesson, you will be able to:

- Work with floppy disks
- Work with tape drives
- Transfer data to tape or disk using the **tar** and **cpio** commands

6.1 Working with Floppy Disks

Transfer data with floppy disks using these guidelines:

- Know the correct device file to use for a specific floppy disk
- When transferring data to a floppy disk, format the disk before using it

6.1.1 Identifying floppy device names

The standard UNIX data transfer commands use device files. Each floppy disk type is represented by a device file.

The device files used for formatting floppies and for transferring data with **tar** and **cpio** are sometimes referred to as *raw* device files. Raw device files bypass the kernel buffer cache. Some of the more commonly used floppy device names include:

	<u>SCO OpenServer</u>	<u>UnixWare 7</u>
3.5" 1.44MB	<code>/dev/rfd0</code> <code>/dev/rfd0135ds18</code>	<code>/dev/rfd0</code> <code>/dev/rdsk/f03ht</code>
3.5" 2.88MB	<code>/dev/rfd0135ds36</code>	<code>/dev/rdsk/f033t</code>

Refer to the on-line documentation for additional information on floppy device names.

6.1.2 Formatting a floppy disk

You may need to use the **format** command to format a floppy disk before using it.

For example, to format a 3.5" 1.44MB floppy disk, enter:

SCO OpenServer: `format /dev/rfd0135ds18`
UnixWare 7: `format /dev/rdsk/f03ht`

Exercise: Formatting a floppy disk

Format a floppy disk.

AV 6-2

Floppy Disk Attributes

```
alias='mdens1HIGH'
bdevice='/dev/dsk/f03ht'
bdevlist='/dev/dsk/f03h'
capacity='2844'
cdevice='/dev/rdisk/f03ht'
cdevlist='/dev/rdisk/f03h'
copy='true'
desc='3.5 inch 1.44 Mbyte (High Density)'
display='false'
erasescmd='/usr/sadm/sysadm/bin/floperase /dev/rdisk/f03ht'
fmtcmd='/usr/sbin/format -v /dev/rdisk/f03ht'
mdenslabel='dtmedia:293^1.44 Mbyte'
mkdtab='true'
mkfscmd='/sbin/mkfs -F s5 /dev/dsk/f03ht 2844:711 2 36'
mkvxfscmd='/sbin/mkfs -F vxfs -o ninode=711 /dev/dsk/f03ht
2844'
mountpt='/install'
removable='true'
type='mdens'
volume='diskette'
```

sa1g-0603

© The Santa Cruz Operation, Inc.

6.1.3 UnixWare 7 **getdev** and **devattr** commands

UnixWare 7 provides the **getdev** and **devattr** commands to help identify device names and provide additional information.

For example, if you don't know the floppy device name, use the **getdev** command to identify an alias for device name:

getdev | more

To further distinguish the floppy device, enter:

devattr -v diskette1

Important attributes to note before formatting a floppy include:

- The description of the device.
- The command to use to format the floppy device (**fmtcmd=**).

6.1.4 SCO OpenServer **dtype** command

SCO OpenServer provides the **dtype** command to help identify the type of data a floppy disk contains. For example:

dtype /dev/rfd0135ds18

This commands recognizes any of the following data types:

- UNIX filesystems
- DOS diskettes
- **tar** archives
- **cpio** archives

6.2 Working with Tape Drives

Follow these guidelines when transferring data with tapes:

- Know the correct device file to use for a specific tape drive
- If you receive a tape that includes data, you must know how the data was transferred to it. Label your tapes. The label should include, a description of the archive, the date of the archive, the size of the tape, the command the tape was archived with, and the number of tapes in the archive.

Computer tapes share some of the same limitations as audio tapes. The tape can develop slack, which can lead to data errors. Tapes need to be rewound before you can access data on them. Just as you can format floppy disks to remove data, you can also erase data from tapes.

6.2.1 Identifying tape device names

A tape drive (whether for a cartridge, 9-track, 8-millimeter, or DAT tape) is a character device. The most commonly used tape device names include:

SCO OpenServer:

<code>/dev/rct0</code>	Rewinding cartridge tape device. This is the most commonly used device file, often linked to <code>/dev/rStp0</code> .
<code>/dev/nrct0</code>	Non-rewinding cartridge tape device, often linked to <code>/dev/nrStp0</code> .
<code>/dev/xct0</code>	Control file used by the tape command, often linked to <code>/dev/xStp0</code> . Data cannot be transferred using this file. It is used to access the tape controller.

UnixWare 7:

<code>/dev/rmt/ctape1</code>	Non-retensioning rewinding device.
<code>/dev/rmt/ntape1</code>	Non-retensioning, non-rewinding device.
<code>/dev/rmt/rtape1</code>	Retensioning and rewinding device.
<code>/dev/rmt/nrtape1</code>	Retensioning non-rewinding device.
<code>/dev/rmt/utape1</code>	Unloads on the device close.

6.2.2 Working with tapes

You can retention, rewind, or erase tapes using the SCO OpenServer **tape** command or the UnixWare 7 **tapecntl** command. Depending upon the type of tape drive installed, there may be additional options available. To determine which options are available, enter:

SCO OpenServer: tape
UnixWare 7: tapecntl

The following are some of the most useful commands:

- Reset a tape. This option initializes the tape controller and positions the tape at the beginning of the tape.

SCO OpenServer: tape reset
UnixWare 7: tapecntl -r

- Rewind a tape. This option moves the tape to the beginning of the tape.

SCO OpenServer: tape rewind
UnixWare 7: tapecntl -w

- Erase a tape. This option erases the tape from beginning to end.

SCO OpenServer: tape erase
UnixWare 7: tapecntl -e

- Use a tape drive other than the default by specifying the appropriate device file at the end of the command. For example, to retention a tape using a cartridge tape drive, enter:

SCO OpenServer: tape reten /dev/rct0
UnixWare 7: tapecntl -t /dev/rmt/ctape1

Exercise: Using the tape or tapecntl command.

List the **tape** or **tapecntl** options that are available for use with your tape drive. Is the unload option available? If so, unload your tape from your tape drive.

6.3 Transferring Data to Tape or Disk

The two major data transfer utilities are **tar** and **cpio**. Both commands can copy files and directories. They also have advantages and limitations. This section addresses when and how to use these commands.

6.3.1 Absolute versus relative pathnames

You can specify files to transfer using either absolute or relative pathnames. The method you use to create an archive affects how you restore data from that archive.

6.3.1.1 Absolute pathnames

When using absolute pathnames, specify the file location relative to the root (/) directory.

The default behavior when restoring the file is to place it in the original location in the directory tree. For example, when creating the archive */etc/default/login*, you can create the archive from within any directory. You can also restore the file from any directory, and the file is placed in */etc/default*.

You can override absolute pathnames during a restore. For example, the **-A** option to **tar** will effectively strip off the leading forward slash (/) when performing data extraction. This is especially useful when data has been archived using absolute pathnames and extraction needs to occur in a different location.

6.3.1.2 Relative pathnames

When using relative pathnames, specify the file location relative to the current working directory.

The default behavior when restoring the file is to place it in a location relative to the current working directory. For example, your current directory is */etc/default*. To specify a relative pathname, use *./login* to designate the file. If you restore the file while in the */tmp* directory, the file *login* is restored to */tmp*.

6.3.2 Transferring symbolically linked files

The root filesystem contains files that are symbolically linked. When performing a directory listing using **ls -l**, a symbolically linked file displays as:

```
lrwxrwxrwx 1 root sys          8 Apr 23 15:02      unix -> /stand/unix
```

The contents of the file *unix* are actually stored in the */stand/unix* file.

When backing up an entire filesystem, all files are archived. However, consider the following when backing up a single directory containing symbolic links:

- If you back up the directory containing the link, the default behavior is to back up the link only, not the data
- With both **cpio** and **tar**, the **L** option backs up the actual data associated with the link. However, if you restore the file, the link is broken and the data now exists in that directory.

If you want to backup the actual data and preserve the links, you must back up both the directory containing the symbolic link and the directory containing the actual data. In this situation, do not use the **L** option.

AV 6-3

Defining the Default Device in `/etc/default/tar`

```
archive0=/dev/rdisk/f0q15dt      15      1200      n
archive1=/dev/rdisk/f1q15dt      15      1200      n
archive6=/dev/rdisk/f03ht        18      1440      n
.
.
.
archive8=/dev/rmt/c0s0            20       0         y
.
.
.

# The default device in the absence of a numeric or "-f device" argument
archive=/dev/rdisk/f0t            15      1200      n
```

sa1g-0605

© The Santa Cruz Operation, Inc.

6.3.3 The tar command

The **tar** command saves files on an archive and restores them from that archive. The advantages of **tar** include:

- It is easy to use.
- Archives can span multiple volumes (tapes or floppy diskettes)
- It can override absolute pathnames when restoring files. In other words, you can specify the final destination of a file even if the archive is created using absolute pathnames.

The limitations of **tar** include:

- It does not support filename expansion for extracting data.
- Data archives created with **tar** might be difficult to transfer to computer platforms other than SCO OpenServer or UnixWare 7 systems when archives span media.
- Compatibility problems with systems that have versions of **tar** that do not support archiving empty directories and device files. The UnixWare 7 version of **tar** does archive empty directories and device files, however the SCO OpenServer version does not.

The default device used by **tar** is specified in `/etc/default/tar`. This file also contains references to other devices that **tar** can access. For example, `archive8` points to the primary tape drive.

Exercise: Modifying the default device

Edit `/etc/default/tar` so that the default device is the tape drive in your system.

AV 6-4

tar(1) Options	
OPTION	DESCRIPTION
v	Verbose (display output)
L	Follow symbolic links
c	Create archive
r	Append to archive
t	Table of contents
x	Extract from archive
#	Archive device in <i>/etc/default/tar</i>

sa1g-0804 © The Santa Cruz Operation, Inc.

6.3.3.1 Creating a tar archive

Listed are several examples of creating archives using **tar**:

- Create an archive of */etc/default* using absolute pathnames. Store the archive on the default device:

```
tar cv /etc/default
```

- Create an archive of */etc/default* using relative pathnames. Store the archive on the primary floppy disk, identified as **archive6** in */etc/default/tar*:

```
cd /etc/default
```

```
tar cv6 *
```

- Create an archive of */etc/default*. Store the archive in */tmp/tarfile*:

```
tar cvf /tmp/tarfile /etc/default
```

- Create an archive of a symbolically linked directory.

```
tar cvL6 /directory_name
```

6.3.3.2 Viewing the contents of a tar archive

Use **tar tv** to view the contents of an archive:

```
tar tv
```

```
tar tv6
```

```
tar tvf /tmp/tarfile
```


6.3.3.3 Extracting data from a tar archive

Several examples of extracting data from a **tar** archive include:

- Restore the entire archive from the primary floppy disk to */tmp*:

```
cd /tmp
```

```
tar xv6
```

- Restore the *./login* file from the primary floppy disk to */usr/tmp*:

```
cd /usr/tmp
```

```
tar xv6 ./login
```

- Restore */etc/default/login* from */tmp/tarfile*. Override the absolute pathname and restore it to */tmp*:

```
cd /tmp
```

```
tar xvAf /tmp/tarfile etc/default/login
```

Note that **tar** extracts the entire directory structure under */tmp*.

AV 6-5

cpio(1) Options

Creating an Archive: (vocBL)

v	Verbose
o	Write archive out
c	ASCII header (for moving to different types of systems)
B	Blocking factor set to 5KB
L	Follow symbolic links

Extracting an Archive (vicdumB)

v	Verbose
i	Read archive in
c	ASCII header
d	Create directories as required
u	Copy unconditionally
m	Retain previous modification times
B	Blocking factor set to 5KB

sa1g-0806

© The Santa Cruz Operation, Inc.

6.3.4 The **cpio** command

The advantages of the **cpio** command include:

- It supports the use of filename expansion on extraction.
- It can span multiple volumes (tapes and floppy diskettes).
- It offers an ASCII header option, making it more portable between different platforms that offer **cpio**.
- It is the only standard data transfer utility that accurately archives entire filesystems.
- It can override absolute pathnames when restoring files. In other words, you can specify the final destination of a file even if the archive is created using absolute pathnames.
- Combined with the **find** command it offers a high degree of control and flexibility.

6.3.4.1 Creating a cpio archive

Use the **-o** option with **cpio** to create an archive. This command must receive a complete file list from *stdin* to back up the data.

The easiest way to generate a current file list is to use the **find** command. Two important options to the **find** command are:

- depth** Always true; causes descent of the directory hierarchy to be done so that all entries in a directory are acted on before the directory itself. This can be useful when **find** is used with **cpio** to transfer files that are contained in directories without write permission.
- mount** Always true; restricts the search to the filesystem containing the directory specified.

Listed are several examples of creating archives with **cpio**:

- Create an archive of the current directory. Store the archive on the primary floppy drive:

```
find . | cpio -vocB -O /dev/rfd0
```

Because the **c** option is used, this archive has a good chance of being restored onto a different type of UNIX system.

- Create an archive of */etc/default* using absolute pathnames. Store the archive on the primary floppy drive. Follow any symbolic links and get the actual data:

```
find /etc/default | cpio -vocBL -O /dev/rfd0
```

- Create an archive of the */stand* filesystem. Store the archive on tape:

```
cd /stand
```

```
SCO OpenServer: find . | cpio -vocB -O /dev/rct0  
UnixWare 7: find . | cpio -vocB -O /dev/rmt/ctape1
```

6.3.4.2 Viewing the contents of a cpio archive

Use this command to display a table of contents for a **cpio** archive:

```
cpio -victB -I /dev/device_name
```

For example, use this command to display the contents of a **cpio** archive on the primary tape drive:

```
SCO OpenServer: cpio -victB -I /dev/rct0  
UnixWare 7: cpio -victB -I /dev/rmt/ctape1
```

6.3.4.3 Extracting data from a cpio archive

Follow these examples when extracting files from a **cpio** archive:

- To extract all files from a tape archive, enter:

```
SCO OpenServer: cpio -vicdumB -I /dev/rct0  
UnixWare 7: cpio -vicdumB -I /dev/rmt/ctape1
```

- To extract a single file from a floppy archived with an absolute pathname, enter:

```
cpio -vicdumB -I /dev/rfd0 /etc/default/login
```

6.4 Accessing DOS data

SCO OpenServer and UnixWare 7 systems let you integrate DOS applications and files. Although DOS filesystems have a different structure than UNIX filesystems, you can mount them to the UNIX directory structure the same as any other UNIX filesystem. This provides two benefits:

- You can copy, move, rename, and delete files using UNIX commands. This eliminates the need to learn special commands.
- You can back up the files in the DOS filesystems using the same commands that are used to back up UNIX filesystems.

