System Adminstrator’s Guide

Network Appliance, Inc.
2770 San Tomas Expressway
Santa Clara, CA 95051, USA
Tel: +1 408 367-3000
Fax: +1 408 367-3151
Support tel: (888) 4-NETAPP
Support email: support@netapp.com
Information email: info@netapp.com
Web: http://www.netapp.com

Part number 210-01799
November 1997

Copyright

Copyright © 1997 Network Appliance, Inc. All rights reserved. Printed in the U.S.A.

No part of this book covered by copyright may be reproduced in any form or by any means — graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system — without prior written permission of the copyright owner.

Portions of this product may be derived from the Berkeley Net2 release publicly distributed by the University of California.

Portions of this product contain NetBSD, which is copyrighted and publicly distributed by Carnegie Mellon University.

Copyright © 1994, 1995 Carnegie Mellon University. All rights reserved. Author Chris G. Demetriou.

Permission to use, copy, modify, and distribute this software and its documentation is hereby granted, provided that both the copyright notice and its permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof, and that both notices appear in supporting documentation.

CARNEGIE MELLON ALLOWS FREE USE OF THIS SOFTWARE IN ITS “AS IS” CONDITION. CARNEGIE MELLON DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

Network Appliance reserves the right to change any products described herein at any time, and without notice. Network Appliance assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by Network Appliance. The use and purchase of this product do not convey a license under any patent rights, trademark rights, or any other intellectual property rights of Network Appliance.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademarks

Data ONTAP, Web Filer, Filerview, Network Appliance, and WAFL are trademarks and FAServer, NetApp and Network Appliance and logo are registered trademarks of Network Appliance Inc.

Macintosh is a registered trademark of Apple Computer, Inc. Digital and StorageWorks are trademarks of Digital Equipment Corporation. Microsoft is a registered trademark, and Windows and Windows NT are trademarks, of Microsoft Corporation. NFS and SunOS are registered trademarks of Sun Microsystems, Inc. Ethernet is a registered trademark of Xerox Corporation. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

All other products or services mentioned in this document are identified by the trademarks, service marks, or product names as designated by the companies who market these products. Inquiries concerning such trademarks should be made directly to those companies.

Communications regulation information

Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the users will be required to correct the interference at their own expense.

Cables

To maintain compliance with applicable emission standards, use cables shielded with metallic RFI/EMI connector hoods on this device.

Canada notice

This digital apparatus does not exceed the Class A limits for radio noise emissions from digital apparatus set out in the radio interference regulations of the Canadian Department of Communications.

Avis Canadien

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la classe A prescrites dans le Règlement sur le brouillage radioélectrique édicté par le ministère des Communications du Canada.

BZT Hinweis
Hiermit wird bescheinigt, daß dieses Gerät in Übereinstimmung mit den Bestimmungen der BAPT-AmtsblVfg 243/1991 funkentstört ist. Der vorschriftsmäßige Betrieb mancher Geräte (z.B. Meßsender) kann allerdings gewissen Einschränkungen unterliegen. Beachten Sie deshalb die Hinweise in der Bedienungsanleitung.

Dem Bundesamt für Zulassungen in der Telekommunikation wurde das Inverkehrbringen dieses Gerätes angezeigt und die Berechtigung zur Überprüfung der Serie auf die Einhaltung der Bestimmungen eingeräumt.

Safety warnings

Failure to follow the directions contained in the following statements could result in bodily harm or loss of life.

- To reduce the risk of personal injury from hot surfaces, allow the internal components time to cool before touching or installing options.
- To reduce the risk of personal injury or damage to the equipment, ensure that the equipment is properly supported or braced when installing options.
- This equipment is designed for connection to a grounded outlet. The grounding type plug is an important safety feature. To avoid the risk of electrical shock or damage to the equipment, do not disable this feature.
- This equipment is furnished with one or more lithium batteries. There is a danger of explosion and risk of personal injury if the battery is incorrectly replaced or mistreated. Do not attempt to recharge the battery, disassemble it, or dispose of it in a fire.

Table of contents

About this guide xxi

If you are installing a filer for the first time xxi

If you are upgrading a filer xxii

Audience xxiii

Terminology and conventions xxiv

Keyboard conventions xxv

Special messages xxvi

Chapter 1:
Introducing Network Appliance filers 1

Appliance philosophy 1

Features and benefits 2

Multiprotocol support 2

Quick installation 2

Fast system boot 2

Single partition 2

RAID, on-line rebuild, hot spare, and hot swap 3

Snapshots 3

System backup and restore 3

Disk quotas 4

NVRAM cache 4

Network protocols 4
(PC) NFS 5
Chapter 2:
Getting started 7
Before you start 8
Setting up the filer 9
Turning on the power and booting 9
Running the setup program 10
Preparing the filer for delivering CIFS service 30
Modifying /etc/passwd and /etc/group 30
Assigning rights to users for each share 35
Preparing the filer for delivering HTTP service 37
Verifying network connections 38
Preparing clients for file sharing 39
NFS clients 39
CIFS clients 40
Getting on-line help from the filer 42
Using the man command to view on-line help 42
Using a web browser to view on-line help 42
Getting help at the filer 43
Storing the filer diskettes 44
Chapter 3:
Sharing files between NFS and CIFS users 47
File-locking interactions 49
Managing symbolic links for CIFS access 50
Enabling symbolic links 50
Redirecting absolute symbolic links 52
Preventing symbolic link cycling 52

NFS and CIFS use of the read-only bit 54
Naming files used by both NFS and CIFS 56
Maximum length of file names 56
Legal characters used in file names 57
Case sensitivity in file names 57
Timing of directory conversion for CIFS access to NFS-created files 59
Chapter 4:
Routine filer administration 61
Using the administration host 62
Editing configuration files 63
From an NFS client 63
From a CIFS client 63
Obtaining access to the9 filer shell 64
Halting and rebooting the filer 66
Understanding the default configuration 68
The root directory 68
The /home directory 70
The /etc/rc file 72
The /etc/rc default contents 72
Editing /etc/rc 75
Changing SNMP commands in /etc/rc 75
Naming conventions for network interfaces 77
Recovering from configuration errors 78
Core files 79
Message logging 80
facility 80
level 80

action 81
Configuring filer options 82
Autosupport options 84
CIFS options 85
DNS options 88
HTTP options 88
NFS and (PC) NFS options 89
NIS options 91
RAID options 92
Snapshot options 92
Miscellaneous options 93
Automatic email to Network Appliance 96
How the autosupport daemon works 96
Contents of automatic email messages 97
Using the options command to configure autosupport 98
Synchronizing filer system time 101
Setting the date and time 101
Synchronizing the filer 101
Maintaining filer security 103
Software licenses 104
Enabling NFS, CIFS, or HTTP service 104
Displaying current license codes 106
Disabling a license 106
Replacing license codes 106
Chapter 5:
Network administration 107
Using SNMP 108

Configuring the SNMP agent 108
Using the Network Appliance Custom MIB 109
Name resolution 111
The /etc/hosts file 112
Using DNS 114
Disabling DNS 116
Changing your DNS domain name 116
Using NIS 117
Disabling NIS 118
Changing your NIS domain name 118
Displaying the NIS server name 118
Routing 119
Routing table on the filer 119
Specifying default routers 119
Displaying routing status 121
The /etc/dgateways file 121
How the filer replies to requests 122
Using ifconfig to configure an interface 124
ATM PVCs 127
Establishing an outgoing PVC on the filer 127
Establishing an incoming PVC on the filer 128
Establishing a PVC on the remote ATM host 128
Establishing a PVC on interconnecting ATM switches 129
Displaying information about a PVC 129
Deleting a PVC 130

Chapter 6:
NFS administration 131
Exporting NFS file systems 132
Exporting to individual clients 132
Exporting to Subnets 136
Configuring a filer for (PC) NFS 138
Starting the (PC) NFS server 138
Default umask 140
Configuring a filer for WebNFS 141
Restricting access 141
Turning on WebNFS 141
Specifying and setting a root directory 142
Changing a root directory 142
Disabling the root directory 142
Displaying NFS statistics 143
Chapter 7:
CIFS administration 149
Adding clients to the filer 150
Using CIFS commands with rsh 151
Guest access 152
Guest account 152
Generic account 153
Creating a directory for a CIFS user 154
Managing shares 155
Creating a share 155
Changing the description of a share 156
Displaying information about shares 157
Deleting a share 158

CIFS share offer of a “home” directory for each user 158
Limits on CIFS open files, sessions, and shares 162
Choosing the method for deciding access rights 163
Selecting the method for deciding access rights..... 163
Managing access rights for users or groups 170
Assigning access rights..... 170
Removing an entry from a share’s ACL..... 170
Displaying access rights to a file 171
Using CIFS oplocks 172
Using NetBIOS scope IDs..... 174
Displaying CIFS statistics..... 175
Displaying CIFS session information 176
Stopping CIFS service 177
Restarting CIFS service 179
Chapter 8:
HTTP administration 181
Enabling HTTP service 182
Protecting web pages with passwords..... 184
The /etc/httpd.access file..... 184
The /etc/httpd.passwd file 185
The /etc/httpd.group file..... 185
Web page protection examples..... 186
Using the HTTP virtual firewall..... 187
Using virtual hosting 188
Using the /etc/httpd.hostprefixes file..... 188
Mapping virtual host addresses..... 189
Specifying MIME Content-Type values 190

Translating URLs 192
Translation rules supported by the filer 192
How the filer processes rules 194
Displaying HTTP connection information 196
Displaying HTTP statistics 197
Chapter 9:
Managing disks 199
Data, parity, and hot spare disks 200
How disks are numbered 201
RAID disk number 201
Disk SCSI address 201
Disk ID number 202
Locating a disk using different identifiers 202
Usable space on each disk 204
Using disks of various sizes 207
When replacing a failed disk 207
When adding a disk to the file system 207
When installing a disk as a hot spare 208
File system size limits and adding disks 209
Installing a new data disk 210
Installing multiple disks 211
Adding or removing hot spares 212
If your filer supports hot swapping 212
If your filer doesn't support hot swap 212
Adding a hot spare to the RAID file system 213
Hot swap 214
Swapping a disk out of the filer 214

Swapping a disk into the filer 217
Hot swapping a disk while dump is running 218
Disk errors during hot swap 218
Moving a disk from one filer to another 220
Increasing total disk capacity 220
Reducing the number of data disks 226
Disk failures 227
Without a hot spare 227
With a hot spare 228
RAID reconstruction speed 229
RAID scrubbing 229
Managing disk space 232
Restricting disk usage by imposing disk quotas 232
Increasing the maximum number of files 242
The df command 242
Using the df command with quota trees 244
Chapter 10:
Snapshots 245
Snapshot commands 247
Automatic snapshot creation 248
How snapshots work 252
Understanding snapshot disk consumption 254
Disk consumption by multiple identical snapshots 254
How df displays snapshots 254
How the snapshot reserve works 255
Managing snapshot disk consumption 257
Scheduling snapshots 257

Displaying snapshot statistics 259
Changing the snapshot reserve 262
Freeing up disk space by deleting a snapshot 262
Accessing snapshots from clients 265
Obtaining access to snapshots from an NFS client 265
Obtaining access to snapshots from a CIFS client 267
Determining snapshot versions 268
Determining access times 268
Effects of snapshots on quotas 269
Snapshot options 269
Chapter 11:
Dump and restore 271
Backing up the filer to tape 273
Features of the filer's dump command 273
Planning for system backups 274
Dump and restore for large file systems 275
Determining the number of tapes required 275
Supported tape drives 277
The dump command syntax 277
Dumping multiple subtrees 283
Restoring dump files 285
Restoring quota trees 286
The restore command 286
The restore command syntax 287
Locally restoring a subtree 290
Restoring the entire file system 291
Using a filer's tape drive from another system 294

Backing up the filer using NDMP 296
Controlling tape devices 299
The mt command syntax 299
Moving a tape to the end of data 300
Appending a dump 300
Rewinding a tape 300
Taking a tape off line 300
Displaying status information 301
Chapter 12:
System information and performance 303
Displaying the Data ONTAP version 303
Displaying filer configuration using sysconfig 304
Displaying disk information using sysconfig -d 304
Displaying RAID information using sysconfig -r 304
Displaying tape drive information using sysconfig -t 306
Displaying overall filer information using sysconfig -v 306
Displaying overall filer information using sysconfig 312
Displaying filer statistics 313
The sysstat command 313
The uptime command 315
Improving filer performance 316
Displaying network statistics using netstat 318
Chapter 13:
Troubleshooting 321
Restarting a down filer 322
NVRAM problems 323
NVRAM battery failure 323
Inconsistent NVRAM contents 323

Disk problems 324
Disk failure without a hot spare 324
Disk failure with a hot spare 325
Inconsistent disk labels 325
Inconsistent file system 325
Configuration problems 326
Filer not accessible from the administration host 326
Damaged or missing /etc/rc file 326
Booting from diskette 328
Recovering from configuration file problems 328
Lost password 330
Option for installing a new file system 331
Maintenance mode boot 331
Networking problems 332
Trouble using UNIX cpio 334
False UNIX df file space display 334
NFS problems 336
Software problems 337
Getting technical help 338
Appendix A:
Time zone names 339
Africa 339
America 340
Antarctica 342
Asia 342
Atlantic 343
Australia 343

Brazil 344
Canada 344
Chile 344
Etc..... 345
Europe 346
GMT..... 347
Indian (Indian Ocean) 347
Mexico 347
Miscellaneous..... 348
Pacific 349
System V 349
US (United States)..... 350
Aliases 350
Appendix B:
Configuration Record 351
Filer information 351
Other systems information 352
CIFS information 352
Appendix C:
Glossary..... 355
Index..... 363

List of figures

Figure 2-1. Add Computer To Domain window 21
Figure 4-1. /etc/rc file 72
Figure 4-2. Customized /etc/syslog.conf 82
Figure 5-1. /etc/hosts file 113
Figure 5-2. The routed status display 121
Figure 5-3. arp -a output 129
Figure 6-1. nfsstat output 144
Figure 6-2. nfsstat -l output 145
Figure 6-3. nfsstat -h output 146
Figure 7-1. cifs shares output 161
Figure 7-2. cifs stat output 175
Figure 7-3. cifs sessions output 176
Figure 7-4. Filer shutdown message displayed on an NT client 177
Figure 7-5. cifs terminate output 178
Figure 8-1. /etc/log/httpd.log example 196
Figure 8-2. httpstat output 197
Figure 9-1. Sample sysconfig -r output 202
Figure 9-2. Disk space on a filer upgraded from a pre-2.1 release 205
Figure 9-3. Disk space on a new filer shipped with Release 3.1 or later 206
Figure 9-4. Sample disk configuration 207
Figure 9-5. Effects of adding a disk to the file system 208
Figure 10-1. An example of the snap sched command 249

Figure 10-2. How snapshots use disk space 252
Figure 10-3. Filer toaster home directory tree 265
Figure 11-1. Using NDMP to back up data to tape 296
Figure 12-1. sysconfig -d output 304
Figure 12-2. sysconfig -r output 305
Figure 12-3. sysconfig -t output 306
Figure 12-4. sysconfig -v output on a FAServer 1400 308
Figure 12-5. sysconfig -v output on a NetApp F330 309
Figure 12-6. sysconfig -v output on a NetApp F630 310
Figure 12-7. sysstat output 313
Figure 12-8. netstat output 318
Figure 12-9. netstat -i output 318
Figure 12-10. netstat -I output 318
Figure 12-11. netstat -r output 319

List of tables

Table 2-1. Ethernet interface media types 13
Table 2-2. CIFS authentication features 19
Table 2-3. Connecting to the filer using CIFS 41
Table 2-4. Chapter to read after you set up the filer 45
Table 4-1. Filer commands that can be executed with rsh 65
Table 4-2. Configuration files in /etc 69
Table 4-3. Events that generate automatic email 97
Table 5-1. Groups in the custom MIB 110
Table 5-2. Default order for name resolution 111
Table 5-3. Media types for Ethernet interfaces 125
Table 7-1. CIFS limits by filer memory configuration 162
Table 7-2. Absolute mode values 167
Table 7-3. Symbolic mode components 168
Table 9-1. Comparing disk space between disks 204
Table 9-2. Usable disk space for various classes of disks ... 204
Table 9-3. Maximum size of file system, by filer model 209
Table 10-1. Commands for snapshots 247
Table 11-1. Option keys for dump 280
Table 11-2. Restore command function keys 288
Table 11-3. Restore option keys 288

Preface

About this guide

This guide describes how to configure, operate, and manage Network Appliance filers that run Data ONTAP™ Release 4.3. It covers all filer models. The following sections describe how to use this guide.

If you are installing a filer for the first time

Start with the chapters listed in the following table to familiarize yourself with the features and functions of the filer. The information in these chapters is sufficient for you to prepare the filer for delivering file-sharing service.

Chapter	Provides information about
Chapter 1, "Introducing Network Appliance filers"	Network Appliance filer features
Chapter 2, "Getting started"	Software setup and preparation of filer and clients for file sharing

If you are upgrading a filer

This guide was prepared with the assumption that you have followed the upgrade instructions in the *Read Me First*.

If you have purchased the license for CIFS or HTTP, read Chapter 2, "Getting started." For information about how to set up the filer for these protocols and Chapter 4, "Routine filer administration," for information about entering license codes.

For both new installations and upgrades

If your filer is used for both NFS and CIFS, read Chapter 3, "Sharing files between NFS and CIFS users," for information about file sharing between NFS and CIFS clients.

For information independent of the file-sharing protocol, refer to Chapter 4, "Routine filer administration," and Chapter 5, "Network administration."

Chapter 6, "NFS administration," Chapter 7, "CIFS administration," and Chapter 8, "HTTP administration" provide NFS, CIFS, and HTTP information, respectively. Refer to these chapters if you have purchased the licenses for the protocols.

Chapters 9 through 13 provide information about advanced features of the filer such as disk management, snapshots, filer backup, filer statistics, and troubleshooting.

Appendix A, "Time zone names,"contains a list of time zone names.

Appendix B, "Configuration Record," contains a form for recording filer information.

Appendix C,"Glossary," explains some of the technical terms used in this guide.

Audience

This guide is intended for system administrators who are familiar with operating systems that run on the filer's clients such as UNIX, Windows 95, and Windows NT. It also assumes that you, as the system administrator, are familiar with how the NFS, CIFS, and HTTP protocols are used for file sharing or transfers. It doesn't cover basic system or network administration topics such as ip addressing, routing, and network topology; it emphasizes the characteristics of the Network Appliance filer.

Terminology and conventions

When the term “filer” is used, the information applies to all Network Appliance filer models.

You can enter filer commands on either the system console or from any client computer that can access the filer through `telnet`.

In examples that illustrate commands executed on a UNIX workstation, this guide uses the command syntax of SunOS 4.1.x. The command syntax and output may differ depending on your version of UNIX.

When describing key combinations, this guide uses the hyphen (-) to separate individual keys. For example, “Ctrl-D” means pressing the “Control” and “D” keys simultaneously. Also, this guide uses the term “Enter” to refer to the key that generates a carriage return, although the key is named “Return” on some keyboards.

The following table describes typographical conventions used in the guide

Formatting convention	Type of information
Italic type	Words or characters that require special attention. File names. Placeholders for information you must supply. For example, if the guide says to enter the `arp -d` *`hostname`* command, you enter the characters "arp -d" followed by the actual name of the host. Man page names. Book titles in cross-references.
`Monospaced` font	Command and daemon names. Information displayed on the system console or other computer monitors. The contents of files.
`Bold monospaced` font	Words or characters you type. What you type is always shown in lowercase letters, unless it must be typed in uppercase letters to work properly.

Keyboard conventions

This guide uses capitalization and some abbreviations to refer to the keys on the keyboard. (The keys on your keyboard may not be labeled exactly as they are in this guide.)

When describing key combinations, this guide uses the hyphen (-) to separate individual keys. For example, Ctrl-D means holding down the Ctrl key while pressing the D key. In addition, this

guide uses the term Enter to refer to the key that generates a carriage return, although the key is named Return on some keyboards.

This guide uses the term “type” to mean pressing one or more keys on the keyboard. It uses the term “enter” to mean pressing one or more keys and then Enter.

Special messages

This guide contains special messages that are denoted by icons in the margin. They are described as follows:



NOTE: A note contains important information that helps you install or operate the system efficiently.



CAUTION: A caution contains instructions that you must follow to avoid damage to the equipment, a system crash, or loss of data.