In addition, you can access data on DOS floppies directly, without mounting the floppy, by using DOS utilities available for SCO OpenServer and UnixWare 7.

6.4.1 Formatting a DOS floppy

Use the **dosformat** command to format a floppy for DOS access:

SCO OpenServer: **dosformat /dev/rfd0135ds18**
UnixWare 7: **dosformat /dev/rdisk/f03ht**

6.4.2 Using DOS utilities

SCO OpenServer provides DOS utilities that allow you to directly store and access data on a DOS floppy. For example:

Copy */etc/group* to a DOS floppy: **doscp /etc/group a:**

Get a directory listing of the DOS floppy: **dosdir a:**

Copy a file from the DOS floppy: **doscp a:group /tmp**

Similar DOS utilities are freely available for UnixWare 7 through the **mttools** package which can be downloaded from the SCO website or installed from a Skunkware CD. For example:

Copy */etc/group* to a DOS floppy: **mcopy /etc/group a:**

Get a directory listing of the DOS floppy: **mdir a:**

Copy a file from the DOS floppy: **mcopy a:group /tmp**

6.4.3 Mounting a DOS filesystem

DOS filesystems devices files include:

Floppy disks */dev/fd0*

Hard disk partitions **SCO OpenServer:** */dev/hd0d*
 UnixWare 7: */dev/dsk/c0b0t1d0p(1-4)*

Mounting and unmounting DOS filesystems are the same as for UNIX filesystems, always mount the block device. For example, to mount a DOS floppy disk on the */mnt* directory, enter:

SCO OpenServer: `mount -f DOS /dev/fd0 /mnt`

UnixWare 7: `mount -F dosfs /dev/fd0 /mnt`

You cannot execute DOS files and applications from a mounted DOS filesystem; to do this, you must boot DOS from the DOS partition (or install SCO Merge).

Exercise: Accessing a DOS filesystem

Mount a DOS floppy filesystem on */mnt*. Copy */etc/hosts* to the DOS floppy.
Do a directory listing of */mnt*. Unmount the floppy when you are finished.

6.5 Lesson Summary

UnixWare 7 systems provide several ways to back up data to diskette or tape. Each method has its advantages and limitations. The correct tool to use depends upon the job being performed and upon your level of comfort with each tool.

Review questions

1. What is the raw device name for a 3.5-inch, 1.44MB floppy disk drive if it is the second floppy disk drive in the system?
2. What type of tape device is `/dev/rmt/ctape1`?
3. Which **tar** command option backs up the actual data associated with a symbolically-linked file?
4. What are the benefits to mounting a DOS filesystem?

6.6 Lab: Creating an Archive and Extracting Selected Files

In this lab, you use the **cpio** and **tar** commands to create archives and extract selected files from them.

6.6.1 Overview

Create a relative backup of the `/var/spool/mail` directory to either tape or disk. Rename `/var/spool/mail/root`, then restore `/var/spool/mail/root` from the archive.

Create a backup of the `/dev` directory. Rename `/dev/fd0` and restore `/dev/fd0` from the archive.

6.6.2 Procedure

These steps describe how to create an archive and extract selected files:

Step 1. Change directory to */var/spool/mail*:

```
cd /var/spool/mail
```

Step 2. Use the **tar** command to create a relative backup of the directory:

- For a floppy disk, enter:

```
tar cv6 .
```

- For a cartridge tape, enter:

```
tar cv8 .
```

Step 3. Rename the */var/spool/mail/root* file:

```
mv root root.bak
```

Step 4. View the **tar** archive:

- For a floppy disk, enter:

```
tar tv6
```

- For a cartridge tape, enter:

```
tar tv8
```

Step 5. Use the **tar** command to extract */var/spool/mail/root*:

- From a floppy disk, enter:

```
tar xv6 ./root
```

- From a cartridge tape, enter:

```
tar xv8 ./root
```

Step 6. Change to the `/dev` directory.

```
cd /dev
```

Step 7. Create an archive of the `/dev` directory using **cpio**:

- For a floppy disk, enter:

```
SCO OpenServer:
```

```
find . | cpio -vocB -O /dev/rfd0135ds18
```

```
UnixWare 7:
```

```
find . | cpio -vocB -O /dev/rdisk/f03ht
```

- For a tape, enter:

```
SCO OpenServer:
```

```
find . | cpio -vocB -O /dev/rct0
```

```
UnixWare 7:
```

```
find . | cpio -vocB -O /dev/rmt/ctape1
```

Step 8. Rename `fd0`:

```
mv /dev/fd0 /dev/fd0.bak
```

Step 9. View the **cpio** archive:

- For a floppy disk, enter:

```
SCO OpenServer:
```

```
cpio -victB -I /dev/rfd0135ds18 | more
```

```
UnixWare 7:
```

```
cpio -victB -I /dev/rdisk/f03ht | more
```

- For a tape, enter:

```
SCO OpenServer:
```

```
cpio -victB -I /dev/rct0 | more
```

```
UnixWare 7:
```

```
cpio -victB -I /dev/rmt/ctape1 | more
```

Step 10. Using **cpio**, restore just the file `/dev/fd0`:

- From a floppy disk, enter:

```
SCO OpenServer:  
cpio -vicdumB -I /dev/rfd0135ds18 fd0
```

```
UnixWare 7:  
cpio -vicdumB -I /dev/rdisk/f03ht fd0
```

- From tape, enter:

```
SCO OpenServer:  
cpio -vicdumB -I /dev/rct0 fd0
```

```
UnixWare 7:  
cpio -vicdumB -I /dev/rmt/ctape1 fd0
```

6.7 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

cpio	Copies archives
doscp	Copy data to or from a DOS disk
dosdir	List DOS directories
dosformat	Format a DOS floppy
find	Locates files
format	Format a floppy disk
mdir	List DOS directories (Skunkware utility)
mcopy	Copy data to or from a DOS disk (Skunkware utility)
tape	Maintains tapes (SCO OpenServer)
tapectl	Maintains tapes (UnixWare 7)
tar	Copies archives

Relevant files

<i>/etc/default/tar</i>	tar device configuration file
-------------------------	--------------------------------------

6.8 Answers to Exercises and Review Questions

This section contains the answers to exercises and review questions presented in this lesson.

Review questions

1. The raw device name for a 3.5 inch, 1.44MB floppy disk drive, if it is the second floppy disk drive in the system, is:

SCO OpenServer: `/dev/rfd1135ds18`

UnixWare 7: `/dev/rdisk/f13ht`

2. The tape device, `/dev/rmt/ctape1` is the UnixWare 7 first tape device. It is a non-retensioning rewinding tape device.
3. The **L** option of the **tar** command backs up the actual data associated with a symbolically linked file.
4. Mounting a DOS filesystem has these benefits: you can copy, move, rename, and delete files using the same commands as for UNIX files. This eliminates the need to learn special commands.

7

Backup Administration

Lesson 7

Backup Administration

Introduction

The most expensive component of any system is the data. Unlike hardware, lost data is often irreplaceable. To protect this business critical information, routine backups are essential. Maintaining system backups and restoring lost data is an important and common task for system administrators.

Although you can transfer data with the **tar** and **cpio** utilities, you can also use more sophisticated backup packages available from SCO and others.

Objectives

This lesson describes several methods available to perform backups that enable the restoration of a non-root filesystem or critical files. By the end of this lesson, you will be able to:

- Organize and maintain filesystem backups
- Describe backup products available for SCO OpenServer and UnixWare 7

7.1 Organizing and Maintaining Filesystem Backups

System backups are a copy of one or more filesystems saved to a tape. From a system backup, you can restore:

- A single filesystem or multiple filesystems
- A single file or multiple files

7.1.1 Deciding when to back up filesystems

Backing up a filesystem when the fewest number of users are logged into the system results in these benefits:

- The backup does not affect system performance during peak times of system use
- Users are not modifying files

7.1.2 Storing backup media

After creating your backups, store them in a secure place. Consider archiving tapes off-site or in a fire safe for additional protection.

WARNING! Store your tapes in a secure location! If an unauthorized person can access your tapes, any data on those tapes is no longer secure.

For correct retrieval, label your tapes with this information:

- System name
- Filesystem name
- Backup date and who performed the backup
- Backup method, list which utility was used to create the backup
- Specify the sequence of volumes if the backup required multiple volumes. For example, specify “1 of 3” to identify the first tape of a backup that required three (3) tapes.
- Rotation date. The date after which a tape can be reused.

7.1.3 Maintaining backup logs

Maintaining a written backup log book for each computer also helps you to keep your backups organized. If you are using a backup utility that does not provide a backup log, for each backup of your system, record this information:

- Filesystem name
- Backup date
- Who performed the backup
- Number of volumes (tapes) required for the backup
- Any problems that occurred during the backup

Additional online reports may also be kept by individual backup utilities. For example, the following SCO ARCserve/Open logs can be printed and placed in your log book:

Activity Log	Contains comprehensive information about the operations performed by ARCserve/Open and provides an audit trail of ALL activity
Media Server Log	Provides an audit trail of all media activity

7.1.4 Using different backup levels

Some backup utilities allow you the option of only backing up files that have been created or modified since a previous backup.

For example in SCO OpenServer, the **SCOadmin Backup Manager** supports three backup levels:

Level 0	Backs up every file in the filesystem
Level 1	Backs up every file created or modified since the last level 0 backup
Level 2	Backs up every file created or modified since the last level 1 backup

In UnixWare 7, SCO ARCserve/Open provides similar functionality, as do most of the third party backup utilities.

Incremental backups often speed archive creation, but may complicate data restoration.

7.2 Describing backup products

There are many backup products available for SCO OpenServer and UnixWare 7, including:

- SCOadmin Backup Manager (SCO OpenServer only)
- SCO ARCserve/Open, SCO ARCserve/Open Lite
- ARCserveIT (UnixWare 7.1.x only)
- Third party backup products

7.2.1 SCO OpenServer SCOadmin Backup Manager

The SCO OpenServer **SCOadmin Backup Manager** is an easy to use graphical front-end to **cpio**.

Its advantages include:

- Allows creation of backup schedules
- Supports incremental backup levels
- Maintains backup lists for ease of restoring selected files

Among its disadvantages:

- Requires administrator interaction (does not support unattended backups)
- Options to **cpio** cannot be specified (for example it does not follow symbolic links)
- Backs up filesystems individually (not the entire system at once)
- No byte-level backup verification (**cpio** header read only)

Note that you can perform unattended backups of an SCO OpenServer using the **cron** and **cpio** commands discussed earlier.

In addition to **cpio**, the SCO OpenServer **cbackup** command will allow you to perform incremental backups from the command line or from a cron job. For example, to perform a level 0 backup of the *root* filesystem using a 250MB tape:

```
/usr/lib/sysadmin/cbackup 0 250000 /dev/rct0 /dev/root
```

7.2.2 SCO ARCserve/Open Lite

SCO ARCserve/Open Lite is freely available for SCO OpenServer and some UnixWare 7 versions/editions. It provides an easy to use, comprehensive backup and restore of your system. It does not support all of the advanced features of ARCserve/Open and ARCserve IT (for example, it will not backup and restore networked clients).

7.2.3 SCO ARCserve/Open

The full SCO ARCserve/Open product is available as an extra cost add-on for SCO OpenServer. It is bundled for free in some UnixWare 7 versions/editions.

ARCserve/Open provides advanced backup and restore services, including the following features:

- Backup and restore of network clients, including
 - UNIX
 - Apple Macintosh
 - IBM OS/2
 - Novell NetWare
 - Microsoft Windows
- Support for fully unattended backups (auto pilot feature)
- Byte level data verification of backup
- Full system backup, as well as individual file and filesystem backup
- Filtering options for specifying files to include/exclude during backup and restore
- Extensive reporting and logging capabilities
- Full featured command line interface



7.2.4 ARCserveIT

ARCserveIT is a fully featured, high performance data management tool used to back up and restore data on servers and all workstations attached to a network. ARCserveIT is licensed by default in all UnixWare 7.1.0 editions except the Base and Free UNIX Editions. The ability to backup up to 250 machines is included in each license.

ARCserveIT uses the client/server architecture that functions within a group of ARCserveIT servers. The components include:

Web browser	Provides access to all of ARCserveIT operations using Java Applets
ARCserveIT manager	Functions as an information bridge between the browser and back-end services
Command line	Provides access to all ARCserveIT operations from the command line
Client Agent for UNIX	Enables UNIX workstations to be browsed and handles the transfer of data. The client module must be present on all systems that you want to backup or restore. Client Agents are included for: ● UnixWare 7 and SCO UnixWare® 2.1x ● SCO OpenServer™ Release 5 ● Microsoft Windows 3.x, Windows 95/98, and Windows NT workstations

New features of ARCserveIT include:

- Java/HTML Graphical User Interface (GUI) using Netscape Communicator 4.09 on UnixWare 7 and SCO OpenServer Release 5 and Netscape Communicator 4.07 on Windows.
- Multiple SCSI card support. Automatically detects tape drives on all SCSI cards.
- No device locking. Devices can be used by other backup programs, i.e., **tar** and **cpio**.
- Other backup methods. You can specify to use either **tar**, **cpio**, or ARCserveIT as the method for backing up files. Note that ARCserveIT cannot restore tapes that are created using the **tar** and **cpio** command-line interfaces.

7.2.5 Third party backup products

There are numerous other backup products available for SCO OpenServer and UnixWare 7 from third party vendors. They are typically easy to use, and provide advanced features such as unattended backup, byte level data verification, and simplified disaster recovery.

Some of these popular packages include:

- BackupEDGE from Microlite Corporation
- BRU from Enhanced Software Technologies
- CTAR from UniTrends Software Corporation
- Lone-Tar from Lone Star Software Corporation

7.3 Lesson Summary

Creating system backups is the one of the best ways to protect your system in case of either hardware failure or accidental loss of data. There are numerous SCO OpenServer and UnixWare 7 backup tools available that allow you to quickly backup and restore either entire filesystems or selected files.

Review questions

1. Why should you store backup tapes off-site from the location of the system?
2. How can you verify that a backup is created free of errors?

7.4 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

cbackup	Performs incremental backups (SCO OpenServer)
scoadmin	Invokes SCOadmin applications (in this lesson, the SCOadmin Backup Manager)

7.5 Answers to Review Questions

This section contains the answers to review questions presented in this lesson.

1. You should store backup tapes off-site from the location of the system to prevent complete data loss in case of on-site property destruction caused by fire or other disasters.
2. To verify that a backup executed free of errors, some backup utilities will allow you to perform byte level verification.

8

Managing Printers

AV 8-1

The slide is titled "Managing Printers" in a blue-bordered box at the top. Below the title is a bulleted list of six tasks. At the bottom, there is a blue footer bar containing the text "sa1g-0801" on the left and "© The Santa Cruz Operation, Inc." on the right.

Managing Printers

- Manage the print service
- Accept and reject print jobs
- Disable and enable printers
- Manage print jobs
- View and change printer settings
- Troubleshoot printer problems

sa1g-0801 © The Santa Cruz Operation, Inc.

Lesson 8

Managing Printers

Introduction

Printing is one of the most common functions that users perform. Maintaining printers, print services, and print queues lets your users print out reports and other items required for their jobs.

This lesson describes how to manage print queues when bringing a printer down for maintenance. It also shows you how to manage user print requests and how to configure printers for specific functions.

Objectives

This lesson addresses the tasks required to manage printing services. By the end of this lesson, you will be able to:

- Manage the print service
- Accept and reject print jobs
- Disable and enable printers
- Manage print jobs
- View and change printer settings
- Troubleshoot printer problems

8.1 Managing the Print Service

Print services are controlled through the **SCOadmin Printer Setup Manager** and the **Print Job Manager**.

8.1.1 Describing the print service

When a user prints a file, a daemon process called the printer scheduler, **lpsched**, sorts the print requests and puts them into the designated printer queue. As soon as the designated printer is ready for that print job, the scheduler runs an interface program to print the file.

The print service is responsible for performing these tasks:

- Receives files to be printed
- Schedules the work of one or more printers
- Starts the programs that interface with specific printers
- Tracks the status of print jobs
- Alerts you to printer problems
- Issues error messages when problems occur

Use the **lpstat** command to view the status of the various printers, enter:

```
lpstat -t | more
```

Exercise: Determining printer status

View the status of the printers on your system.

8.1.2 Stopping the print service

The print service starts as soon as the system enters a multiuser state. Under normal operation, you should never have to stop and start the print service manually. However, sometimes you must stop the print service to clear up a problem.

Follow these steps using the **SCOadmin Printer Setup Manager** to stop the print service.

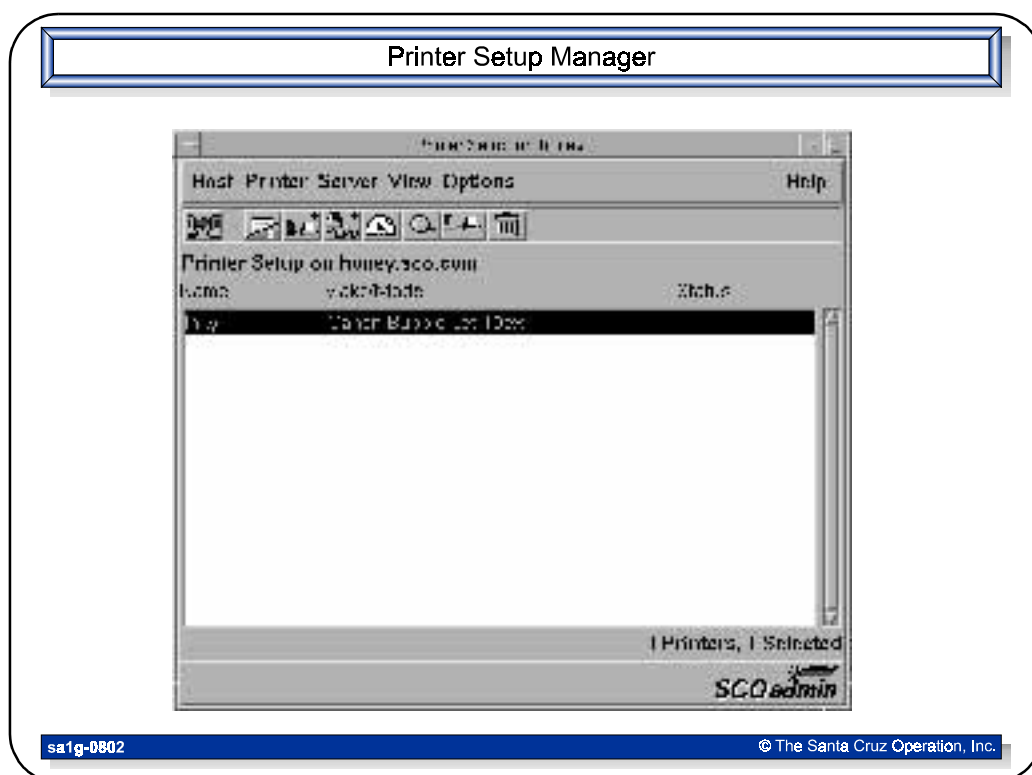
- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. **SCO OpenServer:**
Select **System** ⇒ **Print Services**.
Deselect **Local Print Service Enabled**.

UnixWare 7:
Select **Server** ⇒ **Halt**.
- Step 3. Click on **OK** to stop the print services.
- Step 4. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

You can also use the **lpshut** command to stop the print service, enter:

/usr/lib/lpshut

AV 8-2



8.1.3 Starting the print service

Follow these steps using the **SCOadmin Printer Setup Manager** to start the print service:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. **SCO OpenServer:**
Select **System** ⇒ **Print Services**.
Select **Local Print Service Enabled**.
- UnixWare 7:**
Select **Server** ⇒ **Start**.
- Step 3. Click on **OK** to start the print services.
- Step 4. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

You can also use the **lp sched** command to start the print service, enter:

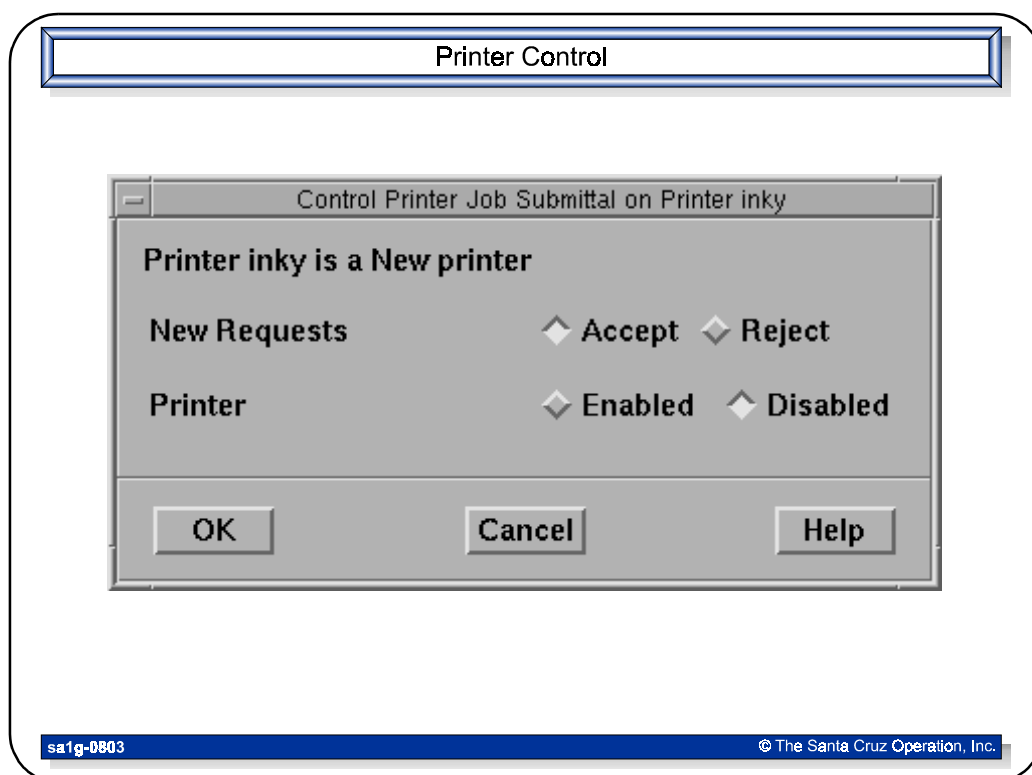
/usr/lib/lpsched

Exercise: Stopping and starting the print service

Shut down the print service and print a file. What is the error message? Run **lpstat**. What is the meaning of the message?

Restart the print service. Run **lpstat** to verify that the print service is running.

AV 8-3



8.2 Accepting and Rejecting Print Jobs

Sometimes a printer needs major maintenance, possibly requiring a trip to the repair shop. Maybe you are waiting for a toner cartridge that is on back-order. In these situations, set the print queue to reject print jobs so that users know to send requests to another printer.

8.2.1 Rejecting print jobs

Use the **SCOadmin Printer Setup Manager** to reject print requests:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. Select the printer.
- Step 3. **SCO OpenServer:**
Select **Settings** ⇒ **Control**.
Deselect **Accept New Local Jobs**.

UnixWare 7:
Select **Printer** ⇒ **Control**.
Select **Reject**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

You can also use the **reject** command to stop a printer from accepting print jobs:

reject printername

To add a message stating why the printer is down or to redirect your users to another printer:

reject -r"reason" printername

To display the "reason," enter:

lpstat -a

8.2.2 Accepting print jobs

Use the **SCOadmin Printer Setup Manager** to accept printer requests:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. Select the printer.
- Step 3. **SCO OpenServer:**
Select **Settings** ⇒ **Control**.
Deselect **Accept New Local Jobs**.
- UnixWare 7:**
Select **Printer** ⇒ **Control**.
Select **Accept**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

You can also use the **accept** command to allow a printer to begin accepting print jobs, enter:

accept printername

Exercise: Rejecting print jobs

Reject print jobs, and provide a message that informs users to redirect print requests to *laser2*.

reject -r"laser1 is down, use laser2" laser1

Run **lpstat**. What does it report? Try to send a print job to *laser1*. What message displays?

Accept print jobs. Run **lpstat** and note the difference. Try to print a file to *laser1*.

8.3 Disabling and Enabling Printers

Printers occasionally need routine checks and maintenance, such as replacing toner cartridges or removing jammed paper. If the maintenance requires only a short disruption of printing, disable the printer. A disabled printer continues to accept print jobs in its queue, but does not print them until you re-enable it.

If you disable the printer while it is in the middle of a print job, the printer restarts the job from the beginning after you enable it.

8.3.1 Disabling a printer

Use the **SCOadmin Printer Setup Manager** to disable printers:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. Select the printer.
- Step 3. **SCO OpenServer:**
Select **Settings** ⇒ **Control**.
Deselect **Enable Printing**.

UnixWare 7:
Select **Printer** ⇒ **Control**.
Select **Disable**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

You can also use the **disable** command to disable a printer:

disable printername

Use the **-r** option to disable the printer so that a reason appears in an **lpstat** report, enter:

disable -r"reason" printername

8.3.2 Enabling a printer

Use the **SCOadmin Printer Setup Manager** to enable printers:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. Select the printer.
- Step 3. **SCO OpenServer:**
Select **Settings** ⇒ **Control**.
Select **Enable Printing**.
- UnixWare 7:**
Select **Printer** ⇒ **Control**.
Select **Enable**.
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

You can also use **enable** to enable the printer:

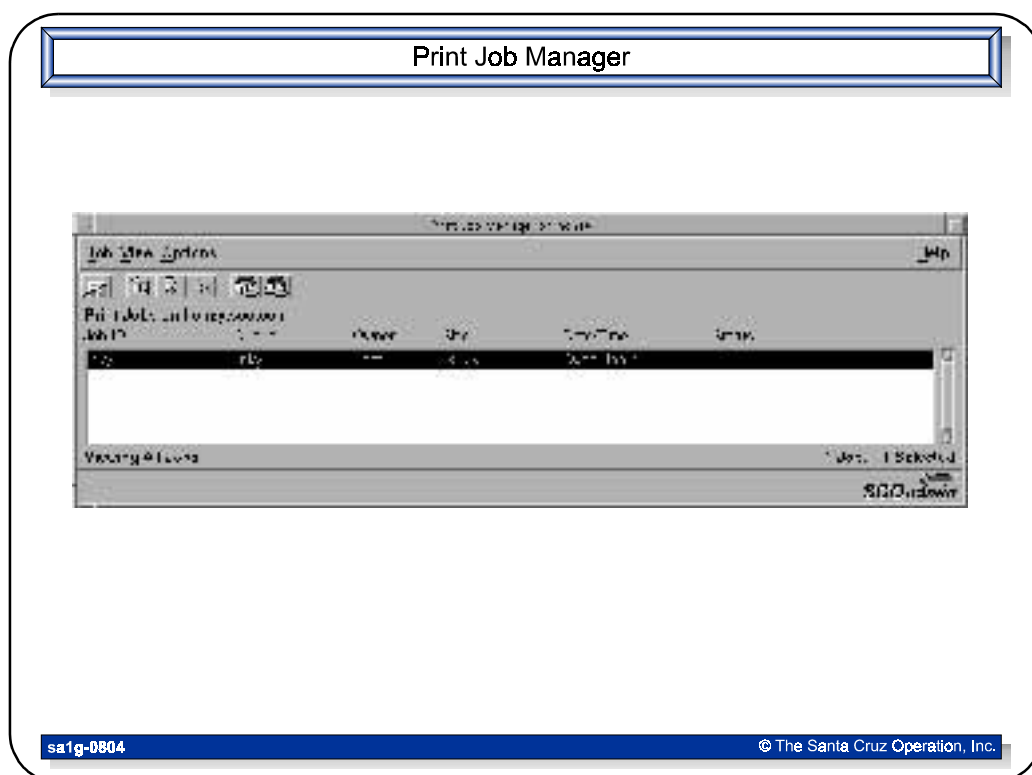
enable printername

Exercise: Disabling a printer

Disable *laser1* and try to print a file to *laser1*. What happens? What does **lpstat -a** report? What does **lpstat -t** report?