WARNING: A warning contains instructions that you must follow to avoid personal injury.

Chapter 1: Introducing Network Appliance filers

The Network Appliance filer is a dedicated, special-purpose server that provides file service to network clients in the UNIX and Windows environments. Just as dedicated routers, bridges, and network printers provide their services more efficiently than general-purpose computers, the filer provides file service that is faster and more reliable than general-purpose servers.

Appliance philosophy

The word appliance implies a design philosophy. An appliance, whether in a household or on a network, performs one task and performs it well. It should be easy to install and, once installed, should require little attention. The Network Appliance filer is designed with this philosophy in mind. It has few configurable daemons, no user applications, and only one disk partition. Its design has eliminated most of the complexities of traditional servers, making the filer faster, easier to learn, and easier to manage.

Features and benefits

This section describes the features and benefits of the filer that make system administration easier than on traditional servers.

Multiprotocol support

The filer supports multiple file-sharing protocols and a file-transferring protocol. Clients using different operating systems can share data stored on a single filer. Having unified data storage makes system administration easier and less expensive.

Network clients running the UNIX or Windows operating system can share data on the filer by using these file-sharing protocols: NFS (Network File Service) and CIFS (Common Internet File System).

The filer also supports HTTP (Hypertext Transfer Protocol) for file transfers, which allows network clients using WWW (World Wide Web) browsers to display files that are stored on the filer.

The filer requires no emulation software to provide NFS, CIFS, and HTTP service. As a result, network clients of the filer experience shorter data access times and higher data availability.

Quick installation

Network Appliance designed the filer to make installation as fast and as simple as possible. Software is installed at the factory, and the default configuration is usable in most environments.

Fast system boot

Unlike conventional file servers, the filer has a file system that does not require time-consuming consistency checks. The filer takes from 30 seconds to three minutes to boot. Most of this time is used to spin up the disk drives and test system memory.

Single partition

All disks in a filer are combined to form a single partition, and all files belong to the same file system. As more disks are added, this single partition and the file system it contains get larger

automatically. Having one partition greatly simplifies disk management. It is not necessary to move files or directories from place to place as old disks fill up and new ones are added.

RAID, on-line rebuild, hot spare, and hot swap

The filer uses a Redundant Array of Independent Disks (RAID). This technique protects against disk failure by reserving one disk for parity. If one block on a disk fails, the filer uses the parity disk to reconstruct the data on that block, and the block is mapped to a new location on disk. If an entire disk fails, the parity disk prevents any data loss and allows the filer to continue running. When the disk is replaced either manually or automatically by a reserved hot spare disk, the filer uses the parity disk to reconstruct the replaced disk's contents automatically. The missing data is rebuilt in the background to minimize the interruption to file service.

The optional hot spare disk automatically replaces a file system disk if one fails. With hot swap support, the filer allows you to add, remove, or replace a disk while the system is running, which minimizes interruption to file system activity.

Snapshots

The filer uses snapshots to simplify file system backup. A snapshot is an on-line, read-only copy of the entire file system. Snapshots can be accessed by users as read-only files. This feature allows users to restore their own files without system administrator intervention. To ensure system security, files within a snapshot carry the same permissions as the original files.

Snapshots also allow the filer to be backed up to tape while the filer is used by clients.

System backup and restore

The filer's dump command backs up the file system. Because dump backs up a snapshot, the backup it creates is self-consistent, with no need to take the system off line. Another feature of the filer dump command is that it can split a dump file into multiple files on one or more tapes, allowing you to dump a subtree of the

file system that is larger than 2 GB to systems that limit tape files to 2 GB. You can use a filer's tape drive to dump and restore files located on another filer or on a UNIX system. The filer uses the rmt protocol for remotely accessing tape drives.

Additionally, the filer supports the Network Data Management Protocol (NDMP) Version 1 and Version 2. If you have a third-party backup application that uses NDMP, you can use that application to back up the filer to local tape devices, to tape devices attached to the workstation running the backup application, or to tape devices attached to a remote filer or workstation.

Disk quotas

You can impose disk usage quotas based on user or group identity, or on a particular tree of the file system.

NVRAM cache

As a filer receives requests, it logs them in non-volatile random-access memory (NVRAM). This feature improves system performance and prevents loss of data in case of a system or power failure. During an orderly shutdown, the halt and reboot commands flush the contents of NVRAM to disk.

Network protocols

The Network Appliance filer supports the following network protocols:

- NFS for clients running UNIX. The filer supports both NFS Versions 2 and 3, NFS over UDP, and NFS over TCP.
- CIFS for clients running Windows 95, Windows NT, and Windows for Workgroups.
- HTTP for use with clients that use web browsers.
- SNMP for network management.
- Domain Name Service (DNS) for name resolution.

- Network Information System (NIS) for password, group, and netgroup access.
- Windows Internet Name Service (WINS) for CIFS name resolution.
- Routing Information Protocol (RIP) for default gateway selection.
- Telnet for remote logins.
- Remote magnetic tape protocol (rmt) to remotely access a filer's tape drive.
- Remote shell (rsh) to remotely execute commands on a filer and to use rmt to remotely access tapes on other machines.
- NDMP for managing filer backups.



NOTE: The filer doesn't support protocols such as the Reverse Address Resolution Protocol (RARP) and Trivial File Transfer Protocol (TFTP), which are necessary for booting network clients (for example, diskless or dataless workstations). While you can keep these types of clients' file systems on a filer, you can boot these clients only from a boot server, not from a filer.

(PC) NFS

The filer supports the (PC) NFS daemon, an NFS authentication server for DOS, Microsoft Windows, Microsoft Windows NT, OS/2, and the Macintosh operating system clients that run the (PC) NFS protocol. The filer supports version 1 and version 2 authentication.

Chapter 2: **Getting started**

This chapter provides step-by-step procedures for:

- configuring the filer to provide file service for clients
- preparing clients for sharing files stored on the filer
- displaying the filer's on-line help files

In addition, this chapter provides information about the diskettes that are shipped with your filer.

Before you start

Before you proceed with the instructions in this chapter, verify that:

- you have installed the filer, disk drives, and system console according to the instructions in the hardware guides for your filer and disk shelves.
- you have physically attached the filer to a network according to the instructions in the hardware guide for your filer.

Fill out the filer configuration information form in Appendix B, "Configuration Record." You will need to supply the information on the form during setup. For explanations of the information required to fill out the form, refer to "Running the setup program" on page 10.

Setting up the filer

The following is an outline of the system setup procedure. These steps are described in greater detail in the sections that follow.

1. Turn on the system and allow the filer to boot.
2. Enter system configuration information in the `setup` session.
3. If the filer runs the CIFS protocol, follow these steps to prepare the filer for file sharing:
    a. Enter user information such as name, password, and group into the filer.
    b. If you are using a domain controller, create user accounts on the controller.



NOTE: This chapter doesn't provide complete information about modifying or creating user accounts on a domain controller. Obtain the information from the manual for your Windows NT server.

    c. Create access control lists (ACLs) for shares.
4. Verify the filer's network connections.
5. Prepare clients for obtaining access to the file system on the filer.
6. Store the diskettes and CD-ROM that are shipped with your system in a safe place.

Turning on the power and booting

Both the console and the LCD display messages indicating the status of the power-on self tests (POSTs). For more information on the startup messages, refer to the hardware guide for your filer model.

The filer loads the operating system and displays boot messages after running the POSTs.

Running the setup program

After the filer is booted for the first time, it runs the `setup` program, which displays a list of prompts for configuration information. Enter the same information at each prompt as you wrote down in the filer configuration information form in Appendix B, "Configuration Record."

This section provides an example of `setup`. Following the example are the explanations for the input fields. The explanations are in two sections:

- "General information" on page 12 describes input fields needed by filers regardless of the file-sharing protocol that has been licensed for the filer.
- "CIFS-specific information" on page 18 describes input fields needed by filers running CIFS.

Except for the password, your input for each prompt in `setup` is echoed on the screen. In the `setup` example below, bold face type represents user input. A prompt without user input means that the default has been accepted. To accept the default, press Enter.

Depending on your filer model and the protocols licensed for your filer, the actual messages displayed by `setup` might be different from those in the example.

Enter IP addresses and subnet masks in *d.d.d.d* format only (where *d* is a decimal number). Do not omit *d* even if it is zero (0).

The following example provides an overview of the setup process and shows the information that you need to supply. The values shown are examples only, and some of them are impossible.

```
Please enter the new hostname: toaster
Please enter IP address for Network Interface
e0:266.266.266.263
Please enter netmask [255.255.255.0]:
Please enter media type for e0 {100tx-fd, 100tx, tp} [tp]:
Please enter IP address for Network Interface
e1:266.266.266.276
Please enter netmask [255.255.255.0]:
Please enter media type for e1 {100tx-fd, tp-fd, 100tx, tp}
[tp]: 100tx-fd
Please enter IP address for Network Interface fi0:
266.266.266.286
Please enter netmask [255.255.255.0]:
Please enter IP address for Network Interface
a5:266.266.266.296
Please enter netmask [255.255.0.0]: 255.255.255.0
Please enter name of the default gateway []: MyRouterBox
Please enter IP address for MyRouterBox: 266.266.266.261
The administration host is given root access to the
filer's /etc files for system administration. To allow
/etc root access to all NFS clients enter RETURN below.
Please enter name of administration host []: adminhost
Please enter IP address for adminhost: 266.266.276.264
Please enter timezone [GMT]: US/Pacific
Do you want to run DNS resolver [n]: y
Please enter DNS domain name []: netapp.com
You may enter up to 3 nameservers
Please enter IP address for the first nameserver []:
266.266.266.265
Do you want another nameserver [n]:
Do you want to run NIS client [n]: y
Please enter NIS domain name []: netapp.com
New password:
Retype new password:
Add net default: gateway MyRouterBox
NFS server is running
Server is in operation with the following export points:
/home  -root=adminhost
/      -access=adminhost,root=adminhost
```

```
toaster> cifs setup
password:
Enable access to the filer by a Windows(tm) PC

Your filer is currently only visible to PCs on the same net.
Do you wish to make the system visible via WINS? [no]:y
Please enter next WINS IP address (or return to end
list):266.266.266.266
Please enter next WINS IP address (or return to end
list):266.266.266.267
Please enter next WINS IP address (or return to end
list):266.266.266.268
Please enter next WINS IP address (or return to end list):
Your filer is currently using UNIX style semantics
By default CIFS accesses use PC style semantics.
Do you want to enforce UNIX semantics instead? [no]:
The name of this filer will be 'TOASTER'.
  Do you want to modify this name? [no]:
Will this filer use an NT domain controller for authentication?
[yes]:
Enter the NT Domain for the filer:NT-DOMAIN
CIFS filer is beginning domain installation
CIFS server is registering...
CIFS server is beginning domain installation.
    CIFS server is locating PDC.
Thu Jul 24 18:08:44 PDT [rc]: CIFS - machine password changed on
DC NT-DOMAIN-PDC
Welcome to the NT-DOMAIN domain.
CIFS server is running.
toaster>
```

General information

The information in this section is required for all filers that run any file-sharing protocol unless specified otherwise.

Host name. The filer begins the setup process by asking for a host name for the filer:

```
Please enter the new hostname:
```

Enter a host name for the filer.

The host name is used as an alias for the first interface you set up (Ethernet interface e0). The filer automatically creates host names for additional interfaces based on the numbers of the interfaces. For more information about interface numbering and naming, refer to "Naming conventions for network interfaces" on page 77. For information about physical locations of the interfaces, refer to the hardware guide for your filer model.

IP address and mask. For each network interface, the filer asks for the IP address and netmask:

```
Please enter IP address for Network Interface e0:
Please enter netmask [255.255.255.0]:
```

Enter the IP address and subnet mask.

Media type. The filer then asks for the media type for each interface.

```
Please enter media type for e0 {100tx-fd, 100tx, tp}
[tp]:
```

For the Ethernet interfaces supported by the filer, you must specify the media type (or accept the default). Prompts display only the media types that are installed on your filer. They are described in Table 2-1.

Table 2-1. Ethernet interface media types

Media types displayed at the prompt	Description
100tx	100Base-T
100tx-fd	100Base-T, full-duplex
1000fx	gigabit Ethernet
auto	10Base-T / 100Base-TX Ethernet using autonegotiation

Table 2-1. Ethernet interface media types (continued)

Media types displayed at the prompt	Description
tp-fd	10Base-T, full-duplex
tp	10Base-T
thick	thick Ethernet
thin	thin Ethernet

Use "auto" when the interface that is connected to the filer interface supports autonegotiation and has 10BaseT or 100BaseTX, either half duplex or full duplex. If an interface on the filer connects to an interface that uses full duplex but does not support autonegotiation, you must explicitly specify full duplex.

The prompt for media type doesn't appear if the network card supports only one media type such as twisted pair or FDDI.

Default gateway. The filer asks for the name and IP address of the default gateway for routing:

```
Please enter name of the default gateway []:
MyRouterBox
```

Enter the name of the default gateway or, if the network that the filer is on is not connected to other networks by a router, press Enter because the filer does not need a default gateway.

If you enter the name of the default gateway, the filer asks for the gateway's IP address:

```
Please enter IP address for
MyRouterBox:266.266.266.261
```

Enter the IP address of the gateway whose name you entered in the previous step.

When the filer sets up the default route, it displays a message such as this one in the example:

```
add net default: gateway MyRouterBox
```

The routing daemon `routed` is turned on by default. For information about configuring `routed`, see "Routing" on page 119.

Administration host. The filer then asks for the name and IP address of the client computer for managing the filer:

```
The administration host is given root access to the
filer's /etc files for system administration. To allow
/etc root access to all NFS clients enter RETURN
below.
Please enter name of administration host []: adminhost
Please enter IP address for adminhost: 266.266.276.264
```

Enter the name and IP address of the administration host for the filer.

By default, the administration host is used as a mail host to send automatic email to Network Appliance Technical Support. Also, the host is added to the filer's */etc/hosts.equiv* and */etc/exports* file, and you can enter filer commands from the host by using `rsh`.

If the administration host is an NFS client, with root access you can mount the filer's root directory (/) on the host and edit the configuration files in the filer's */etc* directory from the host. You can edit configuration files as Administrator from any computer running CIFS as long as the filer is visible to that computer.

If you do not enter the administration host name and address, any NFS client has root access to the filer. Be sure to specify an administration host if security is a concern. For additional information about the administration host, see "Using the administration host" on page 62.

Time zone. The filer asks for a time zone:

```
Please enter timezone [GMT]: US/Pacific
```

The default is Greenwich Mean Time (gmt), which is the same as Universal Coordinated Time (UTC). Other time zone names are listed in Appendix A, "Time zone names."

DNS resolver. You can use DNS, NIS or both for name resolution. The filer asks whether you want to use DNS for name resolution:

```
Do you want to run DNS resolver [n]:
```

If you answer **y**, the following prompt appears:

```
Please enter DNS domain name[]:
```

Enter the name of a DNS domain, like *netapp.com*.

The filer then asks for the names and IP addresses of up to three name servers:

```
You may enter up to 3 nameservers
Please enter IP address for the first nameserver []:
Do you want another nameserver [n]:
```

Enter the IP address of a name server, then enter **y** if you want to enter the name of another one. Enter **n** to stop entering IP addresses.

NIS client. The filer asks if you want to run NIS:

```
Do you want to run NIS client [n]:
```

If you answer **y**, the following prompt appears:

```
Please enter NIS domain name[]:
```

Enter the name of an NIS domain.

For information about running DNS and NIS, see “Name resolution” on page 111.

Password. The filer uses a password for restricting access to the filer shell.



CAUTION: If you specify a password, the filer prompts for the password when anyone tries to connect to it by using the console or Telnet. Only users who know the password can access the filer. If you don’t specify a password during setup, the filer will not prompt for a password when somebody tries to gain access to it through the console or Telnet, and anybody can access the filer without a password. To maintain system security, be sure to create a password for each filer.

The filer asks for a password:

```
New password:
```

Enter a password. The password is not displayed as you enter it. There are no minimum length or special character requirements.

The following prompt then appears:

```
Retype new password:
```

Enter the password again.

The filer confirms a successful setup with the following messages:

```
Add net default: gateway MyRouterBox
NFS server is running
Server is in operation with the following export
points:
/home  -root=adminhost
/      -access=adminhost,root=adminhost
```

If your filer runs NFS only, `setup` is complete, and you can go to "Verifying network connections" on page 38 and proceed from there. If your filer runs CIFS, go on to "CIFS-specific information." If it runs HTTP, follow the instructions in "Preparing the filer for delivering HTTP service" on page 37.

Changing general information

If you want to change the general and NFS configuration information that you entered in the initial `setup` session, enter the `setup` command at the filer prompt. To make the changes take effect, reboot the filer.

CIFS-specific information

If your filer runs CIFS, it needs additional configuration information before you can start file sharing. The following list describes the input fields in `setup` that are for CIFS only.

CIFS preliminaries

Before you start the CIFS setup process, you must do the following:

- If you plan to use WINS servers, have their IP addresses handy.
- Decide whether you want to install the filer into an NT domain or into a workgroup. It is difficult to change this decision after you have completed the setup procedure.

  If a filer is a member of a workgroup, it is not compatible with NT4.0 Service Pack 3.0 unless each client machine adds a registry entry to re-enable plain-text passwords. For information about how to add the entry, consult the Microsoft Knowledge Base on the Web and refer to article Q166730, "With Unencrypted Password SP3 Fails to Connect to SMB Server" at:

  http://www.microsoft.com/kb/articles/q166/7/30.htm

Table 2-2 shows which authentication features are available to domains and which to workgroups.

Table 2-2. CIFS authentication features

Feature	Domain	Workgroup
Compatibility with NT 4.0 Service Pack 3 (encryption).	Y	N
Compatibility with NT 4.0 Service Pack 3 message signing.	N	N
Encrypted password sent to filer.	Y	N
/etc files determine UID and GID.	Y	Y
/etc files contain password.	N	Y
Single login.	Y	N
Change password remotely.	Y	Y (only with NIS)
Multiple net segments.	Y	N[a]

a. Unless "workgroup" is really an NT domain with an NT domain controller.

Installing the filer into a domain integrates the filer into your NT domain topology. You would install a filer into a domain if you wanted centralized administration. Installing the filer into a domain requires at least one Windows NT Domain Controller.

This section presents a method of adding the filer to a domain, but for alternative ways of installing a filer into a domain, consult the Microsoft Knowledge Base on the Web and refer to article Q140387, "Batch File Adds/Removes Machine Accounts in Server Manager" at:

http://www.microsoft.com/kb/articles/q140/3/87.htm

Installing the filer into a workgroup enables you to serve Windows-based machines while maintaining a UNIX base. If you install the filer into a workgroup, you can use the */etc/passwd* file or NIS for authentication.

- If you plan to use WINS servers, have their IP addresses handy.
- Make sure that you have a permanent domain or workgroup name. It is very difficult to change this decision after you have completed the setup procedure.
- If you are installing the filer into a domain, make sure that every user who may access the filer has a domain account.
- If you are installing into a domain, you can add the filer to the domain by following these steps:

1. Make sure that you have the "add workstation to domain privilege."
2. Make sure that you can connect to the Primary Domain Controller.
3. Run Server Manager.
4. Display the Computer menu.

5. Choose Add to Domain. The Add Computer To Domain window, shown in Figure 2-1, appears.

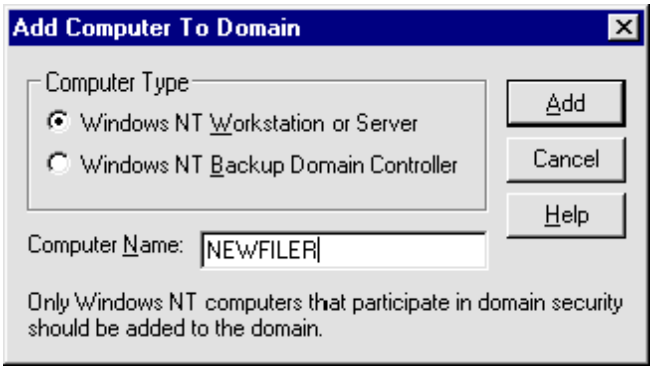


Figure 2-1. Add Computer To Domain window

6. Follow these steps to fill in the window information:

   a. Turn on the Windows NT Workstation or Server radio button

   b. Enter the name of the filer. It can have up to 15 characters and contain only these characters:

      letters (A – Z), numerals (0 – 9), !, #, $, @, %, ‘, (,), -, ^, _, {, }, and ~

   c. Click Add.