Enable *laser1* and run **lpstat** again. What happens to the print job?

AV 8-4



8.4 Managing Print Jobs

What can you do if one of these situations occur?

- A printer breaks down, but there are several jobs already waiting in its queue.
- Someone sends a very large job to the printer several times by accident.
- An extremely important and urgent report is sitting in a queue behind many time-consuming and less important print jobs.

The **Print Job Manager** lets you manipulate individual print requests and groups of print requests to keep the *print queues* running smoothly. A print queue is a list of all the print jobs that belong to individual users. By default, the **Print Job Manager** lists *all* print jobs.

You can also view print jobs by a single *user* or *queue*.

8.4.1 Viewing print jobs

Use the **SCOadmin Print Job Manager** to view print jobs.

Step 1. From a terminal window, enter **scoadmin print job**.

A list of print jobs that are in the printer queues appear in the order in which they were submitted.

Step 2. Select **View** ⇒ **By User(s)**. Select from the list of available users.

Step 3. Select **View** ⇒ **By Queue(s)**. Select from the list of available queues (printers).

Step 4. Select **View** ⇒ **All Jobs** to view all print jobs in all queues.

Step 5. Select **Job** ⇒ **Exit** to exit the **Printer Job Manager**.

You can also use the **lpstat** command to view print jobs:

- To view print jobs by queue, list the queues:

```
lpstat laser1 laser2
```

- To list print jobs by user, use the **-u** option:

```
lpstat -u user1,user2
```

Exercise: Viewing print jobs

Disable both *laser1* and *laser2*. As the **root** account, send several print jobs to each printer. Log in as an ordinary account and send several print jobs to each printer.

Now view the print requests. Sort the jobs by queue, then by user.

8.4.2 Deleting print jobs

Use the **Print Job Manager** to delete print jobs:

Step 1. From a terminal window, enter **scoadmin print job**.

Step 2. Select the print job(s) to delete.

To delete all jobs:

Step 3. Select **Select All Jobs**.

Step 4. Select **Job** ⇒ **Delete**.

Step 5. Select **Job** ⇒ **Exit** to exit the **Print Job Manager**.

Note that each print job has a unique job ID that consists of two parts:

- Name of the printer queue
- Sequence number

You can also use the **cancel** command to cancel print jobs:

```
cancel job_id1 job_id2 ...
```

Exercise: Canceling print jobs

Cancel one print job from each queue. Enable both printers, and monitor the print queues as the printer service processes the requests.

8.4.3 Holding print jobs

You can suspend the processing of a large print job temporarily until several smaller jobs have processed. In this case you can *hold* a print job. The print service continues to process other jobs in the queue until you resume the job.

Use the **Print Job Manager** to hold a print job:

- Step 1. From a terminal window, enter **scoadmin print job**.
- Step 2. Select the print job(s) to place on hold. To hold all print jobs:
Select **Select All Jobs**.
- Step 3. Select **Job** ⇒ **Hold**.
- Step 4. Select **Job** ⇒ **Exit** to exit the **Print Job Manager**.

You can also use the **lp** command to hold print jobs:

```
lp -i job_id -H hold
```

8.4.3.1 Resuming print jobs

Use the **Print Job Manager** to resume a print job:

- Step 1. From a terminal window, enter **scoadmin print job**.
- Step 2. Select the print job(s) to resume. To resume all jobs:
Select **Select All Jobs**.
- Step 3. Select **Job** ⇒ **Resume**.
- Step 4. Select **Job** ⇒ **Exit** to exit the **Print Job Manager**.

You can also use the **lp** command to resume print jobs:

```
lp -i job_id -H resume
```

Exercise: Holding and resuming print jobs

Disable *laser1* so that print jobs accumulate. Send several print jobs to *laser1*.

Place the second print job in the queue on hold. Enable *laser1* and monitor its print queue.

Resume the printing of the held print job. Monitor the print queue.

8.4.4 Moving print jobs

You can move or *promote* a print job to a higher level in the queue. You can also move print jobs within a single queue or between different queues.

8.4.4.1 Promoting print jobs

Promoting a print job places it higher in the print queue. Promoting a job does not necessarily mean that the job prints first. If there is a job currently printing, the job moves to next in line in the queue, unless you place the printing job on hold.

Use the **SCOadmin Print Job Manager** to promote a print job:

- Step 1. From a terminal window, enter **scoadmin print job**.
- Step 2. Select the print job(s), then select **Job** ⇒ **Promote**. The job or jobs move to the top of the queue.
- Step 3. Select **Job** ⇒ **Exit** to exit the **Print Job Manager**.

You can also use the **lp** command to promote a print job:

lp -i *job_id* -H immediate

Exercise: Promoting a print job

Disable *laser1* and send several print requests to the queue. Promote the last item in the queue, then promote the second-to-last. What is the print order now?

Promote the second item again to place it first in the list. Enable *laser1* and monitor the queue.

8.4.4.2 Transferring print jobs between queues

Use the **SCOadmin Print Job Manager** to transfer print jobs between queues:

- Step 1. From a terminal window, enter **scoadmin print job**.
- Step 2. Select the job(s) to transfer. To transfer all jobs:
Select **Select All Jobs**.
- Step 3. Select **Job** ⇒ **Transfer**.
- Step 4. Select the destination queue, then click on **OK**.
- Step 5. Click on **OK** again when prompted to confirm the destination queue.
- Step 6. Select **Job** ⇒ **Exit** to exit the **Print Job Manager**.

You can also use the **lp** command with the **-d** option to transfer print jobs between queues:

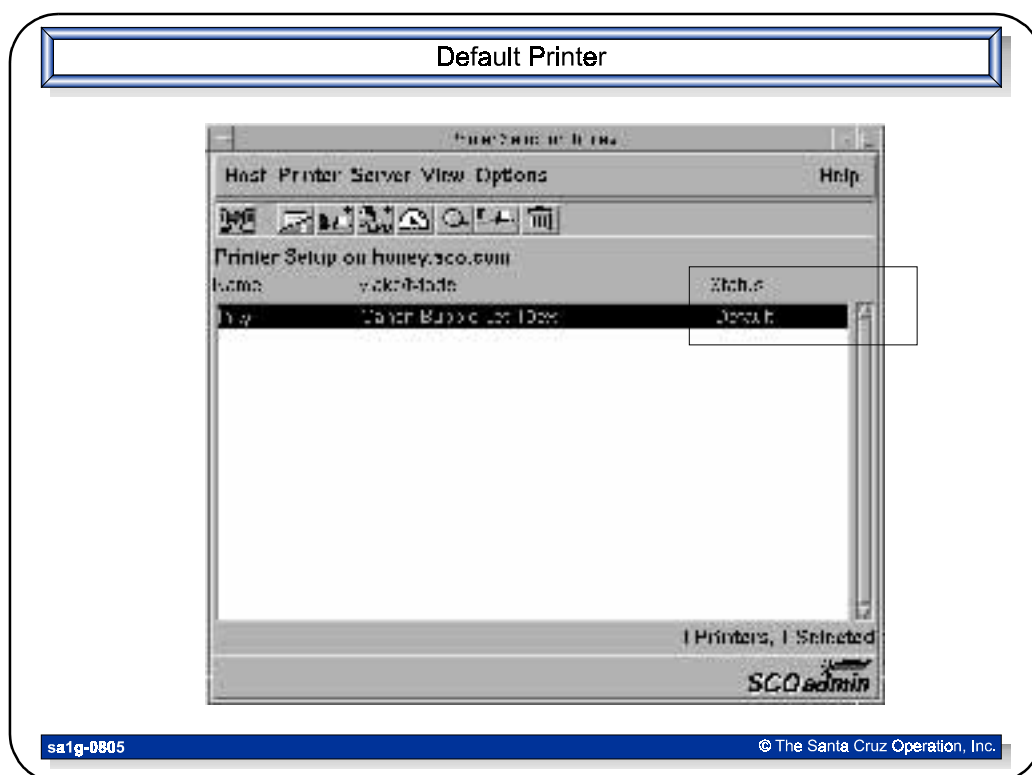
lp -i -d job_id -dprint_queue

Exercise: Transferring print jobs

Disable *laser1* and *laser2*. Send several print jobs to *laser1*. Transfer the last job in the queue to *laser2*.

Does the job ID change when you transfer a job?

AV 8-5



8.5 Viewing and Changing Printer Settings

You can designate a printer to be the system default printer, set a default printer for an individual user, control user access to a printer, and determine whether or not a printer produces banner pages.

8.5.1 Setting the default printer

When a user does not specify a destination, print jobs go to the system default printer. Setting the default simplifies the environment for the user.

Use the **SCOadmin Printer Setup Manager** to set up a printer as the default printer:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. Select a printer, then select **Printer** ⇒ **Set To Default**.
- Step 3. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

8.5.2 Setting a default printer for a user

You can specify default printers for individual users, overriding the system default.

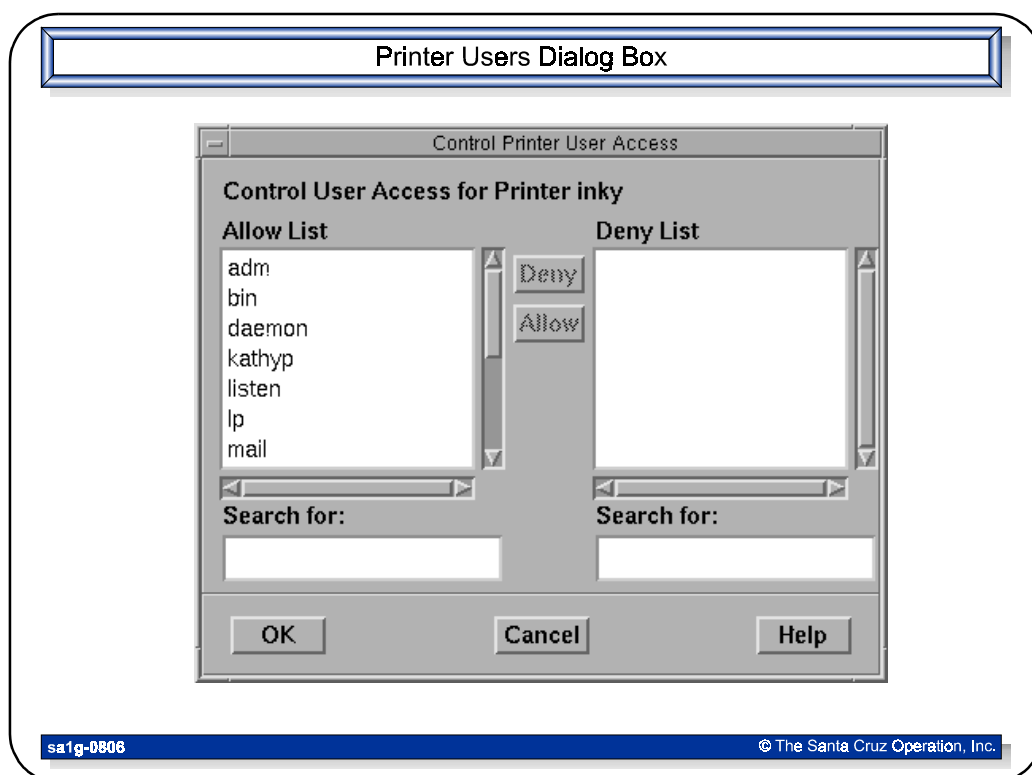
For Bourne and Korn shell users, edit the user's *.profile* and add these two lines:

```
LPDEST=printername
export LPDEST
```

For C shell users, edit the user's *.cshrc* and add this line:

```
setenv LPDEST printername
```

AV 8-6



8.5.3 Controlling access to a printer

There are situations in which you might want to restrict who can send jobs to a specific printer. For example:

- A department needs to print a large job as quickly as possible.
- You want to minimize wear on a high-end printer.
- You have a printer stocked with blank company checks.

Use the **SCOadmin Printer Setup Manager** to restrict access to a printer:

Step 1. From a terminal window, enter **scoadmin printer**.

Step 2. Select the printer.

Step 3. **SCO OpenServer:**
Select **Settings** ⇒ **Advanced** ⇒ **Users**.

UnixWare 7:
Select **Printer** ⇒ **Set User Access**.

The printer user dialog box appears with an **Allow** and a **Deny** list.

Step 4. Allow or deny access to the printer:

- If a user is on the **Allow** list, and you want to deny that user access to the printer, select the user name, then select **Deny**. The user name moves to the **Deny** list.
- If a user is on the **Deny** list, and you want to allow that user to access the printer, select the user name, then select **Allow**. The user name moves to the **Allow** list.

Step 5. Click on **OK**.

Step 6. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.



Exercise: Controlling access to a printer

Set up the *laser2* printer to deny access to your student account. Log in using your student account, then try to print */etc/default/login* to *laser2*. Interpret the error message.