CAUTION: Do not use this account until you have completed setting up CIFS.

CIFS setup procedure

First, terminate CIFS with the `cifs terminate` command. For additional information about the `cifs terminate` command, consult the *na_cifs_terminate*(1) man page. When CIFS has been terminated, run the `cifs setup` command.

WINS server addresses. The `cifs setup` program begins by asking:

```
Your filer is currently only visible to PCs on the
same net.
Do you wish to make the system visible via WINS? [no]:
```

WINS stands for Windows Internet Naming Service, and links names and IP addresses within a network and among logical subnetworks. If you have a WINS server, answer "y."

If you answer "y," `cifs setup` asks for the IP addresses of WINS servers:

```
Please enter next WINS IP address (or return to end
list):
```

Enter the IP address of a WINS server; if you have entered all the WINS server addresses you want, press the Return or Enter key.

Permissions semantics. The `cifs setup` program then tells you whether it is using UNIX or PC-style permissions by displaying either:

```
Your filer is currently using UNIX style semantics
```

or:

```
Your filer is currently using PC style semantics
```

and asks:

```
By default CIFS accesses use PC style semantics.
Do you want to enforce UNIX semantics instead? [no]:
```

Answer as appropriate to specify the method that the filer uses to determine CIFS users' file access rights (read, write, and execute rights). The available methods are PC-style and UNIX-style. If your environment consists of CIFS users only, use PC-style (and accept the default). If both NFS and CIFS users share files on the

filer, read the following information to determine the method suitable for your site. The choice between PC-style and UNIX-style semantics affects CIFS users only. For further information about CIFS, see Chapter 7, "CIFS administration."

- PC style means that a user's rights to a file are determined by the rights assigned in the share's Access Control List (ACL) and limited by the UNIX permissions associated with the file. The following example illustrates how the filer determines whether you can write to a file named *file.1* in the share named *doc*:

  To be able to write to *file.1*, you must meet both of these requirements:

  – You have write permission for the *doc* share according to the ACL for *doc*.

  – You have write permission according to the UNIX-style permission bits associated with *file.1*. At the time a file is created, it inherits the UNIX-style permissions from the parent directory.

  The filer checks the UNIX-style permissions as follows:

  – If you are the file owner, the filer applies the owner's permissions.

  – If you are *not* the file owner, the filer applies the group permissions. For the owner, the filer disregards the group id (GID) associated with *file.1*. That is, the UNIX-style group permissions associated with *file.1* are applicable to everyone other than the owner.

- UNIX style means your read, write, and execute rights to a file depend on your user id (UID), group id (GID), and the UNIX-style permission bits associated with the file. The filer checks the UID, GID, and permission bits in the same way as a UNIX system. The filer only differs from a UNIX system in that the filer gives fewer rights to you if the rights specified by ACL are more restrictive than the UNIX-style permissions. The

following example illustrates whether you can write to a file named *file.1* in the share named *doc:*

If you are not the owner of *file.1*, you must meet the following requirements:

- You belong to a group that has write permission for *file.1*. The filer uses at least one GID of *file.1* and information in */etc/group* or the NIS group map to decide whether you are a member of the group.
- You have write permission for the *doc* share according to the share's ACL.

"Choosing the method for deciding access rights" on page 163 explains why two methods of maintaining file access rights are offered and what criteria to use when selecting a method.



NOTE: If you use UNIX-style permissions on the filer, set the share ACL to `everyone rwx`. With this setting, the UNIX rights will work just as they do on a UNIX system.

CIFS requires NIS or local */etc/passwd* and */etc/group* files because it uses the UID and GID for authentication.

Root password. If you are not running NIS, the filer asks for a root password. Root on the filer corresponds to Administrator in CIFS.

```
CIFS requires local /etc/passwd and /etc/group files.
Default /etc/passwd contains an entry for 'root'.
Enter the password for root:
```

Specify the password that the CIFS Administrator (root) account uses when trying to connect to the filer. The default */etc/passwd* file created after `setup` contains an entry for root with this password encrypted. For more information about the purpose of the */etc/passwd* file, refer to "Modifying /etc/passwd and /etc/group" on page 30.

```
Retype the password:
```

Enter the password again. If you did not retype the password exactly, the following message appears:

```
Mismatch - try again.
```

Type the password again.

Filer name. By default, the host name you specified for the filer earlier in `setup` is also used in CIFS connections. The filer asks for a name for the filer:

```
The default name of this filer will be FILER
Do you want to modify this name? [no]:
```

If you enter "no," the default name of the filer comes into effect. If you answer "yes," you are asked for a new name. Specify a new CIFS name if one or both of the following conditions are true:

- The host name exceeds 15 characters.
- The host name contains characters other than:

  letters A – Z, numerals 0 – 9, or the characters !, #, $, @, %, ‘, (,), -, ^, _, {, }, and ~

If the host name does not meet the CIFS requirements, the filer automatically creates a name by eliminating the illegal characters from the host name or by truncating the host name to 15 characters.

Although the filer displays the CIFS name in uppercase, the name is not case-sensitive.



NOTE: If you plan to make the filer part of a domain, make sure that the name for the filer is the same as the one with which you registered the filer in the domain. Once the filer has been added to the domain, if you change the filer's name, you must remove the old name from the domain, add the new name to the domain, and set up CIFS again.

The setup program verifies that no other workstation is using the name. If another workstation has the same name as the one you entered for the filer, you are asked for another name.

Authentication mode. The filer asks about the authentication mode. This will specify whether the filer is a member of a domain or a workgroup. The following message appears:

```
Will this filer use an NT domain controller for
authentication? [yes]:
```

If you enter "yes," the filer will be a member of a domain. Follow the instructions in "Becoming a member of a domain" below.

If you enter "no," the filer will be a member of a workgroup. Follow the instructions in "Becoming a member of a workgroup" on page 27.

Becoming a member of a domain. The filer asks you for a domain name:

```
Enter the NT Domain for the filer:
```

Enter the NT Domain name for the filer.



CAUTION: Make sure that the account whose name you entered has not been used. If it has, recreate it as described in "CIFS preliminaries" on page 18.

For a user to connect to the filer through a domain controller, all of the following conditions must be satisfied:

- The user must provide a login name and password that match the ones on the domain controller or one of its trusted domains.
- The user must either:
  - be included in the filer's */etc/passwd* file or, if NIS is being used, the NIS password map, or

– be a guest or generic user if there is a generic user account. The generic user account is for users who have an account in a trusted domain but do not have an entry in */etc/passwd* or NIS. For additional information about generic user accounts, see "cifs.generic_account" on page 85.

• The user must either:

– appear in the lists of users in the filer's */etc/group* file or the NIS group map for any groups to which they belong, or

– be a guest or generic user and the generic user account must appear in the filer's */etc/group* file or the NIS group map.

The filer displays the following confirmation messages:

```
CIFS filer is beginning domain installation
CIFS server is registering...
CIFS server is beginning domain installation.
    CIFS server is locating PDC.
Thu Jul 24 18:08:44 PDT [rc]: CIFS - machine password
changed on DC NT-DOMAIN-PDC
Welcome to the NT-DOMAIN domain.
CIFS server is running.
toaster>
```

where *nt-domain* is the name of your NT domain.

The setup process puts the file *.cifsconfig.dat* in */etc*.



CAUTION: Do not change the *.cifsconfig.dat* file because it contains information that the filer needs to communicate with the domain controller.

Becoming a member of a workgroup. The filer asks:

```
This filer will be a member of workgroup WORKGROUP
Do you want to modify this name? [no]:
```

If you enter "n," the filer will be a member of the workgroup *WORKGROUP*.

If you enter “y,” you are prompted for the name of a workgroup.

```
Enter the Workgroup for the filer:
```

The filer displays the following confirmation messages:

```
CIFS server is registering...
CIFS server is running.
```

This completes the setup procedure.

Changing names and domains

To change the name of a filer once it is installed into a domain or to change domains, you must synchronize the filer’s domain account with information on the filer.

> This section presents a method of changing the name of a the filer, but for alternative ways of doing so, consult the Microsoft knowledge and refer to article Q140387, “Batch File Adds/Removes Machine Accounts in Server Manager” at:
>
> http://www.microsoft.com/kb/articles/q166/7/30.htm
>
> While the methods presented there save time, they are be dangerous in that they might delete and active system.

To change the filer’s name, follow these steps:

1. Terminate CIFS on the filer with the `cifs terminate` command. For information about this command, consult the man pages.
2. Run Server Manager
3. Go to the old domain's Server Manager and display the icon for the filer.
4. The filer’s name must age from Network Neighborhood. Wait until the icon grays out, indicating that the filer is no longer active.

The change can take a long time to appear because of the nature of browsing. If it is taking too long for the icon to grey out, you can force the Master Browser on the filer's subnet to refresh its list and then do the same with the Browser on the filer's subnet. Consult NT documentation on Browsing for details on how to do this.

5. Delete the old account.

6. Create a machine account for the filer in the domain it will be a member of as described in "CIFS preliminaries" on page 18.

7. Run the `cifs setup` program to install the filer into a new domain or into the same domain as before, but with a new name.

   You can accept the default values, which are the values you set when you last ran the `cifs setup` program, until you get the prompt:

   ```
   This filer is a member of the NT-DOMAIN
   You will need to create a new computer account to
   change the name of the filer
   Do you want to delete the existing computer account
   [no]:
   ```

8. Continue the setup procedure from "Authentication mode" on page 26.

Preparing the filer for delivering CIFS service

This section describes how to prepare your CIFS filer so that the filer can authenticate users and determine file access rights. The preparation involves these steps:

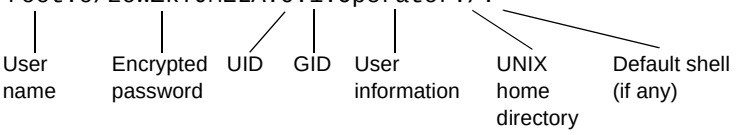
1. Modify the */etc/passwd* and */etc/group* files if necessary.
2. Assign rights to users for each share.

Modifying /etc/passwd and /etc/group

If you are using NIS for authentication, you do not have to modify the */etc/passwd* and */etc/group* files. The filer uses the NIS maps, and these are updated automatically.

After you run cifs `setup`, the */etc/passwd* file contains an entry for root. Fields in */etc/passwd* file entries are separated by colons. The following is an example of an */etc/passwd* entry:

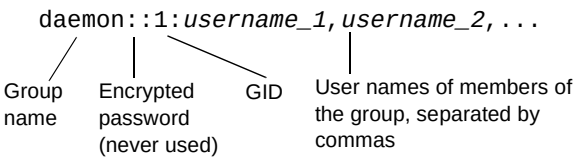
```
root:o/zoWZkYJMELA:0:1:Operator:/:
```





NOTE: Because the filer maps the user name “Administrator” to “root,” you can share files on the filer by using the user name “Administrator.” If you use a domain controller for authentication, use the password for the Administrator account. If you use */etc/passwd* for authentication, use the root password to log in as Administrator.

After you run `setup`, the */etc/group* file is also created. The following is an example of an */etc/group* entry:



The filer creates the following default entry in the */etc/group* file:

```
daemon:*:1:
```

The filer uses */etc/passwd* and */etc/group* in the following situations:

- When the filer authenticates a user. If the filer uses a domain controller for authentication, the domain controller verifies the user's login name and password, and the filer verifies that the user has an entry in */etc/passwd*. If the filer uses */etc/passwd* for authentication, the filer uses the encrypted password in */etc/passwd* to verify the user's login name and password. You can configure the filer to use the NIS password map to verify a user's name and password. For more information, refer to "Name resolution" on page 111.
- In determining guest access. For additional information about guest access, see "Guest access" on page 152.
- When the filer determines a user's file access rights when an attempt is made to access a file. For information about how the filer uses the files to determine access rights in an environment with both NFS and CIFS users, refer to "Choosing the method for deciding access rights" on page 163.

The following sections describe how to change these files depending on whether your filer uses a domain controller or */etc/passwd* for user authentication.

If you want to modify the configuration files from an NFS client, do so on the administration host. Otherwise, connect to the filer as Administrator from a CIFS client and edit the configuration files in the C$ share from the client. To learn about how to gain access to a share, refer to "CIFS clients" on page 40.

If the filer uses a domain controller for authentication

If you specify that the filer uses a domain controller for user authentication, users must have accounts on the domain controller and entries in the filer's */etc/passwd* file. Obtain information about creating accounts on a domain controller from the documentation about the system that is used as the domain controller. If you are using NIS and the user is already in the NIS password map, you do not need to edit the */etc/passwd* file for that user.

In */etc/passwd*, use a text editor to create a line for each user in the following format:

`username:password:uid:gid::`

The fields in the entry are:

username	CIFS user's name.
password	The filer disregards the field.
uid	Unique user id number.
gid	The user's group id number.

For example, the entry for the user "bjohnson" with UID of "12" and GID of "33" should read as follows:

`bjohnson:password:12:33::`

In */etc/group*, use a text editor to create a line for each group in the following format:

`groupname::gid:username_1,username_2,...`

The fields in the entry are:

groupname Unique group name.

password This field is never used and is blank.

gid Unique group id.

username Names of users who belong to the group.

For example, the entry for the group "marketing" with GID of "33" and "bjohnson," "bdole," and "jsmith" as members should read as follows:

```
marketing::33:bjohnson,bdole,jsmith
```

If the filer uses /etc/passwd for authentication

If you specify that the filer uses */etc/passwd* for authentication, users must have entries in the filer's */etc/passwd* file and they must appear in the lists of users for any groups to which they belong in the filer's */etc/group* file. Because the filer's */etc/passwd* file contains only one entry, and the filer cannot create additional entries, you must create a file of password entries and either copy the file to the filer or append the file to the filer's */etc/passwd* file, as described below.

Creating an /etc/passwd file. The following procedure describes how to create an */etc/passwd* file to include all CIFS users. Because the filer does not support the UNIX `passwd` command that creates encrypted passwords in */etc/passwd*, you need a UNIX workstation on which you are root for the following procedure:

1. On a UNIX workstation, back up the workstation's */etc/passwd* file by copying it (for example, by copying it to */etc/passwd.old*).

2. If there are CIFS users who are not included in the workstation's */etc/passwd* file, add the users as you normally would when adding users to a UNIX workstation. For

example, to add "jdoe" with UID "17" and GID "30," create an entry as follows:

```
jdoe::17:30:::
```

You assigned the numbers when you edited the */etc/passwd* and */etc/group* files to create the user and group as described earlier in this section.

3. Save the */etc/passwd* file.

4. If the CIFS users included in */etc/passwd* use their UNIX passwords for connecting to the filer, skip this step. Otherwise, use the `passwd` command to create an encrypted password for each user. The procedure is the same as the one you use when creating passwords for users of the UNIX workstation. For example, to create a password for jdoe, enter the following:

   ```
   unix_station% passwd jdoe
   ```

5. Rename the */etc/passwd* file to */etc/passwd.filer.*

6. Restore the backup copy of */etc/passwd* created in step 1. For example, rename */etc/passwd.old* to */etc/passwd* on the UNIX workstation.

7. Copy */etc/passwd.filer* from the UNIX workstation to a CIFS client from which you can connect to the filer.

8. Connect from the CIFS client to the filer as Administrator.

Transferring the /etc/passwd file to the filer. If */etc/passwd.filer* contains all CIFS users including root, copy */etc/passwd.filer* to the filer's */etc/passwd* file. Otherwise, append */etc/passwd.filer* to the filer's default */etc/passwd*, which already contains the root entry.



CAUTION: Make sure that the filer's */etc/passwd* does not include entries for users who should not be CIFS users of the filer. Use a text editor to remove such entries.

Modifying /etc/group. The following procedure describes how to modify */etc/group* to include all CIFS users:

1. Add entries to */etc/group* as you normally would when you create or change groups on a UNIX workstation.

   Each entry in the */etc/group* file is in this format and is for one group:

   groupname::*gid*:*username_1*,*username_2*,...

   The following is a sample entry for the group "marketing" with a GID of "42," and "jdoe," "msmith," and "jmanning" as members:

   marketing::42:jdoe,msmith,jmanning

   The filer disregards the password field in the entry (the field after the first colon). Delete any entries in */etc/group* that do not apply to the CIFS users of the filer.

2. Save the */etc/group* file.

Assigning rights to users for each share

Before a user can obtain access to a share, you must assign rights to the user or to the group of which the user is a member. The rights you assign are applicable to the entire share, and they are recorded in the share ACL.

Use the following command syntax to assign rights to a user or group:

`cifs access` *share* `[-g]` {*user*|*group*} *rights*

Rights are specified by three characters in the order read, write, and execute as follows:

- In the first position, "r" gives permission to read; "-" denies it.

- In the second position, “w” gives permission to write; “-” denies it.
- In the third position, “x” gives permission to execute; “-” denies it.

The following example assigns members of the group “marketing” read and execute rights to the *HOME* share:

```
cifs access HOME -g marketing r-x
```

The following example assigns members of the group “engineering” read, write, and execute rights to the *HOME* share:

```
cifs access HOME -g engineering rwx
```

The following example denies the user “cyoung” all rights to the *HOME* share:

```
cifs access HOME cyoung ---
```



NOTE: The `cifs access` command allows spaces for separating the rights.

The `setup` procedure for a filer running CIFS is now complete. If your filer also runs HTTP, go to the next section. Otherwise, go to “Verifying network connections” on page 38.

Preparing the filer for delivering HTTP service

If you have purchased the license for HTTP, follow these steps to configure the filer:

1. At the filer prompt, enter the following command to enable the HTTP daemon:

   `options httpd.enable on`

2. Use the following command syntax to specify the root of the path that HTTP requests can use:

   `options httpd.rootdir` *`pathname`*

   For example, if you want HTTP clients to gain access to files or subdirectories under */home/html_files*, enter:

   `options httpd.rootdir /home/html_files`

   Then, if a client requests *http://toaster/demo/1.html*, the filer retrieves */home.html_files/demo/1.html*.

3. Add the commands in steps 1 and 2 to the filer's */etc/rc* file so the commands remain in effect after each filer reboot.

For more information about managing a filer that works as an HTTP server, refer to Chapter 8, "HTTP administration."

Verifying network connections

After the setup session, verify the filer's network connections by using `ping` commands from a computer on a connected network. For example, if a filer called "toaster" has interfaces *e0* and *fi0*, enter the following commands:

```
ping toaster-e0
ping toaster-fi0
```

If the `ping` commands indicate that all network interfaces on the filer are "alive," go to the next section to prepare network clients so that they can share files stored on the filer.

If an interface does not respond to a `ping` command, make sure that:

- the interface is securely attached to the network.
- the media type is set up correctly if the interface is on an Ethernet card containing different media connectors.
- the routers function properly with correct routing information if `ping` is issued from a network not directly attached to the interface.

Preparing clients for file sharing

The procedure required for clients to share files depends on the file-sharing protocol used.

NFS clients

This section describes how to prepare an NFS client for mounting a file system from the filer. In the example provided in this section, the client runs SunOS, and the filer is the same as the one in the example in "Running the setup program" on page 10.

Editing the client's /etc/hosts file

The client needs to know the filer's IP address to mount files from the filer. If you do not use a name service, add to the clients */etc/hosts* file the name and IP address of the filer's various interfaces. The added entries might appear as follows:

```
266.266.266.263   toaster-e0 toaster
266.266.266.276   toaster-e1
266.266.266.286   toaster-fi0
266.266.266.296   toaster-a5
```

In the first line of the above example, the filer host name itself is used as an alias for the first network interface.



NOTE: The filer generates host names for interfaces by appending the number of the interface to the filer's host name. For more information on interface naming, refer to "Naming conventions for network interfaces" on page 77.

On a network using dns or nis, add the filer's interface names and IP addresses to the name server's database.

Editing the client's /etc/fstab file

Create an entry for the file system in the client's */etc/fstab* file so that the client can mount the file system automatically when it boots. The sample file system in the following procedure is the filer's */home* directory.

1. Create the directory */n/toaster/home* on the client.

2. To mount the filer's */home* directory on a mount point named */n/toaster/home*, add the following line to the client's */etc/fstab* file:

   ```
   toaster-e0:/home /n/toaster/home nfs rw 0 0
   ```

   In the example above, the client is on the same network as the filer's Ethernet interface *e0*.