8.5.4 Preventing banner pages

If you have a printer dedicated to printing special forms, such as checks, you can disable banner pages so that the printer does not print a banner across the form.

Use the **SCOadmin Printer Setup Manager** to disable printing a banner page in SCO OpenServer or to let users send the **nobanner** option when submitting print requests in UnixWare 7:

- Step 1. From a terminal window, enter **scoadmin printer**.
- Step 2. Select the printer.
- Step 3. **SCO OpenServer:**
Select **Settings** ⇒ **Advanced** ⇒ **Banners**.
Set Number of banner pages to **0**

UnixWare 7:
Select **Printer** ⇒ **Properties**.
Click on **Yes** at “Allow banner page override.”
- Step 4. Click on **OK**.
- Step 5. Select **Host** ⇒ **Exit** to exit the **Printer Setup Manager**.

In SCO OpenServer, the above steps will disable banner page printing.

In UnixWare 7, the above steps will allow users to submit a print request using the **nobanner** option. For example:

```
lp -o nobanner /etc/default/login
```

8.6 Troubleshooting Printer Problems

If a printer stops printing, try these troubleshooting tips:

- Check your e-mail. Error messages that occur during printing are often mailed to the user who made the request.
- Use **lpstat -t** to view the status of the printing system. Verify that the printer scheduler is running and that the printer is enabled and accepting requests.
- Make sure the printer is online.
- Check the power and cables to the printer.
- Verify that the printer hardware is working by checking the printer documentation for a self-test. Switch the printer off and on again.
- Shut down the print service. Use **ps** to verify that **lpsched** is not running. Restart the print service.

If none of the above work, shut down the system and restart it.

Refer to **SCOhelp** for a list of common problems and suggested solutions.

Exercise: Online help troubleshooting suggestions

Use **SCOhelp** to find suggestions if the printer output is illegible.

8.7 Lesson Summary

The print service lets you control how users access printers. You can temporarily disable a printer for quick maintenance, or you can reject all print jobs if you require a specific printer to be unavailable for a few days.

The print service also lets you control individual print requests. Print jobs can be redirected to different printers, canceled, or moved ahead in the queue.

In addition, the print service lets you control who can access a printer, and how specific printers behave. You can also designate a default printer—system-wide or for individual users.

Review questions

1. What tasks does the print service perform?
2. When should you set up a printer to reject print requests?
3. When should you disable a printer?
4. What is the difference between transferring a print job and promoting it?
5. How would you limit user access to a specific printer?
6. List three things to check when a printer stops printing.

8.8 Lab: Managing Printers

In this lab, *laser1* needs to be shut down for repairs. Reject any new print requests for that printer and redirect any jobs in its queue to *laser2*.

8.8.1 Overview

Disable *laser1* and *laser2* so that the print requests accumulate. Send several print requests to both *laser1* and *laser2*. Set *laser1* to reject requests, then move the jobs in *laser1*'s queue to *laser2*'s queue.

8.8.2 Procedure

These steps describe how to shut down *laser1* for repairs, reject any new print requests for that printer, and send new print requests to *laser2*.

Step 1. Disable *laser1* and *laser2*:

```
disable laser1  
disable laser2
```

Step 2. Send several print requests to both printers. Some examples are listed below:

```
lp -dlaser1 /etc/default/login  
lp -dlaser2 /etc/default/tar
```

Step 3. Reject any new print jobs from going to *laser1*:

```
reject laser1
```

Step 4. Try to send a print job to *laser1*:

```
lp -dlaser1 /etc/passwd
```

Step 5. Start the **SCOadmin Print Job Manager**.

Step 6. Select **View** ⇒ **By Queue(s)** to list all print jobs on *laser1*.

Step 7. Select all print jobs currently on *laser1*, then select **Job** ⇒ **Transfer**. Transfer the jobs to *laser2*. Click on **OK**.

Step 8. Click on **OK** again when prompted to transfer the jobs to *laser2*.

Step 9. Select **Job** ⇒ **Exit** to exit the **Print Job Manager**.

Step 10. Enable *laser2*:

```
enable laser2
```

8.9 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

accept	Allows print requests to a printer or class of printers
cancel	Cancels print requests
disable	Turns off a printer or class of printers
enable	Turns on a printer or class of printers
lp	Sends requests to a printer
lp sched	Starts the print service
lpshut	Stops the print service
lpstat	Prints information about the current status of the print service
lpusers	Sets printing queue priorities
reject	Prevents print requests to a printer or class of printers
scoadmin	Invokes SCOadmin applications (in this lesson, the Print Job Manager and the Printer Setup Manager)

Relevant files

<i>\$HOME/.cshrc</i>	User's C shell login file where a default printer can be set
<i>\$HOME/.profile</i>	User's Bourne or Korn shell login file where a default printer can be set

8.10 Answers to Review Questions

Review questions

1. The print service performs these tasks: receives files for printing, schedules the work on one or more printers, starts the interface with specific printers, tracks the status of print jobs, alerts you to printer problems, and issues error messages when problems occur.
2. You should set up a printer to reject print requests when the printer is going to be unavailable for more than one to two hours.
3. You should disable a printer when it is going to be unavailable for only a brief amount of time.
4. Transferring a print job sends it to a different print queue. Promoting a print job places it at the top of the current print queue.
5. You can edit the allow and deny user lists found in the **Control Printer User Access** dialog box in the **Printer Setup Manager**.
6. Check any of these when a printer stops working: check your mail—error messages that occur during printing are often mailed to the user who made the request; use `lpstat -t` to view the status of the printing system—verify that the printer schedule is running and that the printer is enabled and accepting requests; check the power and cables to the printer; verify that the printer hardware is working by checking the printer documentation for a self-test—switch the printer off and on again.

9

Stopping and Starting the System

Lesson 9

Stopping and Starting the System

Introduction

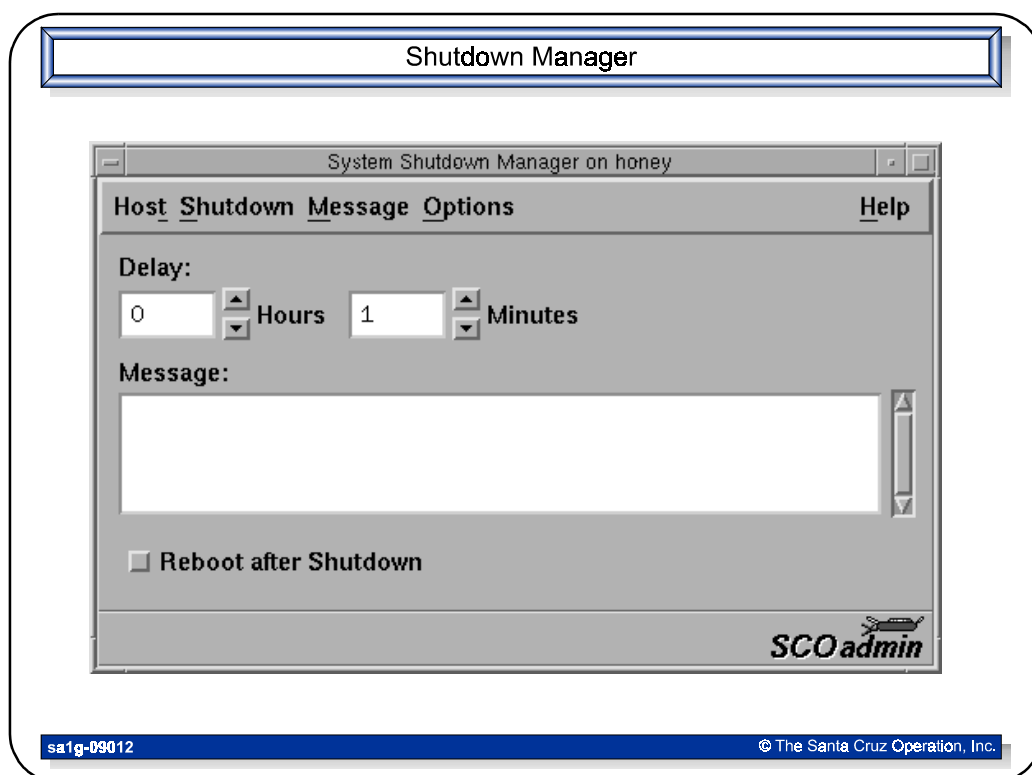
Starting and stopping a Unix system requires more than just turning the power on and off. You must follow a standard set of procedures to start and stop the system. An improper shut down can result in the loss of data and a corrupted system. This lesson demonstrates the necessary steps to ensure proper operation.

Objectives

This lesson describes the proper way to shut down and start up SCO OpenServer and UnixWare 7 systems. By the end of this lesson, you will be able to:

- Stop (shut down) a system
- Start (boot) a system
- Switch between system states

AV 9-1



9.1 Stopping SCO OpenServer and UnixWare 7 systems

When performing routine system maintenance you can shut down your system to:

- Replace system hardware
- Clear hung (defunct) processes or a hung system
- Upgrade system resources, such as processors, memory, or hard disks

Although the SCO OpenServer and UnixWare 7 filesystems are more robust than earlier releases, a proper shut down is still the best way to protect filesystems and user data from corruption.

These precautions are necessary when executing a safe shut down:

- Users must save their work and stop any user processes, including any background processes.
- All users should be logged off the system.

Only the **root**, **owner**, or a user with **shutdown** privileges can shut down the system with the **SCOadmin System Shutdown Manager**. Only the **root** user can shut down the system with the **shutdown** command.

The following topic describes how you can use the **SCOadmin System Shutdown Manager** to set:

- How many hours and minutes before the system shut down begins.
- What message (if any) displays before the shut down begins.
- Whether or not the system should automatically reboot after the shut down.

For additional information on the **System Shutdown Manager**, refer to **SCOhelp**.

9.1.1 Normal shut down

These steps describe how to perform a normal shut down using **SCOadmin**:

- Step 1. Start **SCOadmin**. Select **System** ⇒ **System Shutdown Manager**.
- Step 2. Select **Reboot after shut down** if you want the system to automatically reboot.
- Step 3. Enter the amount of time that elapses before the shut down takes place. If you:
- Do not enter an amount of time, the system automatically shuts down in 60 seconds when you perform Step 7.
 - Enter 0 in the **Minutes** field, the system shuts down immediately when you perform Step 7.
 - Enter 1 in the **Delay** field, and 15 in the **Minutes** field, the system shuts down in 1 hour and 15 minutes when you perform Step 6.
- Step 4. Enter a warning message in the **Message** box to alert users to save their work and log off the system. If you entered 0 in Step 3, you are unable to enter a message.
- If your message is consistent, you can create a message and save it to a file. Retrieve the message, select **Message** ⇒ **Read from File**.
- Step 5. Note that to save your settings as the default you can Select **Shutdown** ⇒ **Save**.
- Step 6. Select **Shutdown** ⇒ **Begin Shutdown**.
- Step 7. Click on **OK** to confirm the shut down. Select **Cancel** if you decide to stop the shut down.

You can also use the **shutdown** command to execute a shut down. Note: In UnixWare 7 you must be in the root (/) directory before issuing this command:

shutdown -g20 -y

This list describes the options used in this example:

- **-g** sets the time until the shut down starts.

NOTE: In SCO OpenServer, the time is in minutes. In UnixWare 7, the time is in seconds.

- **-y** allows the shut down to proceed without any further prompts, halts the system, and prompts with “Hit any key to reboot.”

Exercise: Shutting down the system

Create a warning message in */tmp*. Follow the steps under Section 9.1.1, “Normal shut down,” on page 9-6 to shut down the system so that the system warns users and then shuts down in 2 minutes.

9.1.2 Other shut down circumstances

If you bring the system down while users are still logged into the system, they can lose unsaved work. You should only bring the system down quickly if:

- The system is in single-user state
- The shut down process fails
- A hard disk crashes
- A security breach is detected

To bring the system down quickly, use the **shutdown** command with minimal delay and no final prompt:

```
shutdown -g0 -y
```

To shut down and reboot your system, enter:

```
shutdown -g0 -y -i6
```

NOTE: In addition, SCO OpenServer provides the **haltsys** and **reboot** commands that will bring the system down very quickly, though not gracefully. These commands should only be used from single-user mode or in emergency situations.

9.2 Starting the system

When you power up a system, it goes through several stages before the operating system is fully operational. These stages include:

- Loading the kernel program `/stand/unix` into memory
- Mounting the root filesystem. If necessary, the root filesystem is cleaned before mounting.
- Selecting the state of system operation

At any given time, the system is in one of eight possible states. A system state is a software configuration where a selected group of processes are set to execute. The **init** program is the general purpose process spawner of the system states. Refer to **init**, for more information on each system state.

9.2.1 Autoboot to multiuser mode

Under normal circumstances you can let the system boot automatically.

In addition to checking, cleaning if necessary, and mounting the root filesystem, it cleans and mounts any other filesystems that are configured to mount during system boot.

This automatic boot process is useful if the system loses power. When power is restored, the system can usually reboot to normal operations without requiring intervention.

Refer to the **boot** manual pages for instructions on how to change the automatic boot default values.

9.2.2 Other boot options

If you need to boot a kernel other than */stand/unix* or enter single-user state, you must respond at the boot prompt.

During the SCO OpenServer boot process, the boot prompt appears on the system console, as follows:

Boot
:

During the UnixWare 7 boot process, the UnixWare 7 logo screen displays. You can interrupt the automatic boot process and go to the UnixWare 7 boot prompt by pressing . The boot prompt displays as follows:

[boot]

9.2.2.1 Booting other versions of the UNIX kernel

If you press  at the boot prompt, the system boots the default kernel.

If the default kernel does not load correctly, you can load another version of the kernel. For example, to boot the *unix.old* kernel, enter:

SCO OpenServer:

Boot
: **unix.old** 

UnixWare 7:

[boot] boot unix.old 

9.2.2.2 Booting to single-user state

Some forms of system maintenance can require you to run the system in single-user state. For example, backing up the root filesystem in single-user state ensures that users are not modifying files while the backup occurs.