   If the client is on the same network as Ethernet interface 1, use the following line instead:

   ```
   toaster-e1:/home /n/toaster/home nfs rw 0 0
   ```

3. The file system will be mounted when the client reboots. To mount the */home* directory without a client reboot, enter the following command:

   ```
   mount toaster-e1:/home /n/toaster/home
   ```

After the */home* directory on the filer is mounted, the user on the client system can create, modify, or remove files in */n/toaster/home*.

CIFS clients

After you configure the filer for CIFS service, the filer becomes visible to CIFS clients by automatically registering with the Master Browser on its local network. If cross-subnet browsing is configured correctly, the filer is now visible to all CIFS clients. For more information about cross-subnet browsing, refer to Microsoft networking documentation.



NOTE: You must have a Microsoft system such as an NT workstation or Windows 95 with a browser enabled. One of these must be on one of your directly-connected nets, otherwise the filer will not be visible but will still be accessible.

Table 2-3 describes the various methods by which CIFS clients can connect to the filer. For more information about commands and windows mentioned in the table, refer to the operating system manual for your computer.

Table 2-3. Connecting to the filer using CIFS

Client operating system	Command or window	Description
Windows 95, Windows NT 4.0	Map Network Drive window (from Network Explorer)	Connect to the filer as a network drive.
	Network Neighborhood	Click on the filer icon displayed in the Network Neighborhood window.
Windows NT 3.51, Windows for Workgroups	Connect Network Drive window (from the File Manager)	Connect to the filer as a network drive.
Any operating system that supports the DOS command line interface and supports the net use command	net use command	Map a share to a drive letter on the command line. For example: `net use f:\\toaster\home`

Getting on-line help from the filer

You can get detailed information about filer commands, daemons, and files from on-line help files. To view these files, use either the man command from a UNIX client or a web browser from any HTTP client.

Using the man command to view on-line help

UNIX-style man pages are available in the */etc/man* directory on the filer. They can be displayed by the UNIX man command. On the client, set your MANPATH environment variable to include the filer's */etc/man* directory. For example, if the filer's / directory is mounted on */n/toaster* on the administration host, include */n/toaster/etc/man* in your MANPATH variable definition.

Once the MANPATH variable has been defined, enter the man command on the NFS client to display any filer man page.

To distinguish the filer's man pages from UNIX man pages, precede the filer man page name with *na_* in the man command. For example, to display the man page for the filer's date command, enter the following:

```
man na_date
```

You cannot enter the command form the filer because the filer doesn't itself support the man command.

Using a web browser to view on-line help

If you have a World Wide Web (www) browser on a network client, you can view the on-line help files that are in HTML (Hypertext Markup Language) format.



NOTE: You can display these files whether or not HTTP service has been enabled on your filer.

To view the HTML help files, enter the following URL in your browser:

http://filername/na_admin/man/

where *filername* is the host name of the filer. The browser then displays the *index.html* file, which contains a list of help topics with hypertext links. Clicking on a topic displays the corresponding help file.

Getting help at the filer

You can get a brief description of what a command does by using the `help` command at the filer. The syntax is:

`help [ command ... ]`

where *command* is a filer command. An example of using the help command is:

```
toaster> help help
print summary of commands and help strings
```

Typing **`help`** by itself provides a list of filer commands:

```
toaster> help
?           arp         cifs        date
df          download    dump        exportfs
halt        help        hostname    httpstat
ifconfig    license     maxfiles    mt
ndmpd       netstat     nfs         nfsstat
options     passwd      ping        raid
reboot      quota       rdate       restore
route       routed      savecore    setup
shelfchk    snap        snmp        sysconfig
sysstat     timezone    uptime      version
ypwhich
```

Storing the filer diskettes

Your filer is shipped with several diskettes. Because system software has been installed at the factory, you don't need any of the diskettes to get started with a new filer. However, be sure to store these diskettes in a safe place because you might need them in the future. The following list describes the diskettes:

System files	This disk contains the system software. You need it to reinstall system software if necessary. Call Network Appliance Technical Support for instructions before reinstalling system software on your filer. The system files disks are shipped in two formats: One is to be read from a UNIX workstation, and one is to be read from a PC workstation.
System boot	If your system fails to boot from hard disk, or if you forgot your password, to fix the problem you must boot the system from the system boot disk.
Custom MIB	You use this Management Information Base with SNMP to get detailed information about filer operation. For further information, see "Using SNMP" on page 108.



NOTE: The Custom MIB diskette is readable only on a UNIX system.

After you store the diskettes in a safe place, go to one of the chapters described below for further information about how to use the filer.

Table 2-4. Chapter to read after you set up the filer

If the filer is for	Read
Both NFS and CIFS	Chapter 3, "Sharing files between NFS and CIFS users," which explains how the filer can be used by both NFS and CIFS clients simultaneously.
One file-sharing protocol (NFS or CIFS)	Chapter 4, "Routine filer administration."

Chapter 3: Sharing files between NFS and CIFS users

This chapter describes how the filer works with NFS and CIFS clients simultaneously. Because these two types of clients interact with a file server differently, it is important for you to understand how the read and write operations performed by one type of client affect the operations performed by the other type of client. This chapter discusses the following topics:

- CIFS and NFS file locks
- how the filer manages file locks when both NFS and CIFS users try to read or write to the same file
- how the filer supports CIFS access to symbolic links
- how NFS and CIFS use the per-file read-only bit
- how the filer manages names of files used by both NFS and CIFS users
- how much time the filer takes to convert directories that were created by NFS clients to be used by CIFS clients

This chapter does not discuss the differences between procedures that take place on network clients. For example, the way in which users gain access to files and directories depends on the NFS and CIFS protocols: NFS users *mount* the directories that have been

exported, and CIFS users *share* directories to which they have been given access in the ACL. For more information about these procedures, refer to the documentation for the clients' operating systems.

This chapter also does not discuss filer administration procedures for individual file-sharing protocols. The procedures such as rebooting the filer, recovering files from snapshots, and backing up the filer are different depending on whether there are CIFS clients at your site. The procedures are described in other chapters of this guide.

File-locking interactions

This section describes in general what happens when a client program using one protocol tries to read or write a file that is currently used by a program using a different protocol in an environment consisting of CIFS, (PC)NFS, and NFS users. For details about locking in a particular protocol, consult the documentation for that protocol.

There are two kinds of NFS clients: UNIX-based NFS clients and (PC)NFS clients. The (PC)NFS clients and CIFS clients behave in the same general way.

NFS locks are advisory, while CIFS locks are mandatory. CIFS applications depend on locking to behave properly. Because NFS locks are only advisory, file-manipulation operations (such as `rm`, `rmdir`, and `mv`) by a UNIX-based NFS client on a file opened by a (PC)NFS application can cause the application to crash.

Reads by UNIX-based NFS clients always succeed.

A deny-write operation causes a file-manipulation operation (such as `rm`, `rmdir`, and `mv`) on CIFS-accessed file by a UNIX-based NFS client to fail.

Byte-range locks work on portions of a file. Byte-range operations other than reads by a UNIX-based NFS client fail if the attempted operation is forbidden by the lock. As is appropriate for NFS, a UNIX-based NFS application might be forbidden to access a byte-range that is locked by CIFS.



NOTE: There is one exception to the enforcement of locks set by CIFS clients on the filer. When the filer runs the `dump` command, it ignores the file lock set by a CIFS client that prevents read access to parts of a file or the entire file. Ignoring the “read” file lock allows the filer to back up all files.

Managing symbolic links for CIFS access

CIFS clients can follow symbolic links, which are created by NFS clients. A symbolic link is a special file that points to another file or directory. A symbolic link is in some respects like a shortcut in the Windows environment.

There are two kinds of symbolic links: absolute and relative.

- Absolute symbolic links begin with a slash (/) and are treated as a path relative to the root of the file system.
- Relative symbolic links begin with a character other than a slash (/) and are treated as a path relative to the parent directory of the symbolic link.

You can control CIFS access to symbolic links in three ways:

- Enabling or disabling symbolic links
- Redirecting absolute symbolic links
- Preventing or allowing the following of symbolic links that can refer to a directory higher in the same tree

Enabling symbolic links

When the symbolic links for CIFS feature is enabled (the default), if the object being accessed by a CIFS client is a symbolic link (whether absolute or relative), the filer follows the link under the following conditions:

- The ultimate target is in the same share as the symbolic link.
- A symbolic link encountered in any path component other than the final one is always followed.
- The final component of a symbolic link is followed only if the operation is to open an existing file.

- Other operations, such as deleting and renaming, result in deleting or renaming the symbolic link itself rather than the target of the symbolic link.

Because CIFS client applications often perform operations such as writing to a temporary file, renaming the original file to a backup name, then renaming the temporary file to the original name, take care in using symbolic links whose ultimate target is a file (as opposed to a directory). If the original file were targeted directly by a symbolic link, this sequence of operations would have the result — unintended by the application — of the file being stored in the directory where the symbolic link was, and the renamed symbolic link pointing at the original file rather than to the updated file. For symbolic links to directories, this type of situation does not arise.

Because many PC applications work as described previously, if there are symbolic links that point to files, a PC could encounter such symbolic links. It is best to disable symbolic links for CIFS when there are symbolic links that point to files.

If you expect many files to be changed by applications that update files as described above, you might want to disable symbolic links for CIFS.

You enable and disable symbolic links with the `cifs.symlinks.enable` option. The option is On by default.

To disable symbolic links for CIFS, use the command:

```
options cifs.symlinks.enable off
```

To enable symbolic links for CIFS, use the command:

```
options cifs.symlinks.enable on
```

Redirecting absolute symbolic links

In a UNIX environment, the NFS client interprets the file system location represented by an absolute symbolic link. The CIFS client cannot do this. In a CIFS environment, the filer enables you to redirect absolute symbolic links on the filer. For example, you might want to redirect symbolic links pointing at the directory */u/users/charlie* to the directory */home/charlie* on the filer. You do so by specifying symbolic link redirection mappings in a text file named */etc/symlink.translations*.

The format of */etc/symlink.translations* is:

```
Map link target
```

where both *link* and *target* are absolute symbolic link pathnames.

For example, the entry:

```
Map /u/users/charlie/* /home/charlie/*
```

makes symbolic links pointing at the directory */u/users/charlie* point to the directory */home/charlie* .

Preventing symbolic link cycling

Because it is possible to create directory structures that are cyclic by creating a symbolic link that refers to a directory higher in the same tree (through use of a symbolic link having a “dot” or “dot-dot” component), a simple recursive descent of the tree will just go deeper and deeper until the maximum path length is reached. At that point an error is returned. So, for example, if you used Windows Explorer to search for files in such a cyclic directory, the same files would show up repeatedly.

The `cifs.symlinks.cycleguard` option controls whether symbolic links that might include a directory higher in the same tree are followed.

To eliminate the possibility of cyclic directory structures, make sure that the `cifs.symlinks.cycleguard` option is on (it is on by default) with the following command:

```
options cifs.symlinks.cycleguard on
```

If you use symbolic links having “dot” or “dot-dot” components and want the filer to follow the links, set the `cifs.symlinks.cycleguard` option to Off with the following command:

```
options cifs.symlinks.cycleguard off
```

When you list the contents of a directory, symbolic links that are valid references to files or directories are listed as if the target of the symbolic link existed in the directory. If the symbolic link cannot be expanded, it still looks like a file in a directory listing; however, any attempt by an application to open the link results in an access error.

NFS and CIFS use of the read-only bit

The filer, along with MS-DOS and Microsoft Windows, supports a per-file read-only bit that reflects whether a file is writable or read-only. This bit applies only to files and not to directories. NFS has no protocol operations that know about the per-file read-only bit. However, some software, when used both by NFS clients on UNIX systems and by CIFS clients on Microsoft Windows systems, requires that the read-only bit reflects whether the file is writable.

The following describes how NFS treats the read-only bit:

- Any file with the read-only bit turned on is treated, for all NFS operations, as if it had no write permission bits turned on.
- If a file has at least one write permission bit turned on and an NFS client turns off all write permission bits, the filer turns on the read-only bit for that file. (As described in the previous paragraph, a file with the read-only bit appears to an NFS client not to have any write permission bits turned on.)
- If a file has no write permission bits turned on and an NFS client turns on any write permission bit, the filer turns off the read-only bit for that file.
- If a file's read-only bit is turned on, and an NFS client attempts to find out the permission bits for the file, the actual permission bits for the file are not sent to the NFS client; instead, the filer sends the permission bits to the NFS client with the write permission bits masked off.
- If a file's read-only bit is turned on, and a CIFS client turns the read-only bit off, the filer turns on the owner's write permission bit for the file.
- Files with the read-only bit turned on are writable only by the super-user.

Whenever the read-only bit is turned on by a client (even if it was already on before the client did so), the filer keeps track of whether the client that turned on the bit was an NFS or CIFS client as follows:

- If the bit was turned on by a CIFS client, removing or renaming the file is *not* allowed, because file systems on MS-DOS and Microsoft Windows systems do not allow removing or renaming a file whose read-only bit has been set.
- If the bit was turned on by an NFS client that turned off all write permission bits, removing or renaming the file *is* allowed if the user has sufficient permission to do so, as file systems on UNIX systems allow removing or renaming a file that has no write permission bits set.

Naming files used by both NFS and CIFS

File naming conventions depend on both the network clients' operating systems and the file-sharing protocols. For example, file names are case- sensitive for clients running UNIX and are case-insensitive, but case-preserving, for clients running any Windows operating systems. This section discusses how different kinds of clients can obtain access to the same files despite different file naming conventions.

Maximum length of file names

On the filer, the maximum length of a file name is 255 characters for NFS clients and CIFS clients that support the PC's long file name format. Some CIFS clients, such as MS-DOS and Windows 3.*x* clients, support only file names in the 8.3 format (8 characters for the file name and 3 characters for the file name extension). In any directory that has access from a CIFS client, the filer creates and maintains two names: the original long name and an additional short name in 8.3 format. The filer generates the 8.3 name as follows:

1. It truncates the file name to six characters.
2. It appends a tilde (~) and a number or letter to the name. If it runs out of letters and numbers because there are too many similar names, it creates a unique file name that bears no relation to the original file name.
3. It truncates the file name extension to three characters.



NOTE: The number or letter appended to the short name ensures that the file name is unique. It is not for showing the order of file creation.

For example, if an NFS client creates a file named *specifications.html*, the short name created by the filer is *specif~0.htm*. If this short name already exists, the filer uses a different number at the end of the file name. For example, if the UNIX client creates another file named *specifications_new.html*, the short version of *specifications_new.html* is *specif~1.htm*.

The short names appear on clients that support only the 8.3 format. The short names are not visible to NFS clients. On Windows 95 and Windows NT clients, you can choose to display the short name or the long name by using File Properties.



NOTE: Under some circumstances, an application running on a client that uses names in 8.3 format can "lose" the file's original long-format name. This can occur as a consequence of the way an application saves a file that it has edited. Some applications rename the original file, then save the edited file as if it were newly created. The filer thus receives instructions to delete the original file and create a new one. When the client supports only 8.3 names, this new name will no longer have an equivalent in long format.

Legal characters used in file names

The characters that you can use in file names depend on the client operating systems. Because restrictions on legal characters vary from one operating system to another, refer to the documentation for your client's operating system for more information about prohibited characters.

When you name a file to be shared by users on different operating systems, it helps to use only characters that are common to both. For example, if you use UNIX to create a file and use a colon (:) as its file name, an MS-DOS user will see the name displayed as *~0* because the colon is an illegal character in an MS-DOS file name.

Case sensitivity in file names

Uppercase and lowercase characters are significant to NFS clients but not to CIFS clients. For example, if a file named *specifications* already exists, an NFS user can still save another file under the name *Specifications*, but a CIFS user will be instructed by the application to choose another file name. This section describes how both NFS and CIFS users can use file names that differ only by case.

When a client creates a file name, the filer preserves the case. For example, if a CIFS client creates *Spec.txt*, the file name is displayed by both CIFS and NFS clients as *Spec.txt*. If an NFS user later creates a file named *spec.txt*, NFS and CIFS clients will see the file names as follows:

- On NFS clients, one file is displayed as *Spec.txt* and the other as *spec.txt*. That is, the file names are displayed in the same way as they were created.
- On CIFS clients, even those that support long names, one file is displayed as *Spec.txt* and the other is displayed as *Spec~0.txt*.

Timing of directory conversion for CIFS access to NFS-created files

If you license both NFS and CIFS protocols, your filer supports access via either protocol to the *same* files. File-naming conventions differ between NFS and CIFS; although any CIFS-based file name is valid in NFS, some NFS names are not valid in CIFS. Also, CIFS maintains for each file certain attributes for which there are no NFS equivalents. A filer that runs CIFS maintains a directory in which NFS-style names are assigned equivalents in the 8.3 format.

If you add CIFS access to files that were created by NFS clients, the first time a directory is accessed via CIFS, the filer will automatically convert it to the format required to support CIFS clients. "Accessed" here includes listing the contents of a directory, for example, with the File Manager, Explorer, or DOS `dir` command, or accessing any files or directories contained in that directory.

Under some circumstances, the CIFS client may automatically look at the directory's subdirectories, thereby triggering conversion of the subdirectories as well. (For example, this would happen if the user has set the "Expand All" option under the File Manager "Tree" menu.)

Once a directory has been converted, it will not have to be converted again.

Although this process is automatic, the initial conversion of a directory can take considerable time, especially if the directory contains a large number of files. For example, conversion of a directory of 60,000 files takes the filer about 35 minutes on a NetApp F540, and about 2 hours on a NetApp F330. It is important to take these conversion times into consideration when deciding when to convert the directories.

While a directory is being converted, the filer might not be able to perform any other file system operations until the conversion is complete.

If there are portions of the directory tree that will *never* have CIFS access, there is no need to convert them. However, any future CIFS access to an unconverted directory will immediately trigger its conversion.

If you have a directory that contains more than 50,000 files, before triggering a conversion, use an NFS client to distribute files among a greater number of subdirectories. This speeds up the conversion process and avoids a possible crash.

Chapter 4: Routine filer administration

This chapter describes routine filer administration procedures that are needed regardless of the file-sharing protocols licensed for your filer. This chapter emphasizes the filer characteristics that distinguish the filer from a general-purpose server.

For information about advanced filer features, refer to Chapter 9, "Managing disks," and Chapter 10, "Snapshots."

Using the administration host

During the filer setup procedure, one client computer is specified as the administration host (named *adminhost* in the example in Chapter 2, "Getting started"). The following describes how you use the administration host:

- The filer grants root permissions to the host after setup. If the host is an NFS client, you can mount the filer's root directory and edit a configuration file from the host. (You can edit configuration files from any CIFS client as long as you connect to the filer as root. For more information about connecting from a CIFS client to the filer, refer to Table 2-3 on page 41.)

- The filer configures the administration host as the mail host for sending periodic automatic email messages to Network Appliance Technical Support about the status of your filer. For the host to send email, it must provide a server for the SMTP protocol, such as the `sendmail` program or Microsoft Exchange server. You can, however, designate another host at your site to be the mail host at any time. Refer to "Automatic email to Network Appliance" on page 96 for more information about how the filer sends automatic email messages. Refer to "Using the options command to configure autosupport" on page 98 for information about how to specify a mail host.

- The administration host is a trusted host from which you can enter filer commands by using `rsh`. The filer's setup procedure automatically includes the administration host in the */etc/hosts.equiv* file.

If you plan to manage the filer from an NFS client, use as the administration host, a computer that meets the following requirements:

- It supports a text editor that can display and edit text files containing lines ending with the newline character.

- It supports the `telnet` and `rsh` commands.

- It can NFS-mount directories from the filer.

If you plan to manage the filer from a CIFS client, use a computer that supports the `telnet` and `rsh` commands as the administration host.

Editing configuration files

Because the filer does not include a local editor, use an editor from a client to change configuration files in the filer's */etc* directory. The procedure for modifying configuration files is different depending on whether you edit the files from an NFS or CIFS client. If you use an NFS client, edit the files from the administration host. If you use a CIFS client, connect to the filer as Administrator. For more information about how to gain access to the filer from a CIFS client, refer to "CIFS clients" on page 40.

From an NFS client

To edit a configuration file, mount the filer's root directory on the administration host and edit the configuration files in the directory from the host.



NOTE: You can edit the */etc/exports* file from the administration host to give root permission to other clients. Changes take effect only after the `exportfs` command is invoked. After */etc/exports* has taken effect, you can mount the filer's root directory on these clients and edit the configuration files.

From a CIFS client

Follow these steps to edit configuration files from a CIFS client:

1. Connect from a CIFS client to the filer as Administrator. After the setup procedure, the default */etc/passwd* and */etc/group* files allow you to share files on the filer as Administrator.

2. On the CIFS client, display the contents of the filer's *C$* share and select the file to be edited. The file's root directory is shared automatically as *C$* after setup, and the root has read, write, and execute rights to the share. The *C$* share is a

"hidden" share; you can get to it only by specifying the path manually (for example, as *filer*\C$) rather than accessing it through the Network Neighborhood icon.

Obtaining access to the9 filer shell

The filer supports a command line interface that you can access directly from the system console or remotely using Telnet. The console and Telnet both provide access to a single session, with everything entered through Telnet being echoed at the console, and vice versa. Only one Telnet session can be open at a time. If you are using Telnet, to log out from the filer, press Ctrl-], then press Ctrl-D to log out of Telnet.

Although Telnet and the console share the same shell session, Telnet and the console will each ask you for a password. Both console and Telnet connections use the same password.

The filer also supports remote shell (`rsh`) connections with trusted remote hosts (those listed in */etc/hosts.equiv*), allowing you to execute some commands on the filer from a shell script or `crontab` file.



NOTE: You can use `rsh` only to invoke filer commands. You cannot use `rsh` to remotely log in to the filer. To log in to a filer from a host, use `telnet`.

Table 4-1 lists the filer commands that you can execute using a remote shell.

Table 4-1. Filer commands that can be executed with rsh

cifs	netstat	restore	uptime
df	nfsstat	route	version
dump	options	snap	ypwhich
exportfs	quota	snmp	
halt	raid	sysconfig	
httpstat	rdate	sysstat	
mt	reboot	timezone	



NOTE: Because the console and Telnet share the same session, a Ctrl-C entered remotely aborts whatever command is being run from the console and vice versa. To make sure that a command will not be terminated by an inadvertent Ctrl-C from another station, start the command from a trusted host through `rsh`.

The system password required for establishing a console or `telnet` connection with the filer is created during initial system setup. You can, however, change this password at any time with the `passwd` command, which prompts for the old password, if any, and then requests the new password twice.

For information about other ways to increase filer security besides password protection, use the options described in "Maintaining filer security" on page 103.

Halting and rebooting the filer

To reboot the filer, use one of the following methods:

- Enter the `halt` command. Proceed as follows depending on the prompt you get:
  - If the `mon>` prompt appears, enter the `b` command
  - If the `ok` prompt appears, enter the `boot` command.
- Enter the `reboot` command. The `reboot` command halts the filer and reboots it in one step.

If the filer has a boot diskette in its floppy drive, it boots from the floppy disk; otherwise, it boots from hard disk.



NOTE: The FAServer 450, FAServer 1300, or FAServer 1400 must have a disk at SCSI address 0.0 or 0.1 to boot.

The following are reasons why you should always execute the `halt` command before powering off the filer:

- `halt` flushes all data from nvram to disk, eliminating a potential point of failure.
- If your filer is a FAServer 450, 1300, or 1400, `halt` turns off nvram to conserve the nvram batteries after `halt` flushes all data to disk. Other filer models use lithium batteries for the nvram. These batteries cannot be turned off but last much longer than the ones used on the FAServers.
- `halt` avoids potential data loss on CIFS clients.

  When the filer is halted or rebooted, NFS client applications wait for the filer to resume NFS activity. Following reboot, they can resume without error (unless impatient users have canceled them). The effect on CIFS clients is different. If a CIFS client is disconnected from the filer, the users' applications are terminated, and changes made to open files since the last save are lost.

Therefore, never interrupt CIFS service without giving advanced warnings to users. Before powering off the filer, use the `halt` or `cifs terminate` command to send a warning message to CIFS users. In this way, the users can save files and exit applications within the time period that you specified before the actual shutdown. For more information about `cifs terminate`, refer to "Stopping CIFS service" on page 177.



NOTE: Clients using Windows 95 or Windows for Workgroups can display the CIFS shutdown messages only when the clients' `WinPopup` program is configured to receive messages. The ability to display messages from the filer is built into Windows NT.

Understanding the default configuration

The default configuration of a filer depends on whether the filer is running NFS, CIFS, or both, and NIS. While the default configuration is usable for small sites, it is probably not secure enough for large sites or for sites connected to the Internet. Subsequent sections within this chapter describe in greater detail how you might want to modify the default configuration to suit your needs.

For NFS users, the filer has two exported directories: the root directory (*/*) and the */home* directory.

For CIFS users, the filer has two shared directories: the root directory and the */home* directory. On CIFS clients, these directories are shown as the *C$* share and the *HOME* share. The *C$* share is a "hidden" share; you can get to it only by specifying the path manually (for example, as *\\filer\C$*) rather than accessing it through the Network Neighborhood icon.

The following sections describe the contents of the root and */home* directories.

The root directory

For NFS, the root directory is exported with root permission only to the administration host. From the administration host, you can modify the configuration files in */etc*.

For CIFS, you can share the root directory with read and write access on a CIFS client after connecting to the filer. For the connection, use "Administrator" as the user name and the root password specified during `setup`.

Table 4-2 lists the configuration files in */etc*. Some of the configuration files might not exist on your filer if it is used for CIFS or NFS only.

If you are using NIS, the filer uses the NIS group, password, and shadow maps instead of the */etc/group*, */etc/passwd*, and */etc/shadow* files.

Table 4-2. Configuration files in /etc

File name	Contents	File-sharing protocol
.cifs.cat	Domain information (only if the filer is a member if a domain)	CIFS only
cifsconfig.txt	CIFS commands that the filer has used for configuration	CIFS only
exports	NFS export points	NFS only
hosts	Known hosts and their IP addresses	All
hosts.equiv	Trusted hosts and users for `rsh`	All
group	CIFS group names, GIDs, and members' names	CIFS
netgroup	Network groups	NFS only
nsswitch.conf	The order in which the filer contacts name services.	All
passwd	Users' names, UIDs and primary GIDs	CIFS and (PC) NFS
rc	Script of commands to be executed when the filer is initializing	All

Table 4-2. Configuration files in /etc (continued)

File name	Contents	File-sharing protocol
serialnum	Filer serial number and license codes, which help Network Appliance Technical Support process automatic email sent by the `autosupport` daemon	All
shadow	Encrypted password strings and password aging information	CIFS and (PC) NFS



CAUTION: The filer automatically maintains the */etc/cifsconfig.txt* file. Any changes you make to this file will be lost after the next `cifs` command. If you edit this file, do so only after you have terminated CIFS service by using the `cifs terminate` command. Reboot the filer after you finish editing.

The /home directory

How the */home* directory is used depends on the file-sharing protocol used by the client.

For NFS clients

The */home* directory is exported with read and write permissions to all clients. Only the administration host has root access. A client can mount a directory on the filer as soon as the filer's host name is added to the client's */etc/hosts* file (or to the client's name server).

The svtx bit (also known as the *sticky bit*) is set on the */home* directory so that users cannot delete each other's files in that directory. This is a safe configuration for networks that use UNIX permissions to provide security for network-accessible files. You can use the `chmod` command from the administration host to clear the sticky bit.

For CIFS clients

The */home* directory can be shared by CIFS users as the home share immediately after `setup`. However, before users can read and write files in the home share, you must follow the procedures in "Preparing the filer for delivering CIFS service" on page 30.

By default, CIFS users can write and delete their own files in home. For more information about access rights, refer to "Choosing the method for deciding access rights" on page 163.

The /etc/rc file

The commands in */etc/rc* execute as the system boots to configure the filer. You can execute all the commands in */etc/rc* at the command line interface. However, if you want to make a change permanent or to make the commands take effect at boot time, you must update the */etc/rc* file.

Also, if your filer is licensed to run CIFS, for CIFS to be enabled, */etc/rc* must be present at filer reboot.

The /etc/rc default contents

The best way to understand the commands used in */etc/rc* is to examine a sample */etc/rc* file. The sample setup session shown in Chapter 2, "Getting started" would generate the */etc/rc* file shown in Figure 4-1.

```
#Auto-generated /etc/rc Fri May 30 14:51:36 PST 1997
hostname toaster
ifconfig e0 'hostname'-0
ifconfig e1 'hostname'-1
ifconfig fi0 'hostname'-fi0
ifconfig a5 'hostname'-a5
route add default MyRouterBox
routed on
options dns.domainname netapp.com
options dns.enable on
options nis.domainname netapp.com
options nis.enable on
timezone US/Pacific
snap sched 0 2 6@8,12,16,20
savecore
exportfs -a
nfs on
```

Figure 4-1. /etc/rc file



NOTE: No CIFS-specific information is entered in */etc/rc* as a result of setup. Do not add CIFS commands in */etc/rc*.

The following list explains the sample */etc/rc* file line by line:

```
hostname toaster
Sets the filer's host name.

ifconfig e0 `hostname`-0
ifconfig e1 `hostname`-1
ifconfig f0 `hostname`-f0
ifconfig a5 `hostname`-a5
```

Set the ip address for the filer's Ethernet, FDDI, and ATM interfaces with a default network mask. (The network mask is set during the setup session. See "General information" on page 12.) The arguments in single backquotes expand to "toaster" if you specify "toaster" as the host name during setup. The actual ip addresses are obtained from the */etc/hosts* file; you may prefer to enter ip addresses directly in */etc/rc*. If the specified network interface is not present, `ifconfig` issues an error message and has no other effect.

The actual interface names and numbers depend on the filer model. Refer to "Naming conventions for network interfaces" on page 77 for more information about interface names.

If you change the filer's host name, you must modify the */etc/hosts* file to substitute the new host name. If you don't, `ifconfig` fails.

To override the default network mask, explicitly specify the network mask in the `ifconfig` command after the host name. For example:

```
ifconfig e1 'hostname'-1 netmask 255.255.0.0
```

```
route add default LocalRouter 1
```
Specifies the default router. You can add route commands to */etc/rc* to set static routes for the filer. The network address for LocalRouter must be in */etc/hosts*.

`routed on`
Starts the routing daemon. See “Routing” on page 119 for more information about routing.

```
options dns.domainname netapp.com
options dns.enable on
```
These options set the DNS domain name and enable DNS.

```
options nis.domainname netapp.com
options nis.enable on
```
These options set the NIS domain name and enable NIS.

`timezone US/Pacific`
Sets the time zone. The argument to the `timezone` command specifies which file in the */etc/zoneinfo* directory describes the time zone you want. A list of time zone names is in Appendix A.

`snap sched 0 2 6@8,12,16,20`
Sets the schedule for automatic snapshot creation. The line in the example */etc/rc* file is the default setting, which creates and keeps no weekly snapshots, two daily snapshots (created at midnight every night), and six hourly snapshots (created at 8 A.M., 12 noon, 4 P.M.., and 8 P.M.). Chapter 10, “Snapshots,” describes snapshots in more detail.

`savecore`
Saves the core file from a system panic, if any, in the */etc/crash* directory. Core files are created only during the first boot after a system crash.

`exportfs -a`
Exports all directories specified in the */etc/exports* file. This command is included only if the filer runs NFS.

`nfs on`
Turns on NFS file service. This command is included even if NFS is not licensed for your filer. When the filer runs this command from */etc/rc* on a filer without an NFS license, the command fails and these messages appear:

```
NFS service is not licensed.
(Use the “license” command to license it.)
NFS server is NOT running.
```

`dns on`
Turns on the Domain Name Service.

`nis on`
Turns on the Network Information System.

Editing /etc/rc

To update or make changes to */etc/rc*, follow the instructions in “Editing configuration files” on page 63. Network Appliance recommends that you make a backup copy of the file before you start editing. After you finish editing, reboot the filer. The reboot tests the new configuration immediately.

Changing SNMP commands in /etc/rc

Regardless of whether snmp is enabled at your site, add these commands to */etc/rc*:

- `snmp contact “your email and telephone number”`
- `snmp location “location of your filer”`
- `snmp init 1`

for example:

```
snmp contact “jdoe@abc.com 415-555-1212”
snmp location “ABC corporation, engineering lab”
snmp init 1
```

The information provided by these commands allows Network Appliance Technical Support to contact you after receiving automatic email messages about your filer from your site. For information about automatic email messages, refer to “Automatic email to Network Appliance” on page 96.

Naming conventions for network interfaces

For information on how network interfaces are numbered, refer to the hardware guide for your filer model. This section discusses how software refers to each network interface.

When you write or modify shell scripts that involve interface names, remember that different interface naming conventions are used for different filers:

- For filers based on the EISA bus architecture, Ethernet interface names start with the string *ns* followed by the interface number (for example, ns0 and ns1). FDDI interface names start with the string *fi* followed by the interface number (for example, *fi0* and *fi1*).

  EISA-based filers automatically create host names for network interfaces by appending the number of the interface to the filer's host name. For example, if the filer is named *toaster*, the name for ns0 is *toaster-0*. If the interface is an FDDI interface, an "f" precedes the interface number. For example, the name for fi0 is *toaster-f0*.

- Filers based on the PCI bus architecture use the following interface naming conventions:

  – Ethernet interface names start with the letter *e* followed by the slot number (for example, *e0*) and the port number if the adapter card is a quad-port Ethernet card (for example, *e1a* and *e1b*).

  – FDDI interface names start with the letter *f* followed by the slot number (for example, *f1* and *f2*).

  – ATM interface names start with the letter "a" followed by the slot number (for example, *a5*).

  PCI-based filers automatically create host names for network interfaces by appending the slot number of the interface to the filer's host name. For example, if the filer is named *toaster*, the host name for e0 is *toaster-0*, and the host name for *e1a* is *toaster-*

1a. If the interface is an FDDI interface, the letter *f* precedes the slot number. For example, the host name for *f1* is *toaster-f1*.

Recovering from configuration errors

Because the filer doesn't have a local text editor, recovering from certain configuration errors may require more steps than simply modifying the configuration files.

For example, because of an incorrect ip address in */etc/rc*, the filer might have become inaccessible to clients. To solve this problem, enter commands on the console to configure the interface with the correct address, export the configuration file to the administration host, edit the file from the host, and reboot the filer.

If configuration errors render the hard disk unbootable, try booting from the floppy disk. For more information, see "Recovering from configuration file problems" on page 328.

Core files

The filer might crash if there is a hardware or software failure. On the first boot after a system crash, the filer stores the core file in */etc/crash*. The `savecore` command, which is included in */etc/rc* by default, produces two core files, *core.n-small* and *core.n* (the *n* in the file name is a number). The *core.n-small* file contains everything except the file system buffers. The *core.n* file includes additional information that might be essential for determining the cause of the crash and is often necessary for investigating problems involving the file system.

The `savecore` command also displays a message on the system console and logs a message in */etc/messages*.

If you have followed the instructions for system setup in Chapter 2, "Getting started," your filer will send email automatically to Network Appliance Technical Support upon each system reboot. However, if you disabled automatic email, contact Network Appliance Technical Support when your system creates a core file.

Message logging

The filer has a message logging daemon, `syslogd`, and a configuration file, */etc/syslog.conf*, which specifies how system messages are logged. Based on their severity and origin, system messages can be directed to the console, to a file, or to a remote system. By default, all system messages (except for those of severity level "debug") are written to the console and to the file */etc/messages*.

The */etc/syslog.conf* file consists of lines with two tab-separated (not space-separated) fields of the form:

`facility.level;facility.levelaction`

The following list describes each parameter:

facility

The `facility` parameter specifies the subsystem from which the message originated. It is one of the following keywords:

`auth`	messages from the authentication system, such as login
`cron`	messages from the internal cron facility
`daemon`	messages from filer daemons, such as rshd
`kern`	messages from the filer kernel
*	An asterisk specifies all facilities.

level

The `level` parameter describes the severity of the message. The level is one of the following keywords (higher to lower severity):

`emerg`	panic condition that causes a disruption of normal service

`alert`	condition that you should correct immediately, such as a failed disk
`crit`	critical conditions, such as disk errors
`err`	errors, such as those caused by a bad configuration file
`warning`	conditions that might become errors if not corrected
`notice`	conditions that are not errors, but might require special handling
`info`	information, such as the hourly uptime message
`debug`	used for diagnostic purposes (suppressed by default)
*	An asterisk specifies all levels.

action

The `action` parameter specifies the place to forward the received message when it originates from the specified facility and is of the specified level or higher. This field can be one of the following:

- the pathname of a file beginning with a slash (for example, */etc/messages*)
- a remote host beginning with the @ sign (for example, @adminhost)
- the console (*/dev/console* or an asterisk)

To summarize: if a message matches the specified `facility` and is of the specified `level` or higher, the message is forwarded to the place specified in the `action` field. For example, if you specify `err.kern` and `/etc/messages` on a line, all kernel messages of levels `emerg`, `alert`, `crit`, and `err` are forwarded to the */etc/messages* file.

Every Sunday night at midnight, the file */etc/messages* is copied to */etc/messages.0*, the file */etc/messages.0* is copied to */etc/messages.1*, and so on. The system saves messages for up to six weeks.

You should check the */etc/messages* file once a day for important messages. You can automate checking this file by creating a script on the administration host that periodically searches */etc/messages* and then alerts you.

Figure 4-2 shows an example of a customized */etc/syslog.conf*:

```
# Log anything of level info or higher to /etc/messages.
*.info                          /etc/messages

# Log all kernel messages, and anything of level err or
# higher to the console.
*.err;kern.*                    /dev/console

# Log all kernel messages and anything of level err or
# higher to a remote loghost system called adminhost.
*.err;kern.*                    @adminhost

# Log messages from the authentication system of level notice
# or higher to the /etc/secure.message file. This file has
# restricted access.
auth.notice                     /etc/secure.message
```

Figure 4-2. Customized /etc/syslog.conf

For more information about *syslog.conf*, see the *na_syslog.conf*(5) man page.

Configuring filer options

When used interactively, the `options` command changes the filer's behavior temporarily; the system returns to the default state when rebooted. To make changes permanent, you *must* include the `options` commands in the */etc/rc* file.

The syntax of the `options` command is as follows:

`options [ option value ] ...`

For example, the command to enable the `pcnfsd` daemon is:

`options pcnfsd.enable on`

The values for on and off are not case-sensitive. If no parameters are specified in the `options` command, the command prints the current values of all available options.

Options are described in detail in other sections of this guide and in the *na_options*(1) man page. The sections that follow provide a brief description of each option grouped by function. The default value is below the option name. The following applies to default values listed:

- "None" means that there is no default value.
- If the default is "on," the other possible value is "off."
- If the default is "off," the other possible value is "on."

Autosupport options

These options control whether and how the filer sends automatic status messages. For further information about autosupport, see "Automatic email to Network Appliance" on page 96.

autosupport.doit
None (default)

Immediately sends an email message describing the status of the filer. A word entered as the value for the option is sent in the notification subject line and should describe the reason for the notification.

autosupport.enable
On (default)

This option enables the `autosupport` daemon, which sends automatic email messages to report the status of the filer.

autosupport.from
autosupport (default)

Specifies the sender of the automatic email. It should be the email address of the person whom Network Appliance contacts after receiving an automatic email message.

autosupport.mailhost
administration_host (default)

Specifies the mail hosts that receive automatic email messages. Use a comma-separated list with no spaces.

autosupport.noteto
None (default)

Specifies up to five recipients of the automatic short email. Use a comma-separated list with no spaces.

autosupport.to
autosupport@netapp.com (default)

Specifies up to five recipients of automatic autosupport email messages. Use a comma-separated list with no spaces.

CIFS options

You use these options to control CIFS features on the filer.

cifs.generic_account
None (default)

When `cifs.generic_account` is set to the name of an account, users who have an account in a trusted domain but not an entry in */etc/passwd* or NIS can use that account. All generic users have the same rights to the generic users' account.

cifs.guest_account
None (default)

When `cifs.guest_account` is blank, a CIFS user can log onto the filer without an account in the password database, provided a domain controller authenticates the user.