To boot the system to single-user state, access the boot prompt:

SCO OpenServer:

Boot

: 

UnixWare 7:

[boot] **INITSTATE=S** 

[boot] **boot**

When prompted, enter the **root** password.

While in single-user state:

- Only the **root** account is logged into the system.
- Many system services, such as printing and networking are not operating.
- No user filesystems are mounted.

Exercise: Booting to single-user state

Boot your system to single-user state. While in single-user state, enter the **ps** command. Which processes are running? Enter the **mount** command. Which filesystems are mounted?

9.2.2.3 Entering multiuser mode

The most commonly used run levels are:

0	shut down state
1,S	single-user state
6	reboot state
2	multiuser state for SCO OpenServer
3	multiuser/networking state for UnixWare 7

From the single-user state, to proceed to normal multiuser mode, use the **init** command to change run levels:

SCO OpenServer: **init 2**

UnixWare 7: **init 3**

When the system enters multiuser mode, the following occurs:

- Mounts any non-root filesystems that have been configured to mount during startup. If necessary, the filesystems are cleaned.
- All system-generated processes are running.
- Users can log into the system.

9.3 Switching Between System States

If you are in multiuser mode and need to enter single-user state, you can switch to single-user state without rebooting your system. This is useful when performing some system administration tasks.

Use the **shutdown** command to enter single-user state.

shutdown -g20 -y -i1

The **-i1** option represents the single-user state also referred to as *run-level 1*.

When the change completes, log in as **root**.

To return to multiuser mode, shut down your system and reboot.

9.4 Lesson Summary

Shutting down the system in an orderly fashion provides users with enough warning to save their work. In addition to the **SCOadmin System Shutdown Manager**, you can also use the **shutdown** command to switch between system states.

Left alone, the system can start up and enter the network state automatically. You can also boot different versions of the UNIX kernel, or bring the system into single-user state for system maintenance.

Review Questions

1. Which users can perform a shut down?
2. What is the difference between single-user and multiuser states?
3. Which command would you use to bring the system from the multi-user state to single-user state in 15 minutes, and warn the users that the system is coming down?

9.5 Relevant Commands and Files

Relevant commands

The following commands were presented in this lesson:

haltsys	Shuts down the system (SCO OpenServer only)
init	Is the general purpose process spawner of the system states
reboot	Shuts down the system and reboots (SCO OpenServer only)
scoadmin	Invokes SCOadmin applications (in this lesson, the System Shutdown Manager)
shutdown	Shuts down the system in an orderly manner

Relevant files

<i>/stand/unix</i>	The UnixWare 7 kernel that is loaded into memory during system startup.
<i>/etc/default/shutdown</i>	File where system shutdown defaults are kept (UnixWare 7).

9.6 Answers to Exercises and Review Questions

This section contains the answers to exercises and review questions covered in this lesson.

Review questions

1. The root, owner, and a user with shutdown privileges can perform a shut down.
2. In single-user state only the root account is logged into the system, and many system services, such as printing and networking, are not operating. Also, the system mounts only the boot (/stand) and root filesystems. In multi-user mode, the system checks and repairs filesystems, if necessary, and then mounts any non-root filesystems that are configured to mount during startup. In addition, all system-generated processes are running and users can log into the system.
3. To warn users and bring the system down to single-user state in 15 minutes, enter this command:

```
SCO OpenServer: shutdown -g15 -i1 -y  
UnixWare 7: shutdown -g900 -i1 -y
```

A

***ARCserveIT
Administration***

Appendix A

ARCserveIT Administration

Introduction

ARCserve®IT™ from Computer Associates® provides an easy to use graphical interface, integrated tape device management, and reports.

Objectives

This appendix describes how to use ARCserveIT to perform backups that enable the restoration of a non-root filesystem or critical files. By the end of this lesson, you will be able to:

- Describe ARCserveIT products
- Back up filesystems and files
- Verify a filesystem backup
- Restore filesystems and files
- Schedule unattended backups

1.1 Describing ARCserveIT Products

ARCserveIT is a fully featured, high performance data management tool used to back up and restore data on servers and all workstations attached to a network.

ARCserveIT uses the client/server architecture that functions within a group of ARCserveIT servers. The components include:

Web browser	Provides access to all of ARCserveIT operations using Java Applets
ARCserveIT manager	Functions as an information bridge between the browser and back-end services
Command line	Provides access to all ARCserveIT operations from the command line
Client Agent for UNIX	Enables UNIX workstations to be browsed and handles the transfer of data. The client module must be present on all systems that you want to backup or restore. Client Agents are included for: ● UnixWare 7 and SCO UnixWare® 2.1x ● SCO OpenServer™ Release 5 ● Microsoft Windows 3.x, Windows 95/98, and Windows NT workstations

New features of ARCserveIT include:

- Java/HTML Graphical User Interface (GUI) using Netscape Communicator 4.09 on UnixWare 7 and SCO OpenServer Release 5 and Netscape Communicator 4.07 on Windows.
- Multiple SCSI card support. Automatically detects tape drives on all SCSI cards.
- No device locking. Devices can be used by other backup programs, i.e., **tar**(1) and **cpio**(1).
- Other backup methods. You can specify to use either **tar**, **cpio**, or ARCserveIT as the method for backing up files. Note that ARCserveIT cannot restore tapes that are created using the **tar** and **cpio** command-line interfaces.

1.1.1 System requirements

Your system must be configured to meet ARCserveIT 6.6 minimum requirements:

Processor	Pentium Class
Memory	64MB
Disk space	250MB
Tape device	1 SCSI or IDE

1.1.2 Licensing ARCserveIT

ARCserveIT is licensed by default in all UnixWare 7 editions except the Base and Free UNIX Editions. You must purchase a license if it is not licensed by an edition.

The ability to backup up to 250 machines is included in each license.

1.1.3 Installing ARCserveIT

You can choose to install ARCserveIT during system installation. If not, add it later with the **SCOadmin Application Installer** or the **pkgadd(1M)** command.

For more information on how to install ARCserveIT, **refer to scohelp ⇒ Backup and Restore ⇒ ARCserveIT from Computer Associates ⇒ ARCserveIT Release and Installation Notes ⇒ Installing and removing ARCserveIT products.**

1.1.4 Configuring ARCserveIT

After installing the ARCserveIT component, configure ARCserveIT before using it. You are required to answer several prompts during the configuration:

Primary discovery host	A host that manages ARCserveIT data between a group of machines.
------------------------	--

Domain name	One or more ARCserveIT servers can be configured to form a distinct group. This group is referred to as an <i>ARCserveIT domain</i> .
Secondary discovery host	A host which is used when the primary is not available. (Optional)
Port number	The web server port number. The default port number is 6060.
Printer name	Name of printer used for log file printing.
Password	The arcroot account is used to manage ARCserveIT. You must assign the account a password.

To configure ARCserveIT after installation, log in as **root** and enter:

/usr/sbin/asetup

1.1.5 Starting ARCserveIT

The ARCserveIT services are started during configuration. After configuration, ARCserveIT services are automatically started each time the system boots to the networking state.

ARCserveIT is started automatically from the */etc/rc2.d/S96ARCserveIT* file.

If it becomes necessary to manually start ARCserveIT, enter:

/usr/sbin/astart

These ARCserveIT services are started:

ARCserveIT Loader (arcserve)	Responsible for loading and unloading all ARCserveIT services.
The Discovery Service (asdiscovd)	Responsible for starting and monitoring the discovery servers.
Authentication Service (asauthd)	Controls ARCserveIT resources on a per domain basis and verifies a user's permissions associated with each ARCserveIT server.

Scheduler (asqd)	Manages the jobs that are scheduled through the ARCserveIT Manager or from the command line.
Media Server (asmediad)	Handles the transfer of data to and from the media.
Database Server (asdbd)	Keeps track of the jobs run and all of the files backed up or restored by ARCserveIT.
Processing Server (aprocess)	Manages the backup and restoring of current jobs.
Event Logger (asloggerd)	Handles message logging and retrieving.

Use the **astatus** command to list the services started by ARCserveIT.

Exercise: Listing ARCserveIT services

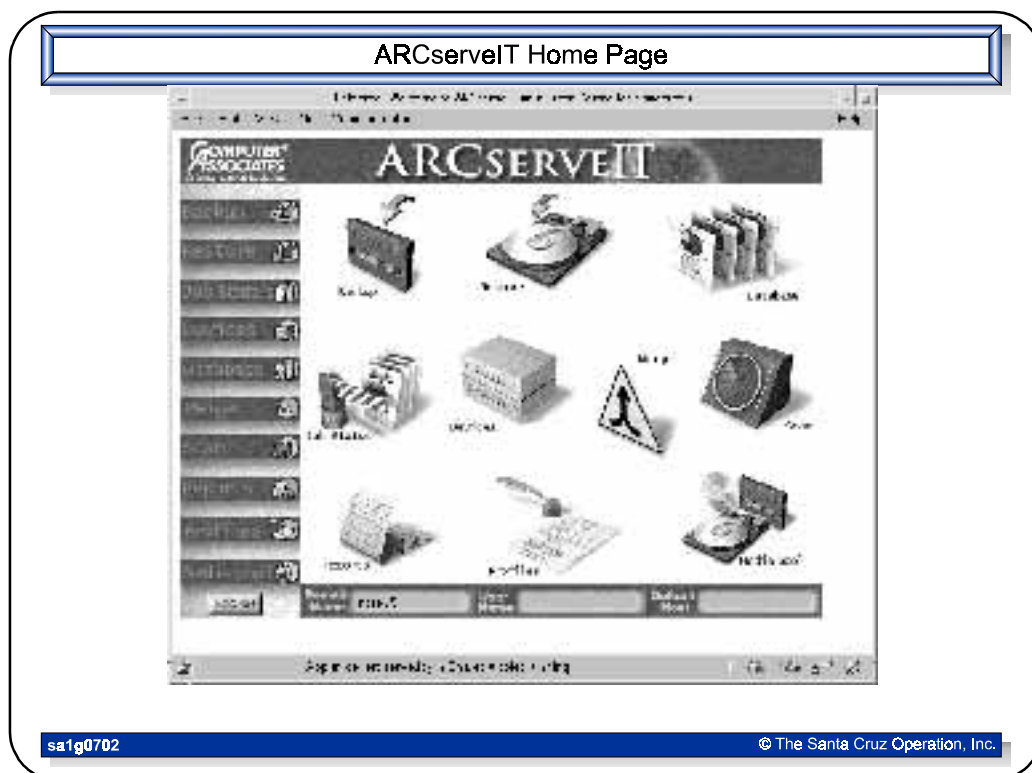
List the services started by ARCserveIT.

1.1.6 Stopping ARCserveIT services

ARCserveIT services are automatically stopped during shut down. If it becomes necessary to manually stop the ARCserveIT services, enter:

/usr/sbin/astop

AV 1-1



1.2 Accessing ARCserveIT

To access ARCserveIT, follow these steps:

- Step 1. Log in as **root**.
- Step 2. Start **SCOadmin**. Select **ARCserveIT Backup Manager**.

Menu options include:

- **Backup**—Backs up selected files and filesystems or an entire system
- **Restore**—Restores selected filesystems or files
- **Job Status**—Allows you to view and manage jobs currently in the job queue
- **Devices**—Displays information about your tape device and tapes. It is also used to perform tape maintenance.
- **Database**—Displays the amount of space required for a full system backup
- **Merge**—Merges one or more sessions on a given tape into the database
- **Scan** —Scans to see what sessions/files are on a given tape
- **Reports**—Allows you to view and manage logs and reports
- **Profiles**—Allows the **root** user to customize user access to ARCserveIT
- **Media pool**—Is a collection of media managed as a unit

Before making a selection you must log on.

- Step 3. Click on **LOG ON**. Enter the user name **arcroot** and the password. Click on **OK**.
- Step 4. Click on **OK** to accept the default host.

1.2.1 Exiting ARCserveIT

Before exiting the **Backup Manager**, you must log off. Continuing with the above steps:

Step 5. Click on **LOG OFF**.

Step 6. Select **File** ⇒ **Exit** to exit **ARCserveIT**.

1.2.2 Setting up user profiles

The **Profiles** menu selection lets the ARCserveIT **arcroot** user customize user access to ARCserveIT. This is done by adding a user to a specific group. The group determines what level of access a user has. Groups include:

ARCserve Administrator Group	Access to all functions. Equivalent to arcroot .
ARCserve Operator Group	Access to all functions except Profiles
ARCserve User Group	Access to functions required by a user to manage jobs owned by that user. Exceptions include: ● Profiles—can change own password ● Device configuration and media pool modification ● Control over ARCserveIT services (daemons) ● Adding of clients ● Database ● Cannot add or delete groups

1.2.3 Creating a user

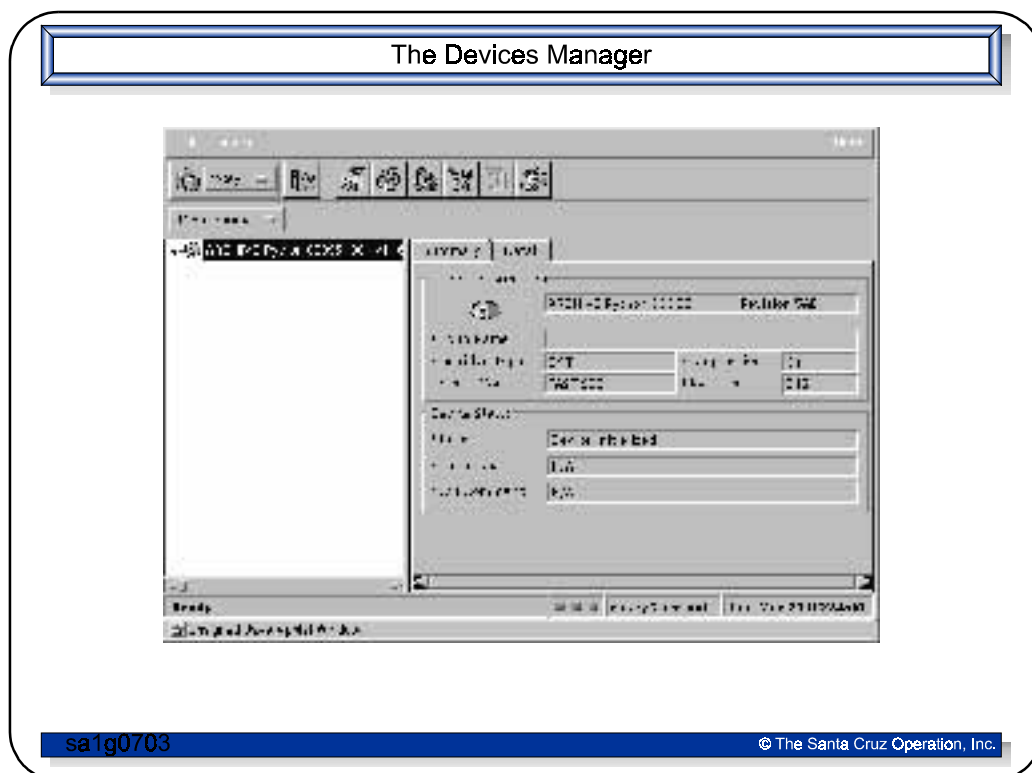
From the ARCserveIT main menu, follow these steps to add an ARCserve user:

Step 1. Select **Profiles**.