When `cifs.guest_account` is set to the name of an account in the NIS passwd map or */etc/passwd* (typically **`guest`**), guest access to the filer is enabled, and the user has the UNIX user ID and the group ID of the guest account. For further information, see "Guest access" on page 152.

cifs.home_dir
None (default)

This option specifies the complete path name of the "homes directory." The directories under this path should have the names of users as their names. When a CIFS user connects to the filer and there is a directory name that exactly matches the user's name, the user sees a share of that name (truncated to 12 characters) that is the user's home directory. Only the user can access the home directory using this share. All other users are denied access. For further information, refer to "Creating a directory for a CIFS user" on page 154.

cifs.oplocks.enable
On (default)

When this option is on, the filer allows clients to use oplocks (opportunistic locks) on files. Oplocks are a significant performance enhancement, but have the potential to cause lost cached data on some networks with impaired reliability or latency, particularly wide-area networks. In general, this option should be disabled only if there are problems with databases and to isolate problems.

cifs.scopeid
(blank) (default)

Specifies a second element for a single-element NetBIOS computer name. This element is case-sensitive. You use this option to isolate a group of computers on a network that communicate only with other computers with the identical NetBIOS Scope ID.

Microsoft recommends that you not use NetBIOS Scope IDs.

Do not use this option if you are using DNS on your network.

For further information, refer to "Using NetBIOS scope IDs" on page 174.

cifs.symlinks.cycleguard
On (default)

If an object being accessed by a CIFS client is a symbolic link, the cifs.symlinks.cycleguard option, when set to ON, eliminates the possibility of cyclic directories by preventing the following of symbolic links that contain the "dot" (".") or "dot-dot" ("..") component — symbolic links that could refer to a directory higher in the same tree. With the cifs.symlinks.cycleguard option set to Off, if you are careful, you can use symbolic links having "dot" or "dot-dot" components. For additional information about this option, refer to "Preventing symbolic link cycling" on page 52.

cifs.symlinks.enable
On (default)

When cifs.symlinks.enable is set to On (the default setting), if the object being accessed by a CIFS client is a symbolic link, the filer follows the link with the condition that the ultimate target turns out to reside within the originating share (thus ensuring that the client has access permission to the target). This applies both to relative symbolic links (links to paths beginning with a character other than "/" and treated as a path relative to the parent directory of the symbolic link) and absolute symbolic links (links to paths beginning with "/" and treated as a path relative to the root of the file system). For additional information about this option, refer to "Managing symbolic links for CIFS access" on page 50.

DNS options

These options control how the filer works with DNS. For further information about DNS, see Chapter 5, "Network administration."

dns.domainname

None (default)

Sets the DNS domain name to the specified domain name.

dns.enable

Off (default)

Enables the DNS client on the filer. Before you enable DNS, you must set the DNS domain and the */etc/resolv.conf* file must exist.

HTTP options

These options enable and control HTTP services. For further information about HTTP on the filer, see Chapter 8, "HTTP administration."

httpd.admin.enable

On (default)

Enables http access to the filer's on-line help files and other files used by FilerView.

httpd.enable

Off (default)

Enables the http server.

httpd.log.max_file_size

2147483647 (2 GB - 1) (default)

Specifies the number of bytes the HTTP log file, */etc/log/httpd.log*, can grow to. The maximum value is 4294967295 (4 GB - 1).

httpd.rootdir
None (default)

Specifies the root directory containing files and directories that can be transferred to clients by the http protocol.

httpd.timeout
900 (15 minutes) (default)

Specifies the minimum amount of time (in seconds) before an idle HTTP connection times out.

httpd.timewait.enable
On (default)

When this option is set to On, the filer will drop an HTTP connection one minute after it has been closed by the client. When this option is set to Off, the connection is not dropped, and resources are consumed until the connection times out.

NFS and (PC) NFS options

You use these options to enable and control the Network File System. For further information about NFS, see Chapter 6, "NFS administration."

df_2gb_lim
Off (default)

Limits to 2 gb the response the filer gives to requests regarding total space, free space, or available space. This option is necessary for some NFS clients to calculate the amount of free space accurately. Without this option, a file system with more than 2 GB of free disk space may appear to be full to the client that initiates the "file system statistics" request.

nfs.big_endianize_fileid
Off (default)

When enabled, puts the file ID in the file handle reported by the server in big-endian form. This helps improve performance on directory lookup operations on certain clients that use the file ID within the file handle as a hash key.



NOTE: If you toggle this option, all file handles that clients currently have will become stale, so to use the filer as they did before the toggle, all clients will have to remount everything they have mounted from the filer.

nfs.per_client_stats.enable
Off (default)

Specifies whether the filer collects and displays NFS statistics from individual clients.

nfs.tcp.enable
Off (default)

Specifies whether the filer supports NFS over TCP. Enable this option if a client has problems using NFS over UDP.

nfs.v3.enable
On (default)

Specifies whether the filer supports NFS Version 3. Disable this option if a client has problems using NFS Version 3 and that client cannot be configured to use NFS Version 2.

nfs.webnfs.enable
off (default)

Turns webNFS on and off.

nfs.webnfs.rootdir
None (default)

The specified directory becomes the root or public directory for WebNFS. When a request specifies a relative path, lookups for files are done with respect to this directory.

nfs.webnfs.rootdirset
FALSE (default)

When set to TRUE, this option sets the directory specified in the `nfs.webnfs.rootdir` option to be the WebNFS root or public directory.

pcnfsd.enable
Off (default)

Specifies whether the filer runs the (PC) NFS server.

pcnfsd.umask
022 (default)

The default is 022, which means that users can read and write their own files, but that members of their group and others can only read the files.

NIS options

These options control how the filer works with the Network Information System. For further information about NIS, see Chapter 5, "Network administration."

nis.domainname
None (default)

Sets the NIS domain to the specified domain name.

nis.enable
Off (default)

Enables the NIS client on the filer. You must set the NIS domain before you enable NIS.

RAID options

These options control how the filer uses RAID. For additional information about RAID on the filer, see Chapter 9, "Managing disks."

raidtimeout
24 (default)

Sets the time in hours (from one through 24) that the system runs in degraded mode before an automatic shutdown.

raid.reconstruct_speed
4 (default)

Specifies the speed of raid reconstruction. The speed ranges from 1 (slowest) to 10 (fastest). The filer uses the number to determine the percentage of CPU time used for raid reconstruction.

raid.scrub.enable
On (default)

Specifies whether filer performs raid scrubbing.

Snapshot options

These options control snapshots. For additional information about snapshots, see Chapter 10, "Snapshots."

nosnap
Off (default)

Temporarily disables automatic snapshots.

nosnapdir
Off (default)

Makes invisible the snapshot directory that's normally present at the client mount point or at the root of the CIFS share. It also turns off access to the snapshot directory and all snapshot directories under the mount point or the root of the CIFS share.

After you toggle this option, you may not notice the effect immediately because the information about the snapshot directories might still be in the client's attribute cache. To force the change to take effect immediately, unmount and remount the file system.

Miscellaneous options

These options control additional aspects of filer operation.

ip.path_mtu_discovery.enable
On (default)

Enables or disables path MTU discovery, which is currently used only by TCP. When enabled, it allows the filer to discover and use the largest packet size that the filer can send to another host without fragmenting a packet. This means that the filer doesn't have to limit itself to sending many small packets, which takes more time and resources than sending fewer large packets.

If you cannot establish a connection, set this option to Off.

minra
Off (default)

Configures the filer to perform minimal read ahead. By default, the option is disabled and the filer does aggressive read ahead.

mount_rootonly
On (default)

Configures the filer to give mount privilege to root using only privileged ports (ports 1 through 1024).

no_atime_update
Off (default)

Prevents the update of the access time (atime) on an inode when a file is read. This option prevents inode updates from contending with reads from other files. Use it only on a filer with extremely high read traffic (for example, on a news server used by an Internet access provider or on a filer used mainly as an http server).

root_only_chown
On (default)

Allows only the root user to change the owner of a file. When the option is disabled, the owner of a file can change its ownership without being root. By default, this option is enabled.

When a non-root user changes the owner of a file, the set-user-id and set-group-id bits are cleared. If a non-root user tries to change the owner of a file but the change would cause the file's recipient to exceed his or her quota, the attempt will fail.

rps.status
None (default)

When set to **good**, this option clears the message that a power supply in the RPS (redundant power supply) is bad. Set this option to **good** after you have replaced a failed power supply in the RPS or else you will keep getting the message even though the power supply is no longer bad. With no value, this option reports the status of the filer's redundant power supply.

telnet.hosts
All hosts (default)

Specifies a list of hosts that can log in to the filer using Telnet. You can limit Telnet access to up to five specified hosts. The hosts should be listed in a comma-separated list. You can disable Telnet for all hosts by specifying a hyphen (-).

wafl.maxdirsize
10240 (default)

Sets the maximum size (in kilobytes) of a directory file. A directory file with a size of 10240 kilobytes can hold about 300,000 files.

Automatic email to Network Appliance

This section describes the `autosupport` daemon, which controls how automatic email messages are sent from your site to Network Appliance Technical Support.

How the autosupport daemon works

The `autosupport` daemon is enabled by default on the filer. The daemon triggers automatic email messages to members of Network Appliance Technical Support, alerting them to potential filer problems. If necessary, Network Appliance will contact you at the email address that you specify to help you resolve a potential system problem. Refer to "Using the options command to configure autosupport" on page 98 for more information about specifying the email address and other options.

Because the filer doesn't function as a mail host, it relies on another host at your site that listens on the SMTP port (25) to send mail. Therefore, `autosupport` requires at least one host reachable by the filer that runs an SMTP server or a mail forwarder such as the `sendmail` program or Microsoft Exchange server. By default, the administration host defined during system setup is used as a mail host. You can specify additional mail hosts.

The mail host sends email to Network Appliance Technical Support about your filer after any of the events listed in Table 4-3.

You can specify up to five addresses of email recipients. The default recipient is Network Appliance Technical Support. For receiving automatic messages, the email address of Network Appliance Technical Support is *autosupport@netapp.com*. Refer to "Using the options command to configure autosupport" on page 98 for more information about `autosupport`.

Table 4-3. Events that generate automatic email

Event	Subject line of the email message
Low NVRAM lithium battery[a]	BATTERY_LOW!!!
Disk failure	DISK_FAIL!!!
A disk scrub occurred	DISK_SCRUB!!!
Fan failure[a]	fan_fail!!!
Shutdown because of overheating[a]	OVER_TEMPERATURE_SHUTDOWN!!!
Partial redundant power supply failure[b]	POWER_SUPPLY_DEGRADED!!!
System reboot	reboot
Disk shelf error	shelf_fault!!!
Weekly backup of */etc/messages*	weekly_log
`option autosupport.doit` command	The string specified in `option autosupport.doit`

a. This email message is not sent by the FAServer 1300, 1400, and 450.
b. This email message is sent only by a filer that uses a redundant power supply. It means that one of the two power supplies has ceased functioning.

Contents of automatic email messages

Each email message generated by autosupport contains the following types of information:

- date and time stamp of the message
- NetApp software version
- system id of the filer

- host name of the filer
- software licenses enabled for the filer
- snmp contact name and location (if specified in */etc/rc*)
- output of the following commands (some are applicable only to the licensed protocols):

```
sysconfig -v
options
ifconfig -a
nfsstat -c
cifs stat
cifs sessions
cifs shares
httpstat
df
df -i
snap sched
sysconfig -r
```

- contents of */etc/messages*
- contents of */etc/serialnum*

Using the options command to configure autosupport

To change the default behavior of the autosupport daemon, use the options command. As with other filer commands, you can add options commands to the /etc/rc file if you want to execute them automatically when the filer reboots.

Disabling or enabling the autosupport daemon

The autosupport daemon is enabled by default. The syntax of the command to disable or enable the daemon is:

```
options autosupport.enable on|off
```

For example:

```
options autosupport.enable off
```

Specifying mail hosts

The command to specify hosts that send autosupport email messages to Network Appliance Technical Support is:

```
options autosupport.mailhost hostname,...
```

You can specify up to five mail host names. Separate names by commas and do not include spaces in the list. (Default is the administration host.) For example:

```
options autosupport.mailhost host1,host2,host3
```

Specifying addresses for autosupport mail

The command for specifying the recipients of automatic email messages sent by the `autosupport` daemon is:

```
options autosupport.to addresses,...
```

You can specify up to five email addresses. Separate email addresses by commas and do not include spaces in the list. (Default is autosupport@netapp.com.) For example:

```
options autosupport.to autosupport@netapp.com,d@c.com
```

Be sure to enter the actual command on a single line.

Specifying the filer administrator's address

Sometimes Network Appliance Technical Support may need to contact the administrator of the filer after receiving automatic email. The `options` command for specifying the filer administrator is:

```
options autosupport.from address
```

For example:

```
options autosupport.from jdoe@abc.com
```

If there is a need to determine whether further action is necessary for solving or prevent filer problems, Network Appliance Technical Support sends email to jdoe@abc.com.

Sending an immediate message

The command to send an automatic email message immediately is:

`options autosupport.doit` *`string`*

The `string` is used in the subject line of the email message to explain why the email was sent. For example:

`options autsupport.doit TESTING`

Immediate messages contain the same filer data as automatic messages.

Sending a short message

The `options autosupport.noteto` command specifies the recipients of short email messages sent by `autosupport`. The short email messages are for urgent events such as disk failures or filer reboots. The following is an example of the short message:

```
Return-Path: <autosupport>
Received: from toaster by netapp.com (4.1/SMI-4.1)
id AA14370; Thu, 26 Sep 96 07:51:31 PDT
Message-Id: <9609261451.AA14370@netapp.com>
From: autosupport
To: jdoe
Date: Thu, 26 Sep 1996 07:51:27 -0700
Resent-Date: Thu, 26 Sep 1996 7:58:42 PDT
Subject: System Alert from toaster
REBOOT on Thu Sep 26 07:51:27 PDT 1996
```

Short messages are useful if the person who should be notified of urgent events reads email on a small screen such as that on an alphanumeric pager.

Synchronizing filer system time

The filer uses two commands to control the system time. The `date` command sets the time locally, while the `rdate` command synchronizes the filer's time with the time on another host.

Setting the date and time

Set the date and time. For example:

```
toaster> date 199705220925
```

would set the date and time to 9:25 A.M. on May 22, 1997.

Synchronizing the filer

The `rdate` command synchronizes the filer's time with a target computer's time using the UDP time service (typically UDP port 37). This service is available on most UNIX computers. Check */etc/services* and */etc/inetd.conf* on the target computer to see whether this service is supported.



NOTE: There is currently no Windows 95 or Windows NT counterpart to the `rdate` command. If you want to use the `rdate` command, you must have a UNIX workstation that supports `rdate` on your filer's network.

To keep the filer's clock accurate, regularly run the `rdate` command on the filer with the target machine, a computer that maintains accurate time and that supports the port 37 UDP time service. For example, if the name of the target computer is "time_node," enter:

```
toaster> rdate time_node
```

A typical scheme would be to have a UNIX computer run a periodic `cron` job that executes the appropriate `rdate` command on your filer through `rsh`. The computer and the user (if not root) running the `cron` job must be in the filer's */etc/hosts.equiv* file.

For example, on the UNIX computer `adminhost`, the user `adminuser` sets up a `cron` job to run every day at 3 A.M. It directs the filer `toaster` to request a time update from the computer named "time_node," which maintains an accurate time and supports the UDP time service.

The `crontab` entry on the UNIX system would be:

sys% **`0 3 * * * rsh toaster -l root rdate time_node`**

The */etc/hosts.equiv* file on *toaster* must contain the following line:

adminhost adminuser

The host names, *adminhost* and *time_node*, must be known to the filer.

Maintaining filer security

The following options to the `options` command help to maintain filer security:

- The `telnet.hosts` option restricts Telnet access to a limited number of hosts. Using this option, you specify a comma-separated list of up to five hosts that can log in to the filer using `telnet`. Alternatively, you can disable `telnet` for all hosts by specifying a hyphen (-). By default, the option argument consists of an asterisk (*), which means all hosts have `telnet` access.

  A record of all Telnet and console logins is maintained in the */etc/messages* file.

- The `mount_rootonly` option restricts the mount privilege to root using privileged ports (ports 1 through 1024). By default, the option is enabled. This option accepts the arguments “off” and “on” to disable and enable it. This option is applicable only if your filer runs NFS.



NOTE: Some PC clients and some older implementations of NFS on UNIX workstations use nonprivileged ports to send requests. If you have these clients at your site, disable the `mount_rootonly` option or upgrade the client software.

- The `root_only_chown` option allows only root to change the owner of a file. When the option is disabled, the owner of a file can change its ownership without being root. By default, this option is enabled. Use the arguments “off” and “on” to disable and enable the option.

  When a non-root user changes the owner of a file, the set-user-id and set-group-id bits are cleared. If a non-root user tries to change the owner of a file but the change would cause the file’s recipient to exceed his or her quota, the attempt will fail. This option is applicable only if your filer runs NFS.

Software licenses

The filer requires software licenses to enable NFS, CIFS, or http services. Licenses are installed on the filer at the factory per your order, so the initial setup of your filer does not involve entering license codes.

You need to enter license codes only if any of the following conditions applies:

- You purchased a filer with a release earlier than Release 4.0, and you are upgrading it to Release 4.0 or later.
- You want to enable a protocol not previously licensed for your filer.
- The filer's file system becomes corrupt and must be rebuilt.

Network Appliance provides you with the appropriate license codes when shipping you the software upgrade kit or when giving you instructions for obtaining the software upgrade over the Internet.

Enabling NFS, CIFS, or HTTP service

To enter a license code to enable a protocol on your filer, use the `license` command with the following syntax:

license *protocol=code*

For example, if license code is "abcdefg" for NFS, enter:

```
license nfs=ABCDEFG
```

The events that take place after a `license` command depend on the protocol specified:

NFS The filer displays the following messages:

```
nfs license enabled.
        nfs enabled.
```

The filer also automatically runs the `nfs on` command to start NFS service. The filer, however, does not add the `nfs on` command to */etc/rc* as a result of the `license` command. If you want the filer to run NFS service after each reboot, remember to add `nfs on` to */etc/rc*.

CIFS — The filer displays the following messages:

```
CIFS license enabled.
Run cifs setup to enable CIFS.
```

To start CIFS service, set up the filer's CIFS configuration by running `cifs setup`. You don't need to run `cifs setup` after the `license` command if you have already set up the CIFS configuration as described in "CIFS-specific information" on page 18.

http — The filer displays the following messages:

```
http license enabled.
Use "options httpd.enable" to enable http.
```

To start http service, enter the following command:

```
toaster> options httpd.enable on
```

The options command takes effect immediately. If you want the filer to automatically start http service after each reboot, add the options command shown above to /etc/rc.

The following is an example showing how to activate several protocols at once:

```
license nfs=ABCDEFG CIFS=HIJKLMN http=PQRSTUV
```

Displaying current license codes

To display licensing information, enter the `license` command without parameters. The following is an example:

```
toaster>license
nfs=ABCDEFG
cifs=currently not licensed
http=PQRSTUV
```

Disabling a license

To disable a license, enter the word "disable" as the code for the protocol. For example, to disable your filer's license for NFS, enter:

```
license nfs=DISABLE
```

Once a license is disabled, your filer stops service for the corresponding protocol. You can restart the service by re-entering the license code.

Replacing license codes

If you misplace a license code, contact Network Appliance Technical Support to obtain a copy.

Chapter 5: Network administration

This chapter discusses the following topics:

- how to use SNMP on the filer, including information about the Network Appliance custom management information base (MIB)
- how the filer resolves names
- how the filer routes packets using its routing table
- how to configure a network interface
- how to set up a virtual firewall
- how to work with ATM (Asynchronous Transfer Mode) PVCs (permanent virtual circuits)

Using SNMP

For diagnostic and other network management services, the filer supports the SNMP MIB-II specification. Based on SNMP version 1, this specification provides data on the following MIB-II groups: system, interfaces, address translation, IP, ICMP, TCP, UDP, and SNMP MIB-II. SNMP commands permit a user to specify up to eight communities and trap notifications for up to eight management stations.

Configuring the SNMP agent

To use SNMP, configure the SNMP agent using the `snmp` command. A typical set of SNMP commands in the */etc/rc* file would be as follows:

```
snmp contact 'jdoe@abc.com 415-555-1212'
snmp location 'ABC corporation, engineering lab'
snmp community add ro private
snmp traphost add snmp-mgr1
snmp init
```

SNMP commands that NetApp supports are described in detail in the *na_snmp*(1) man page. Brief explanations of these commands are provided below.

`snmp contact 'jdoe@yourorg.com 415-555-1212'`
Specifies the e-mail address and telephone number of the person responsible for the filer. You can include the person's full name, but an e-mail address and a telephone number allow Network Appliance Technical Support to contact the right person after receiving an automatic e-mail via autosupport from your site concerning your filer status.

`snmp location 'ABC corporation, engineering lab'`
Specifies the physical location of the filer. This value will be returned by the SNMP agent.

`snmp community add ro private`
Create a read-only community called "private." The SNMP manager uses the community name as a password to communicate with the filer's SNMP agent.

`snmp traphost add snmp-mgr1`
Make the system "snmp-mgr1" the recipient of all SNMP traps from the filer.

`snmp init 1`
Initialize the SNMP daemon with the values specified above. This command also sends the SNMP cold start and links up or down traps, as appropriate, to any trap hosts that have been previously registered using the `snmp traphost` command.

Using the Network Appliance Custom MIB

The Network Appliance custom MIB provides detailed information about many aspects of filer operation. If your filer was shipped with Release 4.0 or later, the MIB is in the */etc/mib/netapp.mib* file. If you upgraded your filer from a pre-4.0 release, the */etc/mib/netapp.mib* file does not exist on your filer and you can obtain the custom MIB from the NOW web page. Install this MIB file on your Network Management Station (NMS) according to the installation procedure for your station so that the station can obtain information from the filer about the objects that are part of this MIB.

The top-level groups in the custom MIB and the information they contain are described in Table 5-1.

Table 5-1. Groups in the custom MIB

Group name	Contents
product	Product-level information such as the software version string and the system ID.
sysstat	Some system-level statistics such as CPU uptime, idle time, and the number of kilobytes transmitted and received on all network interfaces.
nfs	Statistics like those displayed by the `nfsstat` command, including statistics for each client if per-client statistics are enabled. The per-client statistics are indexed by client IP addresses.
quota	Information related to disk quotas, including the output of the `quota report` command. To access quota information, quotas must be turned on. For more information about quotas, see "Restricting disk usage by imposing disk quotas" on page 232.
filesys	Information related to the file system, including the `maxfiles` and `df` output, and some of the `snap list` output. For more information about `maxfiles`, see "Increasing the maximum number of files" on page 242. For more information about the `df` command, see "The df command" on page 242. For more information about snapshots, see Chapter 10, "Snapshots."
raid	Information on RAID equivalent to the `sysconfig -r` output. For more information about `sysconfig`, see Chapter 12, "System information and performance."

Name resolution

The filer resolves host names by searching maps (or databases) for services to use. The filer tries name resolution services in a default order or in the order that you specify in the */etc/nsswitch.conf* file. For example, by default, first the filer tries to resolve host names locally by searching the */etc/hosts* file (designated by "files" in the table below and in the */etc/nsswitch.conf* file). If the name is not resolved, it tries NIS (if NIS is enabled). If the name is still not resolved, the filer requests services from a DNS server (if DNS is enabled). You can specify any or all of the resolution methods.

Table 5-2 shows the default search order for each map.

Table 5-2. Default order for name resolution

Map	Services in search order
hosts	files NIS DNS
passwd	files NIS
netgroup	files NIS
group	files NIS
shadow	files NIS

To specify a different order in which the filer contacts name services, create the file */etc/nsswitch.conf*. Each line must have the following format:

map: *service* ...

map is one of the following maps or databases: `hosts`, `passwd`, `shadows`, `group`, and `netgroups`.

service is one or more of the following:

- `files` for local files in the */etc* directory
- `dns` for DNS
- `nis` for NIS

List services in the order in which you want the filer to contact the services. For example, the following file:

```
hosts: nis dns files
passwd: nis files
```

instructs the filer to contact first NIS for hosts, then DNS, and finally local files in */etc*, but to contact first NIS, then local files in */etc* for passwords. When a name is resolved, the search stops.



NOTE: When performing CIFS operations, the filer can use wins servers for name service. However, filer commands unrelated to CIFS use the */etc/hosts* file, DNS, or NIS to resolve names as described in the following sections.

The /etc/hosts file

The filer can use the */etc/hosts* file to resolve names used in */etc/rc*, */etc/syslog.conf*, */etc/dgateways*, */etc/exports*, */etc/netgroup*, and */etc/hosts.equiv*. If no host names are used in any of these files, */etc/hosts* is not needed.

The sample setup session shown in Chapter 2, "Getting started" would generate the */etc/hosts* file shown in Figure 5-1.

```
#Auto-generated by setup Wed Jun 4 19:06:18 PST 1997
192.9.200.100   toaster-0 toaster
192.9.201.100   toaster-1
192.9.202.100   toaster-f0
192.9.200.1     MyRouterBox
192.9.200.2     Adminhost
192.9.200.2     Mailhost
```

Figure 5-1. /etc/hosts file

By default, the filer reads the */etc/hosts* file whenever it needs to resolve host names, so changes to the file take effect immediately. If you change the IP address for an interface, the new IP address won't take effect until you reboot the filer and the appropriate `ifconfig` command is executed from */etc/rc*. It is safest to reboot the filer after changing any */etc* file to make sure that the new configuration works correctly.

If you use DNS and have put files first in the */etc/nsswitch.conf* file, the filer looks up host addresses in */etc/hosts* before sending queries to the DNS server. Therefore, it is important that each entry in */etc/hosts* contains accurate information. To reduce the need for updating information in */etc/hosts*, when using DNS, keep only a minimum number of entries in */etc/hosts*.

If you use NIS, you can modify the Makefile of the NIS master to copy the NIS master's */etc/hosts* to the filer when it is changed or you can have the filer use NIS directly. For example, you could put the following at the end of the NIS Makefile section for `hosts.time`:

```
@mntdir=/tmp/netapp_etc_mnt_$$$$;\
if [ ! -d $$mntdir ]; then rm -f $$mntdir; mkdir $$mntdir; fi;\
for filer in toaster1 toaster2 toaster3 ; do \
mount $$filer:/etc $$mntdir;\
mv $$mntdir/hosts $$mntdir/hosts.bak;\
cp /etc/hosts $$mntdir/hosts;\
umount $$mntdir;\
done;\
rmdir $$mntdir
```

Be sure to substitute the name of each filer in the "for" list in place of `toaster1`, `toaster2`, and so on.

Using DNS

The filer includes DNS client capabilities. This means that it can query DNS servers for host-name-to-IP-address and IP-address-to-host-name translation services. With DNS enabled, you no longer have to update the filer's */etc/hosts* file every time a new host is added to the network. (If you use the default search order or put files before DNS in the */etc/nsswitch.conf* file, you still have to update the */etc/hosts* file if one of its entries changes before the filer tries to resolve host names.)

To prevent naming inconsistencies, Network Appliance recommends that when DNS is enabled, you use only the default */etc/hosts* file.

Enabling DNS during system setup

At system setup, if you enter **y** in response to the following prompt:

```
Do you want to run DNS resolver [n]: y
```

The `setup` program prompts you for a DNS domain name as follows:

```
Please enter DNS domain name []:
```

After you have entered a DNS domain name, `setup` asks you for the IP addresses of up to three DNS name servers. Based on the IP addresses you enter, `setup` generates the */etc/resolv.conf* file. Entries in */etc/resolv.conf* file consist of the word "nameserver" followed by an IP address:

`nameserver` *`ip_address`*

For details of name server query policies, see the *na_resolv.conf*(5) man page.

Enabling DNS without using setup

If you didn't start DNS during the system setup process, you can start DNS using the following procedure:

1. Create an */etc/resolv.conf* file. The file consists of up to three lines, each specifying a name server host in the following format:

   `nameserver` *`ip_address`*

   For example:

   ```
   nameserver 192.9.200.10
   nameserver 192.9.200.20
   nameserver 192.9.200.30
   ```

2. Edit the */etc/rc* file to make sure that the option specifying the DNS domain name is set and that the option to enable DNS is set to "on." For example:

   ```
   options dns.domainname netapp.com
   options dns.enable on
   ```

3. For the commands in step 2 to take effect, reboot the filer or enter the commands at the filer prompt.

DNS is now enabled, and you no longer have to update the filer's */etc/hosts* file every time a new host is added to the network (unless you specify `files` first in the */etc/nsswitch.conf* file).

Disabling DNS

To disable DNS, enter the following command:

```
options dns.enable off
```

or put the command in the */etc/rc* file to make the change permanent.

Changing your DNS domain name

To change your DNS domain name, enter the following command:

```
options dns.domainname domainname
```

or put the command in the */etc/rc* file to make the change permanent.

Using NIS

The filer includes NIS client capabilities. This means that it can query NIS servers for host-name-to-IP-address and IP-address-to-host-name translation services.



NOTE: Because the *nsswitch.conf* file already allows you to specify the order in which the filer finds password information, you do not need to use + or - entries in the filer's */etc/passwd* file. Any existing + or - entries are ignored.

The filer uses the following NIS maps:

```
hosts.byname
hosts.byaddr
passwd.byname
passwd.byuid
passwd.adjunct
group.byname
group.bygid
netgroup.byhost
```

Shadow password information is obtained from the `passwd.byname` map.

Enabling NIS during system setup

At the following system setup prompt:

```
Do you want to run NIS Client [n]:
```

If you enter **y**, `setup` prompts you for an NIS domain name:

```
Please enter NIS domain name [] :
```

Enter an NIS domain name.

Enabling NIS without using setup

If you didn't start NIS during the system setup process, you can start NIS using the following procedure:

1. Edit the */etc/rc* file to make sure that the option specifying the NIS domain name is set and the option to enable NIS is on. Insert lines like the following:

   ```
   options nis.domainname netapp.com
   options nis.enable on
   ```

2. Enter the above options commands from the command line or reboot the filer. Options entered only on the command line are not saved if you reboot.

NIS is now enabled, and you no longer have to update the filer's */etc/hosts* file every time a new host is added to the network.

Disabling NIS

To disable NIS, enter the following command:

```
options nis.enable off
```

or put the command in the */etc/rc* file to make the change permanent.

Changing your NIS domain name

To change your domain name on a running system, enter the following command:

```
options nis.domainname new.domain
```

or put the command in the */etc/rc* file to make the change permanent.

Displaying the NIS server name

To display the NIS server name, enter the following command:

```
ypwhich
```

Routing

Even though the filer can have multiple network interfaces, it does not function as a router; that is, the filer does not route packets between its interfaces on behalf of other network hosts.

Routing table on the filer

For routing its own packets, the filer relies upon the default route and explicit routes. Normally, the filer learns explicit routes through ICMP redirect messages received from the default router; you do not need to enter explicit routes in the filer's routing table. To display the filer's current routing table, use the `netstat -r` command. For example:

```
toaster> netstat -r
Routing tables
Internet:
Destination      Gateway            Flags Refs   Use  Interface
default          nwo-               UG    1      138  e0
298.295.227      link#1             UC    0        0  e0
298.295.227.1    link#1             UHL   1       24  e0
nwo-             8:0:20:79:f9:79    UHL   1        0  e0
298.295.227.255  link#1             UHL   1     3696  e0
```

If it is necessary to enter explicit information into the filer's routing table, use the `route` command. See the *na_route*(1) man page on how to add or modify information in the routing table.

If the filer cannot find an explicit route in the routing table for a particular destination, it uses the default route. This means that the filer sends the traffic to the default router, which is specified in the */etc/dgateways* file.

Specifying default routers

One default router is specified during initial system setup. You can, however, add potential default routers at any time to the */etc/dgateways* file. For each added router, you should also specify the metric, which is a number indicating the preference for the router. The highest preference is 1, which is also the default

preference for the router specified during system setup. The lowest preference is 15. For information about the format of the */etc/dgateways* file, refer to "The /etc/dgateways file" on page 121.

To help manage multiple routers and to allow you to create redundant routing schemes, the filer runs a simple routing daemon, `routed`, which is enabled at boot time. This daemon "listens" for Routing Information Protocol (RIP) packets being exchanged between routers on the network to determine which routers are alive. From the routers that are alive, the daemon selects the one with the highest preference to use as the filer's default router.

However, the filer doesn't rely on `routed` to construct the routing table. The function of the filer's `routed` is merely to check the status of the default router. Refer to the *na_route*(1), *na_routed*(1), and *na_dgateways*(1) man pages to learn more about routing on the filer.

You turn `routed` off and on with the `routed` command.

To turn `routed` off, enter:

```
toaster> routed off
```

To turn `routed` back on, enter:

```
toaster> routed on
```

If you choose to turn `routed` off by editing */etc/rc*, you should still manually designate a default router in */etc/rc*.

Displaying routing status

To display the status of the default gateway list, use the `routed status` command. The `-n` option forces the command to display numeric values for gateway names. The display shows the following:

- whether Routing Information Protocol (RIP) snooping is active
- the current list of default gateways
- the metrics of the of default gateways
- he state of the gateways (ALIVE or DEAD)
- the last time each gateway was heard from.

An example of the `routed status` display is shown in Figure 5-2.

```
toaster> routed status
  RIP snooping is on
  Gateway   Metric State   Time Last Heard
  karl           1  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  groucho        1  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  292.0.266.366  1  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  292.0.266.377  1  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  june           2  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  292.0.266.332  2  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  292.0.266.352  3  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  292.0.266.351  4  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  292.0.266.350  5  ALIVE  Wed Mar 9 03:38:41 GMT 1994
  119 free gateway entries, 9 used
```

Figure 5-2. The routed status display

The /etc/dgateways file

The */etc/dgateways* file is the configuration database for `routed`. Among the routers that `routed` has determined to be alive, `routed` selects the one with the highest preference to be the default router. When the filer cannot find an explicit route for a packet, it routes the packet to the default router.

The file consists of lines with the following format:

gateway metric

where *gateway* is the name or the IP address of a default router and *metric* is a preference indicator, which ranges from 1 (highest) to 15 (lowest).

```
#GatewayMetric
192.9.200.101
eng_gateway2
```



NOTE: Each entry for such a default router must have an IP address that belongs to the IP subnet of one of the interfaces configured for the filer.

How the filer replies to requests

The following list describes how the filer uses its interfaces to respond to different types of packets:

- NFS-over-UDP requests

  The filer does not use the conventional IP routing mechanisms to reply to NFS over UDP requests. The filer sends the response on the network interface on which the request was received, to the same address that generated the request. For example, the filer named “toaster” uses the interface toaster-e1 to send packets in response to NFS requests received on the interface toaster-e1.

  This way of handling NFS over UDP requests allows you to attach multiple interfaces of the filer to networks with the same IP subnetwork number while keeping NFS over UDP traffic isolated to the appropriate physical networks.

  Because of this scheme, it is possible that NFS over UDP responses may be returned through a different path than you would expect from an examination of the IP routing table using `netstat -r`. This scheme generally works well although it could result in different routes than expected if your

environment contains one-way routes. (For example, the IP packets may not be routed as you intended if you configured the network so that the IP traffic from host1 to host2 is routed through router1 and the IP traffic from host2 to host1 is routed through router2.

- NFS over TCP, CIFS, and HTTP requests

  The filer tries to return NFS-over-TCP and HTTP traffic over the interface on which the traffic was received. However, there are exceptions. For example, if the filer experiences excessive queuing to a response or experiences a time-out followed by a retransmit, the filer will route the traffic by using conventional IP routing table lookups. If the filer has multiple interfaces attached to networks with the same IP network number, the filer uses the first interface it finds with that number to send the responses.



NOTE: If the NFS-over-UDP, NFS-over-TCP, CIFS, or HTTP traffic takes place on an ATM interface, the filer always uses conventional IP routing table lookups and routing for that traffic.

- IP-based traffic other than NFS and HTTP requests

  For other types of traffic (for example, traffic generated by `telnet`, `rsh`, and `ping`), the filer uses IP routing table lookups and routing. If the filer has multiple interfaces attached to networks with the same IP network number, the filer will use the first interface it finds with that number to send the responses.

Using ifconfig to configure an interface

The */etc/rc* file contains `ifconfig` commands to configure network interfaces at system boot. You can also manually run `ifconfig` when the system is operating; the command syntax is as follows:

```
ifconfig interface [ address ] [ netmask mask ]
[ broadcast address ] [ mediatype type ]
[ mtusize size ] [ up | down ] [ trusted | untrusted ]
```

or

```
ifconfig -a
```

Usually, you use the `ifconfig` command for the following purposes:

- Changing the interface's IP address, network mask, or broadcast address. The following are examples of configuring an Ethernet interface on a filer:

```
ifconfig e1 292.9.299.6
ifconfig e2 netmask 255.255.0.0
ifconfig e3 broadcast 292.9.299.255
```

  The interface names on your system may be different depending on your filer model. For information about interface naming conventions, see "Naming conventions for network interfaces" on page 77.

- Setting the media type on an Ethernet interface. The following are examples of configuring an Ethernet interface on a filer:

```
ifconfig e1 mediatype tp
ifconfig ns1 mediatype thick
ifconfig ns2 mediatype thin
```

The media types you can use depend on the type of Ethernet card. The possible types you can enter in the `ifconfig` are the same as those you can select when running `setup`. They are described in Table 5-3 below.

Table 5-3. Media types for Ethernet interfaces

Media type	Description
100tx	100Base-T
100tx-fd	100Base-T, full-duplex
tp	10Base-T
tp-fd	10Base-T, full-duplex
auto	10Base-T / 100Base-TX Ethernet using autonegotiation
1000fx	gigabit Ethernet
thick	thick Ethernet
thin	thin Ethernet

- Setting the maximum transmission unit (MTU). The following is an example of setting the MTU for an FDDI interface on a NetApp F330:

  ```
  ifconfig f0 mtusize 1500
  ```

  By default, the MTU is 4352 on an FDDI interface, 1500 on an Ethernet interface, and 9188 on an ATM interface. Use a smaller MTU value for an interface if a bridge or router on the attached network cannot break large packets into fragments.

  To view the current setting of the MTU value, use the `netstat -i` command.

- Configuring the interface up or down. For example:

```
ifconfig e1a up
ifconfig f1 down
```

If you want changes made with `ifconfig` to remain in effect after a reboot, include the `ifconfig` commands in the */etc/rc* file.

To show the current configurations of all network interfaces, enter the following:

```
ifconfig -a
```

To show the current configuration of a specific network interface, enter the following:

```
ifconfig interface
```

For example, to show the current configuration of interface `e0`, enter the following:

```
ifconfig e0
```

For more information about the `ifconfig` command, refer to the *na_ifconfig*(1) man page.

ATM PVCs

The filer dynamically assigns SVCs (Switched Virtual Circuits) when interoperating with ATM hosts and with switches that support the FORE IP / SPANS protocols. To interoperate with ATM hosts and with switches that do not support FORE IP / SPANS, you must use ATM PVCs (Asynchronous Transmission Mode Permanent Virtual Circuits). PVCs are static, and for each destination, you must be establish (attach the IP layer to) a PVC explicitly and delete (detach the IP layer from) the PVC explicitly.

For each destination that needs to establish a PVC with the filer, you must establish an outgoing and an incoming PVC in at least three places:

- at the filer
- at the destination ATM host
- on all interconnecting ATM switches

Establishing an outgoing PVC on the filer

To establish an outgoing PVC, enter the following filer command:

`arp -s` *`hostname`* `pvc` *`iface vpi vci`* `[`*`aal`* `[`*`encap`* `[`*`addr`*`]]]`

where

hostname is the name or IP address of the remote host.

iface is the name of the ATM interface.

vpi is the VPI (virtual path identifier); this must be 0.

vci is the VCI (virtual channel identifier); this number must:

- not be in use on your filer.
- be less than 1024.
- obey the limits of the destination host and interconnecting devices.

aal is the ATM adaptation layer type. It must be 4 or 5, and should be the AAL type supported by the destination host (typically 5). The default is 5.

encap is the encapsulation type. The encapsulation type specified should be one of:

  - null (no encapsulation; this is the default)
  - llc_routed (IEEE LLC encapsulation for routed PDUs)
  - llc_bridged_8023 (IEEE LLC encapsulation for Ethernet/802.3 bridged PDUs)

and should be the same as the encapsulation type used by the destination host.

If the encapsulation is llc_bridged_8023, you must include *addr*.

addr is a six-byte colon-separated destination MAC address.

Establishing an incoming PVC on the filer

To establish an incoming PVC, use the following command at the filer:

```
arp -l pvc iface vpi vci [aal [encap]]
```

Establishing a PVC on the remote ATM host

Set up a PVC on the remote ATM host according to the documentation for that host.

Establishing a PVC on interconnecting ATM switches

On the interconnecting ATM switches, assign virtual channels corresponding to the VPI and VCI entries made on the filer and the remote ATM host.

Displaying information about a PVC

To display information about all PVCs (and other interfaces) on a host, enter the following:

```
arp hostname
```

where *hostname* is the name of a host.