Step 2. Click on the **Add User** icon.

- Step 3. Enter the user name and password. Confirm the password.
- Step 4. Click on the group to which the user belongs. Click on **OK**.

AV 1-2



1.3 Accessing Tape Device(s)

Use the **Devices Manager** to access important information about your tape device. Before accessing the tape device:

- The tape device must be installed and configured.
- You must log in as **arcroot**.

1.3.1 Selecting devices and device views

The **Devices Manager** provides three ways to view your tape device using a detailed or summary view:

- The **Adapter View** displays information about your host adapters installed on your system.
- The **Device View** displays the name and a description of your tape device.
- The **Group View** displays to which group the device is a member.

To retrieve device information:

Step 1. Select **Devices**.

Step 2. Click on **Device View**. Click on **Detail**.

Important details to note:

- The name of the tape device
- The cartridge type
- Whether the tape device supports data compression
- The status of the tape device

Exercise: Retrieving tape information

Access the **Devices Manager**. What is the cartridge type? Does your tape device support data compression?

1.4 Maintaining Media

ARCserveIT provides three basic tape maintenance functions:

- **Formatting**—ARCserveIT automatically formats blank media. However, you can use this option to name and manually format your media. You can also assign an expiration date:

Expiration dates are based on how you will use the media. For example, if you are using the media only once a month, set the expiration date to two or three years from the current date.

A media's life is generally based on *passes*. For example, a backup without verification constitutes one pass. With verification, two passes.

Media manufacturers rate their media's life span from 500 to 1500 passes. You can use the media beyond the expiration date.

- **Erasing**—All data is removed from the media and all references to the contents of the media are removed from the database. You can perform three types of an erase:

Quick Erase—Erases the media label. The data is inaccessible.

Quick Erase Plus—Erases the label and allows you to re-label it.

Long Erase—Erases the label and the data.

- **Retensioning**—Ensures that a tape is evenly wound and properly tensioned. If you are having problems writing to, or reading from a tape, retension the tape.

You can only retension a Quarter Inch (QIC) tape. Do not retension an 8mm or 4mm Digital Audio Tape (DAT).

For example, to perform a Quick Erase of your tape, follow these steps:

- Step 1. Select the **Device Manager**. Verify that you are in the **Device View**.
- Step 2. Click on **Device**. Click on **Erase**.

- Step 3. Click on **Quick Erase**. Click on **OK**.
- Step 4. Click on **OK** for a successful completion.
- Step 5. Click on **File** ⇒ **Close** to exit the **Device Manager**.

1.4.1 Ejecting a tape

You can eject a tape from the **Devices Manager** if your tape device supports automatic ejection.

If you eject a tape that does not support ejection, ARCserveIT acts as though the tape is ejected. Physically eject the tape and reinsert.

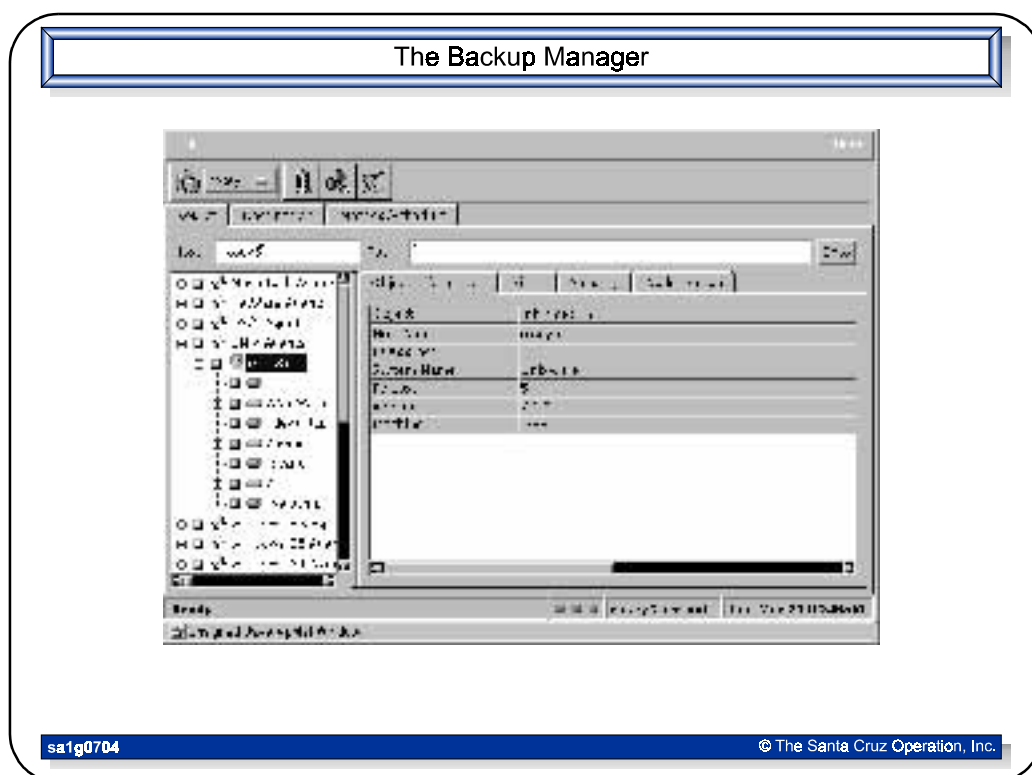
1.4.2 Using compression

If your tape device supports data compression, ARCserveIT turns it on by default. If it does not support data compression, N/A appears in the **Compression** window. To view if compression is applicable, select **Devices** ⇒ **Detail**.

Exercise: Performing Tape Maintenance

Perform a quick erase of your tape. When finished, eject the tape.

AV 1-3



1.5 Backing up files and filesystems

From the **ARCserveIT** home page, select **Backup** to back up individual files and filesystems.

The **Backup Manager** appears with the **Source** tab selected by default. The default view option “View all filesystems agents” lets you see the Client Agents that are running.

When you select the “UNIX Agents”, the UNIX machine(s) display. Each machine is known as an *object*.

1.5.1 Expanding and object

You can select the entire machine or select filesystems, directories, or files within a machine.

The machine’s **root** password is required before viewing data on a machine. After you enter this information, the machine’s filesystems display.

1.5.2 Viewing filesystems and files

ARCserveIT provides a *Tree* view for viewing filesystems and files. This view displays files in a single file hierarchy:

- The machine name
- The root (/) filesystem

- Mounted non-root filesystems appear as a disk drive with the name of the mount point (directory) adjacent. Mounted filesystems are shown in the directory structure to which they are mounted.

To expand and collapse the directory tree, click on the plus sign or minus sign adjacent to the system name.

To expand and collapse a machine's directory structure, follow these steps:

- Step 1. Select **Backup** from the ARCserveIT home page.
- Step 2. Click on **Unix Agents**.
- Step 3. Click on the plus sign adjacent to the machine name.
- Step 4. Enter the user name, **root**, and password. Click on **OK**.
- Step 5. Click on the plus sign adjacent to the root (/) directory to expand the directory and view files.
- Step 6. Click on the plus sign adjacent to a directory to view directories and files.
- Step 7. Click on the minus sign adjacent to a directory to collapse the directory and files.
- Step 8. Click on **File** ⇒ **Close** to return to the **Backup Manager**.

Exercise: View filesystems and files

Follow the steps in Section 1.5.2, "Viewing filesystems and files," on page A-19 and expand the */etc* directory. Collapse the tree so that only the machine name displays. Return to the **Backup Manager**.

1.5.3 Performing a system backup

You can quickly perform a complete backup of your system. The features of ARCserveIT are powerful, yet simple to use.

Depending on the capacity of your tape device, you might need several tapes to perform a complete system backup. During the backup you are prompted to insert additional tapes.

From the **ARCserveIT** home page, you can determine the approximate number of tapes needed for a full system backup from the **Database Manager**.

From the **ARCserveIT** home page, perform a complete backup of your system by following these steps:

- Step 1. Insert a tape into the tape drive.
- Step 2. Click on the **LOG ON** button. Enter the **arcroot** user name and password. Click on **OK** twice.
- Step 3. Select **Backup**. Click on **Unix Agents**.
- Step 4. Click on the box adjacent to your machine name in the **Source** area of your screen.
- Step 5. Enter the user name **root** and password for the machine.

 This box becomes solid, indicating that all files on the system are selected.
- Step 6. Click on **Destination** ⇒ “Tape group.”
- Step 7. Click on the **Traffic Light** icon. Click on **Run now**.
- Step 8. Enter a description of the job.
- Step 9. Click on **OK** to acknowledge a successful submission. The backup begins.
- Step 10. Select **File** ⇒ **Close** to exit the **Backup Manager**.

Exercise: Performing a full system backup.

Follow the steps under Section 1.5.3, “Performing a system backup,” on page A-23 and perform a full system backup.

1.5.4 Backing up filesystems

When you select the mount point of a filesystem, all files in that filesystem are selected for backup.

From the **ARCserveIT** home page, you can backup a filesystem by following these steps:

- Step 1. Insert a tape into the tape drive.
- Step 2. Perform this step if you are not logged on. Click on the **LOG ON** button. Enter the **arcroot** user name and password. Click on **OK** twice.
- Step 3. Select **Backup**. Click on **Unix Agents**.
- Step 4. Click on the box beside your machine name in the **Source** area of your screen.
- Step 5. Enter the user name **root** and password for the machine. Click on **OK**
- Step 6. Click on the box adjacent to the appropriate filesystem.
- Step 7. Click on the **Traffic Light** icon. Click on **Run now**.
- Step 8. Enter a description of the job.
- Step 9. Click on **OK** to acknowledge a successful submission. The backup begins.
- Step 10. Select **File** ⇒ **Close** to exit the **Backup Manager**.

1.5.5 Backing up selected files

You can also backup selected files within a filesystem. From the ARCserveIT home page, follow these steps to backup selected files:

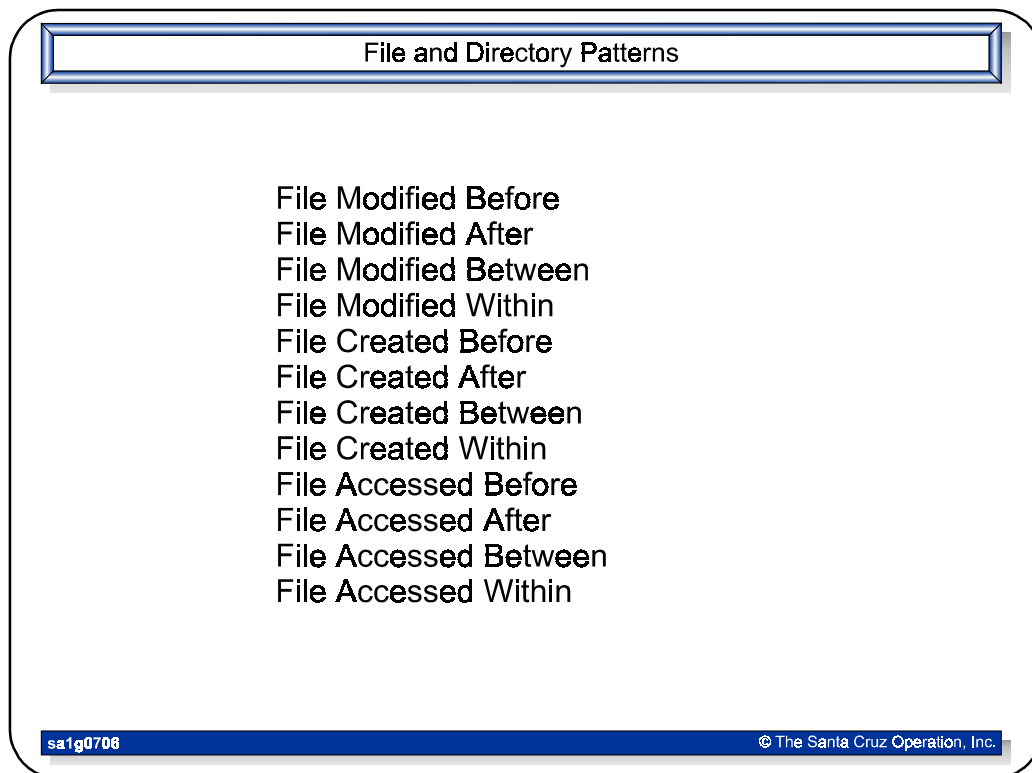
- Step 1. Complete Steps 1 through 6 in Section 1.5.4, “Backing up filesystems,” on page A-24.
- Step 2. Click on the plus sign adjacent to the filesystem that contains the files. The filesystem is expanded.

- Step 3. Click on the box adjacent to the filesystem that contains the files. This deselects all files in the filesystem.
- Step 4. Click on the box adjacent to the files that you want to back up.
- Step 5. Click on **the Traffic Light** icon. Click on **Run now**.
- Step 6. Enter a description of the job.
- Step 7. Click on **OK** to acknowledge a successful submission. The backup begins.
- Step 8. Select **File** ⇒ **Close** to exit the **Backup Manager**.

Exercise: Backing up selected files

Use the above steps and back up all of the files in the `/etc/default` directory.

AV 1-5



1.5.6 Monitoring the status of a backup

You can monitor the status of the backup by following these steps:

- Step 1. Click on **Job Status** from the **ARCserveIT** home page.
- Step 2. Select the **Job**.
- Step 3. Select **Job Status** ⇒ **Monitor Job**.
- Step 4. Select **File** ⇒ **Close** to exit **Job Status**.

1.5.7 Backup filter options

Filter options are provided as a means to further distinguish files that you want to include as part of your backup. Filter options allow you to include or exclude both files and directory patterns as well as other options.

For example, to include all files that were created before a specified date, enter:

- Step 1. **Backup** ⇒ **Source**.
- Step 2. Click on the **Filter** icon.
- Step 3. Select **Include**.
- Step 4. Click on **File Accessed Before**.
- Step 5. Enter the date. Click on **Add**.
- Step 6. Click on **OK**.

The filter is added. You can select this filter for future backups or delete the filter when it is no longer needed.

1.5.8 Backup method

You can apply options to the destination media which include:

- | | |
|--------------------|---|
| Eject Media | If you select this option, the media is ejected after the backup job completes. |
|--------------------|---|

- First Media Options** Use these options if you have one tape drive. By default, ARCserveIT appends any data to the selected media. Select from the following to override the default:
- **Overwrite Same Media Name, or Blank Media.** The media name must be the same or must be blank media to overwrite the media. If it is not, the backup will fail.
 - **Overwrite Same Media Name, or Blank Media First, then any Media.** Overwrite *any* media that is in the drive.
 - **Overwrite Same Media Name, or any Media first, then Blank Media.** This is the same as the first option, except that if the media has been used in a rotation scheme, the backup will fail.
- Span Media Options** Use these options in addition to First Media Options if you have multiple tape drives. The definitions are similar to First Media Options.

Time-out for First Media

Number of minutes ARCserveIT waits before accessing the media. If ARCserveIT cannot access the media in the specified time, the job is cancelled.

To apply any of these options, from the **Backup Manager**, select **Destination** ⇒ **Method/Schedule** ⇒ **Custom** ⇒ **Media Rules**.

1.5.9 Backup schedule

In addition to running the backup immediately, you can schedule the backup to run at a specified time, or submit a backup that is on hold.

After specifying the source and destination, to schedule the backup for a later time, select the **Submit** icon from the **Backup Manager**:

- To schedule the job at a specified date and time, click on **Schedule at**. Enter the date and time.
- To submit a job that is on hold, click on **Schedule** ⇒ **Submit job on HOLD** and select from the jobs in the queue.

A backup schedule can also be created to run on a repeating basis. For example, if you wanted a backup to run every Monday, Wednesday, and Friday at 3:30 a.m., follow these steps:

- Step 1. Select **Method/Schedule** from the **Backup Manager**.
- Step 2. Enter 3 in the **Hours(s)** field and 30 in the **Minute(s)** field.
- Step 3. Exclude the days that you do not want the backup to run. For this example, click on Sunday, Tuesday, Thursday, and Saturday in the “Exclude Days” menu.
- Step 4. Click on the **Submit** icon. Click on **OK** to submit the job.

1.5.10 Placing backup jobs on hold

If you have scheduled a backup job to run at a future time and decide that you do not want the job to run, place the job on hold using the **Job Status Manager**.

1.5.11 Saving backup jobs as scripts

When running the same backup repeatedly, rather than performing the steps to create a backup each time, you can save the information to a script (file).

Before submitting the job, save the information to a script.

The script contains the:

- Source
- Destination
- Any scheduling information
- Any filters to include or exclude directories and files

Follow these steps to create a script:

- Step 1. Initiate a job from the **Backup Manager**.
- Step 2. Schedule and customize the job
- Step 3. Enter the backup job description.

- Step 4. Click on **Save**.
- Step 5. Enter the script name. Click on **OK**. Click on **OK** for a successful save of the script.
- Step 6. Click on **OK** to submit the job. Click on **OK** for a successful submission.

To run the backup script:

- Step 1. Select the **Backup Manager**.
- Step 2. Click on **File** ⇒ **Open**.
- Step 3. Click on the **Run** icon.

1.6 Verifying the backup and checking logs

You can verify the results of your backup by scanning your media and checking the available backup logs.

1.6.1 Scanning media

Use the **ARCserveIT Scan Manager** to verify your media. This manager provides you with information about your media sessions. You can scan a single session or the entire media. Results of the media scan can be viewed in the **Reports Manager** under the **Database Log**.

To scan the contents of your tape that is currently in the tape drive:

- Step 1. Select **Scan** from the **ARCserveIT** home page.
- Step 2. Click on **All Sessions** ⇒ Click on **Current media in drive**.
- Step 3. Click on the **Run** icon. Click on **Run now**. Click on **OK**.
- Step 4. Click on **OK** for a successful submission.
- Step 5. Click on **File** ⇒ **Close** to exit the **Scan Manager**.

The contents of the tape is scanned and detailed information including filenames and sizes is written to the **Reports Manager** ⇒ **Database Report** ⇒ **Media Session Detail Report**.

1.6.2 Viewing Reports and Logs

Whenever a job is submitted to the scheduler, the ARCserveIT logs are updated. The logs should be viewed on a regular basis to make certain errors are not encountered while performing backup jobs. ARCserveIT communicates errors to the operator through these reports.

Important logs include the **Activity Log** and the **Media Server Log**.

1.6.2.1 Activity Log

The **Activity Log** contains information about the operations performed by ARCserveIT. For each job, the log includes the following:

- Date and time the job started and ended
- Type of job
- Average throughput of the data
- Job session number
- Result of the job
- Messages that indicate errors display in red

To display the **Activity Log**:

- Step 1. Select **Report** from the ARCserveIT home page.
- Step 2. Select **Activity Log**.
- Step 3. The **Activity Log** displays.

Exercise: Viewing the Activity Log

View the **Activity Log** for your classroom system.

1.6.2.2 Media Server Log

The **Media Server Log** provides an audit trail of all ARCserveIT Media Server activity, including when the Media Server is loaded and unloaded.

The **Media Server Log** also provides information about tape jobs including:

- Backups
- Formatting
- Retension
- Erasing

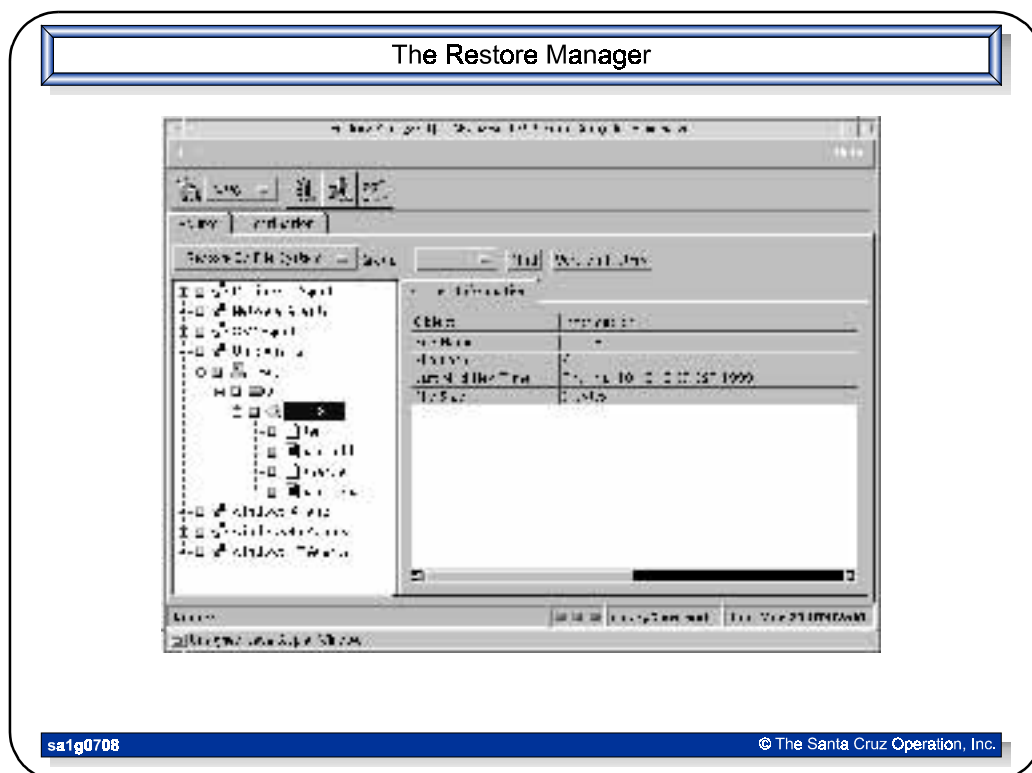
For each job, the log includes:

- Time the job started
- ID number of the job
- User who submitted the job
- Activity performed
- Job session number
- Errors displayed in red

Exercise: Viewing the Media Server Log

View the Media Server Log for your classroom system.

AV 1-7



1.7 Restoring files and filesystems from backups

You can restore files, directories, and filesystems using the ARCserveIT **Restore Manager**. View and restore data using three different methods:

Restore by File System	This is the default method. Using this method shows you a graphical directory tree of each filesystem. You can restore an entire filesystem(s) or specific files and directories.
Restore by Session	This method allows you restore an entire single session or select files within the session.
Restore by Backup Media	This method restores all files in a session.

1.7.1 Browsing media

The **Restore Manager** is divided into two sections:

- The **Source** section describes how the data is stored on your media. You specify what data is being restored.
- The **Destination** section displays the directory hierarchy of the machine to which data is restored. You specify to where the data is being restored.

To access the **Restore Manager**,

Step 1. Select **Restore** from the ARCserveIT home page.

The **Source** tab and **Restore by File System** are selected by default.

Step 2. Click on the plus sign adjacent to Unix Agents. Continue expanding the directory structure.

To view files by session:

Step 3. Click on **Restore by File System** and select **Restore by Session**.

Step 4. Click on the plus sign adjacent **Media** icon. Select a media session. Continue expanding the media session.

Step 5. Exit the **Restore Manager**. Click on **File** ⇒ **Close**.

1.8 Performing a Basic Restore Job

Restoring data can be a quick and flexible task. When accessing the **Restore Manager**, remember that **Restore by Filesystem** is the default method. To restore files using the **Restore Manager**:

- Step 1. Insert the tape into the drive.
- Step 2. Select **Restore** from the ARCserveIT home page.
- Step 3. Click on the plus sign adjacent to the UNIX Agents.
- Step 4. Click on the plus sign adjacent to the machine name.
- Step 5. Click on the plus sign adjacent to the root (/) directory.
- Step 6. Continue expanding the directory structure until you reach the directories and files to restore.

 To restore all files in a non root filesystem, click on the box adjacent to the filesystem name.

 To restore an entire directory structure, click on the box adjacent to the directory name.
- Step 7. Specify the destination. Click on **Destination**.
- Step 8. Click on the plus sign adjacent to UNIX Agents.
- Step 9. Click on the plus sign adjacent the destination machine. Enter the root user name and password. Click on **OK**.
- Step 10. Expand the directory structure until you reach the destination directory.
- Step 11. Restore the data. Click on the **Run** icon.
- Step 12. Click on **Run now**. Click on **OK**.
- Step 13. Click on **OK** for a successful submission. The files are restored.
- Step 14. Click on **File** ⇒ **Close** to exit the **Restore Manager**.

Information about your restore can be found in the **Reports Manager**.

1.8.1 Restore filter options

Filter options are provided as a means to further distinguish files that you want to include as part of your restore. Filter options allow you to include or exclude certain file or directory patterns.

To access the filter options, click on the **Filter** icon in the **Restore Manager** and set the filters before restoring the job.

Exercise: Restoring the /etc/default/directory

Restore all files in the /etc/default/directory to /tmp. As a guide, follow the steps in Section 1.8, “Performing a Basic Restore Job,” on page A-38.

1.9 Lesson Summary

The UnixWare 7 system provides the ARCserveIT backup tool. This tool allows you to quickly backup and restore either entire filesystems or selected files.

Review questions

1. How are the files and directories displayed and accessed in the source section of the ARCserveIT Backup Manager?
2. What are the two areas where you must make selections when performing a restore?

1.10 Relevant Commands and Files

The following commands and files were presented in this lesson.

Relevant commands

scoadmin(1M)	Invokes SCOadmin applications (in this lesson, the ARCserveIT Backup Manager)
/usr/bin/asetup	Configures ARCserveIT after installation
/usr/bin/astart	Starts ARCserveIT from the command line
/usr/bin/astatus	Lists the services started by ARCserveIT
/usr/bin/astop	Stops ARCserveIT from the command line

Relevant files

<i>/etc/rc2.d/S96ARCserveIT</i>	Starts up ARCserveIT when entering the networking state
---------------------------------	---

1.11 Answers to Review Questions

This section contains the answers to review questions presented in this lesson.

1. The backup source files and directories are displayed in a Tree View. By clicking on the directory icons you can expand or collapse the tree.
2. When performing a restore you must select the directory or files to be restored by clicking on the desired filesystem, file, or directory tree in the Source Area. The arrow designates whether or not a directory tree is expanded. You must also select the directory to be used for the restored data, in the Destination Area. The destination can be different than the location from where the files were backed up.