To display information about all of the current PVCs, enter the following:

```
arp -a
```

A display like the one in Figure 5-3 appears:

```
toaster> arp -a
aye (172.16.1.1) at 0:0:3b:80:36:88
bdellium (172.16.1.2) at 0:0:3b:80:36:d4
ctenophore (172.16.1.3) at 0:0:3b:80:36:d7
mnemonic (172.16.1.4) at 0:0:3b:80:2e:5b
oedipus (172.16.1.5) at 8:0:20:80:8d:47
psych (172.16.1.6) at 8:0:20:80:8d:d3
xenon (172.16.1.7) at 8:0:20:81:d9:47
yreka (172.16.1.9) at 8:0:20:7a:65:19
Incoming ATM Connections:
iface=a5 switch.port=f21a2420.56 vpi.vci=0.114 aal=5
    encapsulation=NULL
iface=a5 switch.port=f21a2420.25 vpi.vci=0.113 aal=5
    encapsulation=NULL
```

Figure 5-3. arp -a output

Deleting a PVC

To delete an outgoing PVC entry, enter the following:

`arp -d hostname`

where

hostname is the name or IP address of the remote host.

To delete an incoming PVC for a remote host, enter the following:

`arp -x iface vpi vci`

where

iface is the name of the interface.

vpi and *vci* have the values of the VPI and VCI of the PVC to be deleted for the specified interface.

Chapter 6: **NFS administration**

This chapter discusses the following topics:

- exporting file systems to NFS clients
- configuring a filer for (PC) NFS
- configuring a filer for WebNFS
- collecting statistics about NFS performance

Exporting NFS file systems

You can export file systems or directories (mount points) to clients individually or to subnets.

Exporting to individual clients

To export file systems or directories from the filer, list them in the filer's */etc/exports* file and run the `exportfs` command. For example:

```
toaster> exportfs -a
```

This command exports all the file systems listed in the */etc/exports* file. Use the `-u` option to unexport file systems.

The /etc/exports file

The */etc/exports* file lists all directories that the filer can export by the `exportfs` command. The default *exports* file contains the following lines:

```
#Auto-generated by setup Wed May 31 17:44:29 PDT 1995
/-access=adminhost,root=adminhost
/home-root=adminhost
```

In your */etc/exports* file, *adminhost* is the name of the administration host that you entered during system setup.

The following explains the contents of the default */etc/exports* file:

- Only the administration host can mount the / directory and modify the files in that directory (for example, */etc/hosts*, */etc/exports*).
- Clients anywhere have full access to any files in the */home* directory, with the administration host having root permission.

NOTE: If you are concerned that the default */etc/exports* file gives all clients read and write permissions to */home*, use the `chmod` command to change modes for */home* so that only users belonging to particular groups have permissions to read and modify files in */home*. You can also use the `-rw` or `-access` option in */etc/exports* to limit write privilege to listed hosts. The `-rw` option lists the hosts that can modify the exported file system; hosts not listed by the `-rw` option have read privilege only. The `-access` option restricts mount access to listed hosts. Refer to the *na_exports*(1) man page for more information about options in the */etc/exports* file.

You can specify up to 256 host names for each `-rw` and `-root` option. There is no limit on the number of host names for the `-access` option, provided that each line in the */etc/exports* file has no more than 1024 characters. Use netgroups for lists that would otherwise result in lines longer than 1024 characters.

Be sure that the filer can resolve all host names specified in the */etc/exports* file. Refer to "Name resolution" on page 111 for more information about host-name-to-ip-address resolution.

NOTE: While you *can* specify in */etc/exports* which clients can mount files, the filer (and NFS) doesn't offer a mechanism for you to specify a list of clients that are *not* allowed to mount the filer.

In many UNIX implementations, it is illegal to export a directory that has an exported ancestor in the same file system. The filer has no such restriction. Because the filer has one large file system, this limitation would be unduly restrictive. Therefore, it is legal to export both the */* and */home* directories.

The filer uses the longest matching prefix in determining permissions. In the example above, a client mounting */home/user1* would get permissions for */home*, because */home* is the longest matching prefix. A client mounting */* would get *-access=adminhost,root=adminhost*

permissions. Because of this, it makes little sense to give a client greater permissions at a higher level in the file system. For example, if the *exports* file were to contain the following lines:

```
/
/home-access=bashful:dopey:sleepy
```

only the three clients specified in the second line could mount */home* directly, but any client could access the */home* subtree simply by mounting */*. This would allow a breach in system security and should be avoided.

Instead of naming individual clients in the `-access` option of */etc/exports* or the `exportfs` command, you can specify a group of clients (a network group) defined in the */etc/netgroup* file. See the next section on network groups.

After editing */etc/exports*, either execute `exportfs -a` or `reboot` to make the new configuration take effect.



NOTE: If you are deleting entries in the new */etc/exports* file, you need to run `exportfs -au` first and then `exportfs -a`; otherwise, the previously loaded exports will remain in effect.

The /etc/netgroup file

The filer's */etc/netgroup* file defines groups of clients that the filer uses for checking access permission while processing a mount request. Each line of this file defines a group and has the following format:

```
groupname member-list
```

Each element in `member-list` is another group name or a triplet of the form:

```
(hostname, username, domainname)
```

An element can be blank, but the commas must be present.

Because only the group name is used in the filer's */etc/exports* file or `exportfs` command, you do not have to specify the `username` and `domainname` fields. Changes made to the */etc/netgroup* file take effect immediately.

In the */etc/netgroup* file, the maximum number of characters per line is 4096. The following is an example of the */etc/netgroup* file:

```
trusted-hosts(adminhost,,)
untrusted-hosts(bashful,,) (dopey,,) (sleepy,,)
all-hoststrusted-hosts untrusted-hosts
```

With this */etc/netgroup*, you could write entries in */etc/exports* as follows:

```
/-access=trusted-hosts,root=adminhost
/home-access=all-hosts,root=adminhost
```

Do not use netgroup names in the lists for the `root=` and `rw=` options in the */etc/exports* file.

The network groups on the filer are not linked with NIS; an existing NIS network group must be copied over to the filer's */etc/netgroup* before it can be used by `exportfs`. You can modify the Makefile of the NIS master to copy the NIS master's */etc/netgroup* to the filer when it is changed. For example, you could put the following in the NIS Makefile section for `netgroup.time`:

```
@mntdir=/tmp/nac_etc_mnt_$$$$;\
if [ ! -d $$mntdir ]; then rm -f $$mntdir; mkdir $$mntdir; fi;\
for filer in toaster1 toaster2 toaster3 ; do \
mount $$filer:/etc $$mntdir;\
mv $$mntdir/netgroup $$mntdir/netgroup.bak;\
cp /etc/netgroup $$mntdir/netgroup;\
umount $$mntdir;\
done;\
rmdir $$mntdir
```

Be sure to substitute the name of each filer in the "for" list in place of *toaster1*, *toaster2*, and so on.

Exporting to Subnets

You can export a mount point to machines in a subnet rather than only to machines individually. You can include netgroups in the exported paths unless the path is exported with the `-rw` or `root` command options. You would export a mount point to a subnet when you want to export a mount point easily to many machines rather than to machines on an individual basis with individual commands.

You export a mount point to a subnet as you would to an individual client, but you specify a subnet address rather than a full IP address in an export option. The valid export options for exporting to a subnet are:

```
-o ro=hostname[:hostname]...
-o rw=hostname[:hostname]...
-o root=hostname[:hostname]...
```

where each *hostname* can be a subnet address.

Specifying a subnet

You specify a subnet address with a dotted IP subnet address and a mask using the following syntax:

```
dotted_ip/num_bits
```

where *dotted_ip* can be a standard IP address (“a.b.c.d”) or an IP subnet (a for a class A network, a.b for a class B network or a.b.c for a class C network). The size of the subnet is specified by the number of leading bits of the netmask, *num_bits*.

Subnet specification examples

Here are some examples of exporting subnets.

To export */home* on the filer for root access to tooting and all addresses of the form “123.45.67.x” with a netmask 255.255.255.0 (24 leading bits), enter:

```
exportfs -o root=tooting:123.45.67.0/24 /home
```

To export */home* on the filer for read and write access to all addresses of the form “123.45.x.y” with a 16 bit netmask 255.255.0.0, enter:

```
exportfs -o rw=123.45.67.8/16 /home
```

The following export commands are equivalent, and export */home* on the filer to larry, the specified subnet, and curly:

```
exportfs -o rw=larry:123.45.67.8/24:curly /home
exportfs -o rw=larry:123.45.67/24:curly /home
```

Configuring a filer for (PC) NFS

The `pcnfsd` authentication daemon allows you to use (PC) NFS to access a filer. (PC) NFS refers to NFS client implementations for dos, Microsoft Windows, Windows NT, OS/2, and the Macintosh operating systems. Using (PC) NFS, such systems can mount a filer's file system after the user provides the UNIX user name and password for authentication. Without the `pcnfsd` daemon running on the filer, the system administrator would have to set up another host to provide the authentication.

NetApp supports (PC) NFS version 2 except for printer support. With (PC) NFS version 2, clients can use the filer for authentication, including obtaining group permissions (GIDs) and group names through the */etc/group* file or the corresponding group.byname NIS map.

To use (PC) NFS to access a filer, the client system must run (PC) NFS and must be configured to mount the filer's file system. Also, the `pcnfsd` daemon can run only if NFS is licensed for your filer. Otherwise, the following warning messages appear:

```
pcnfsd is NOT running
Cannot use pcnfsd without nfs.
(Use the "nfs on" command to enable nfs.)
```

Starting the (PC) NFS server

On the filer, perform these steps to start the (PC) NFS server:

1. Enable the `pcnfsd` daemon, using the `options` command:

   ```
   toaster> options pcnfsd.enable on
   ```

   To preserve this option setting after reboots of the filer, place this command in the filer's */etc/rc* file.

2. Verify that each user of (PC) NFS has an entry in your UNIX */etc/passwd* file (and */etc/shadow* file, if applicable).

3. If the filer has not been configured to use NIS for the password database, copy */etc/passwd* and */etc/shadow* (if applicable) to the

filer. For example, on the UNIX system where */etc/passwd* and */etc/shadow* reside, enter the following:

```
% cp /etc/passwd /nfs/toaster/etc/passwd
% cp /etc/shadow /nfs/toaster/etc/shadow
```

In this example, the mount point for the filer's root file system is */nfs/toaster*.

To make sure these files on the filer are always up to date, modify the Makefile for the NIS master to copy them to the filer. For example, you could put the following at the end of the NIS Makefile section for `passwd.time`:

```
@mntdir=/tmp/nac_etc_mnt_$$$$;\
if [ ! -d $$mntdir ]; then rm -f $$mntdir; mkdir $$mntdir; fi;\
for filer in toaster1 toaster2 toaster3 ; do \
mount $$filer:/etc $$mntdir;\
mv $$mntdir/hosts $$mntdir/hosts.bak;\
cp /etc/passwd $$mntdir/passwd;\
cp /etc/shadow $$mntdir/shadow;\
umount $$mntdir;\
done;\
rmdir $$mntdir
```

Bu sure to substitute the name of each filer in the "for" list in place of *toaster1*, *toaster2*, and so on. Add the line copying the */etc/shadow* file only if you are using a shadow password file.

If you are not using NIS, you could establish a `cron` job to update the filer.

The filer is now ready to serve files to NFS clients using (PC) NFS. Each user on the client system can mount the file system from the filer after entering a correct user name and password.

Default umask

You can set a default umask for clients with the following options command:

```
options pcnfsd.umask nnn
```

where *nnn* is the umask.

The default is 022, which means that users can read and write their own files, but that members of their group and others can only read the files.

Configuring a filer for WebNFS

WebNFS enables the filer respond to requests for file transfers over the Web using NFS rather than HTTP made through a browser that has WebNFS client support. The headers of WebNFS requests have URLs that start with “nfs://”.

With WebNFS, the filer can transfer files much faster than with HTTP because WebNFS can transfer several files (including graphics files) with only one TCP connection, where HTTP (in version 1.1) requires a separate connection for each file. You do not need an HTTP license to respond to WebNFS requests.

Restricting access

WebNFS access does not use the `mount` command to enable access to a subtree, and does not consider UID/GID mapping. Requests are restricted as follows:

- Requests are honored only for files in subtrees that have been exported.
- If a subtree has been exported with the `-o access` option, files in that subtree are not available through WebNFS.

For information about the exportfs command, consult the *na_exportfs*(1) man page.

Turning on WebNFS

To turn on WebNFS, set the `nfs.webnfs.enable` option to the `options` command to on, as follows:

```
options nfs.webnfs.enable on
```

To turn WebNFS off, set the option to off, as follows:

```
options nfs.webnfs.enable off
```

Specifying and setting a root directory

You can specify a public directory (known as the root directory) for WebNFS access. All relative NFS lookups are done relative to the root directory. An administrator would turn on WebNFS to make documents in the root directory available to the public.

Specifying a root directory

To specify a root directory, use the `options nfs.webnfs.rootdir` command with the syntax:

```
options nfs.webnfs.rootdir directory
```

Setting the root directory

To set the root directory and, use the `nfs.webnfs.rootdirset` options command as follows:

```
options nfs.webnfs.rootdirset TRUE
```

Changing a root directory

Once you have specified a directory, you can change it by specifying a different directory with the `options nfs.webnfs.rootdir` command using the following syntax:

```
options nfs.webnfs.rootdir newdir
```

then setting the root directory with the `nfs.webnfs.rootdirset` options command as follows:

```
options nfs.webnfs.rootdirset TRUE
```

Disabling the root directory

To disable the root directory, use the `options nfs.webnfs.rootdirset` command as follows:

```
options nfs.webnfs.rootdirset FALSE
```

Displaying NFS statistics

The `nfsstat` command displays statistics on NFS and RPC for the filer. You can use the output of this command to find performance bottlenecks or inefficiencies in your NFS setup.



NOTE: A full description of the meaning of NFS statistics is outside the scope of this guide. A good source of information on this topic is *Managing NFS and NIS* by Hal Stern, O'Reilly & Associates, Inc.

The syntax of the `nsfsstat` command is:

```
nfsstat [ interval ]
nfsstat -h [ ip_address | host_name ]
nfsstat -l
nfsstat -z
```

The command is explained in detail in this section.

With no arguments, `nfsstat` displays statistical information since the last time you rebooted the filer.

If you specify an interval (in seconds), `nfsstat` continuously displays summary information. Figure 6-1 shows an example of `nfsstat` output.

```
toaster> nfsstat
Server rpc:
calls    badcalls  nullrecv badlen   xdrcall
10273    0         0        0        0

Server nfs:
calls    badcalls
10273    0

Server nfs V2:
null     getattr   setattr  root     lookup    readlink read
1 0%     3103 30%  0 0%     0 0%     7056 69%  0 0%     99 1%
wrcache  write     create   remove   rename    link     symlink
0 0%     10 0%     0 0%     0 0%     0 0%      0 0%     0 0%
mkdir    rmdir     readdir  statfs
0 0%     1 0%      2 0%     1 0%

Server nfs V3:
null     getattr   setattr  lookup   access    readlink read
0 0%     00%       0 0%     0 0%     0 0%      0 0%     0 0%
write    create    mkdir    symlink  mknod     remove   rmdir
0 0%     0 0%      0 0%     0 0%     0 0%      0 0%     0 0%
rename   link      readdir  readdir+ fsstat    fsinfo
pathconf
0 0%     0 0%      0 0%     0 0%     0 0%      0 0%     0 0%
commit
0 0%
```

Figure 6-1. nfsstat output

To enable per-client NFS statistics, enter the following command:

```
toaster> options nfs.per_client_stats.enable on
```

If you enable per-client NFS statistics, use the following options to display them:

- To display a list of the clients whose statistics have been collected, along with the total NFS calls for that client, use the `nfsstat -l` command.

Figure 6-2 shows an example of `nfsstat -l` output.

```
toaster> nfsstat -l
172.17.25.13  sherlock   NFSOPS =     2943506 (90%)
172.17.25.16  watson     NFSOPS =     3553686 ( 2%)
172.17.25.18  hudson     NFSOPS =     2738083 ( 1%)
172.17.230.7  conan      NFSOPS =     673247l ( 3%)
172.17.230.8  baker      NFSOPS =   202614527 ( 1%)
172.17.230.9  moriarty   NFSOPS =     1006881 ( 0%)
175.17.230.10 doyle      NFSOPS =        1185 ( 0%)
```

Figure 6-2. nfsstat -l output



NOTE: If you want per-client statistics, enable per-client statistics collection as soon as possible after you have started the filer or have reset the counters with `nfsstat -z`, otherwise, `nfsstat -l` reports incorrectly low percentages and displays statistics that include clients that have generated RPC calls but no NFS calls to the client.

- To display details for one client only, use `nfsstat -h` with the client's ip address or host name as an additional argument:

  ```
  toaster> nfsstat -h hostname
  ```

Figure 6-3 shows an example of `nfsstat -h` output:

```
toaster> nfsstat -h eng_host
Client: 172.17.25.3 (eng_host)  ------------------------------
Server rpc:
calls   badcalls nullrecv badlen  xdrcall
33374   0        0        0       0

Server nfs:
calls   badcalls
33345   0

Server nfs V2:
null    getattr  setattr  root     lookup     readlink  read
0 0%    8410 25% 19 0%    0 0%     13687 41%  42 0%     489 22%
wrcache write    create   remove   rename     link      symlink
0 0%    3225 10% 12 0%    7 0%     9 0%       0 0%      0 0%
mkdir   rmdir    readdir  statfs
0 0%    0 0%     416 1%   29 0%

Server nfs V3:
null    getattr  setattr  lookup   access     readlink  read
0 0%    0 0%     0 0%     0 0%     0 0%       0 0%      0 0%
write   create   mkdir    symlink  mknod      remove    rmdir
0 0%    0 0%     0 0%     0 0%     0 0%       0 0%      0 0%
rename  link     readdir  readdir+ fsstat     fsinfo    pathconf
0 0%    0 0%     0 0%     0 0%     0 0%       0 0%      0 0%
commit
0 0%
```

Figure 6-3. nfsstat -h output

To reset counters, enter:

```
toaster> nfsstat -z
```

For more information about `nfsstat`, see the *na_nfsstat*(1) man page.

NOTE: The information displayed by the `nfsstat` command (including per-client NFS statistics) is also available through SNMP if you use the Network Appliance custom MIB. For more information about the custom MIB, see "Using the Network Appliance Custom MIB" on page 109.

Chapter 7: **CIFS administration**

This chapter describes how to use filer commands for these tasks:

- adding clients to the filer
- using CIFS commands through a remote shell
- enabling guest access
- preventing unintended guest access
- creating a home directory for a CIFS user
- managing shares
- managing access rights
- using oplocks
- using NetBIOS Scope IDs
- displaying CIFS statistics
- displaying CIFS session information
- stopping and restarting CIFS service

This chapter does not discuss procedures that take place on the clients. Also, it does not describe how a user joins a workgroup or domain. For information about these topics, refer to the manuals for your client operating systems or books about pc networking.

Adding clients to the filer

You can add CIFS users to the filer at any time. The following is an outline of the procedure for adding a CIFS user:

1. This step is applicable only if you use a domain controller for user authentication.

   Create an account for the CIFS user on the domain controller that is used for authenticating users. For more information about this step, refer to the manual for the system used as a domain controller.

2. Unless you are using NIS, create entries for the user in */etc/passwd* and */etc/group*.

   a. Create an entry for the user in */etc/passwd*.

   b. Include the user in */etc/group* if the user needs to be in more than one group.

3. If necessary, modify the ACL (Access Control List) for each share to be used by the user.

Step 2 in the outline is described in "Modifying /etc/passwd and /etc/group" on page 30. Step 3 is described in detail in "Assigning rights to users for each share" on page 35.



NOTE: How you manage file access rights depends on whether you specified during `setup` that the filer uses PC style or UNIX style to determine users' access rights. Refer to "Choosing the method for deciding access rights" on page 163 to see whether you need to manage the UNIX-style permissions for directories created for the new user.

Using CIFS commands with rsh

All CIFS commands can be used through a remote shell. For example, on the administration host, you can set access rights for a user as follows:

```
adminhost% rsh -l root -n toaster cifs access home
jsmith r-x
```

In this example, the filer's name is "toaster," the user is "jsmith," the share is "home," and the user has read and execute rights to the *home* directory.

Because you can use CIFS commands through `rsh`, you can automate the task of defining access rights for multiple CIFS filers with similar user information. For example, you can create a script containing CIFS commands to enter user information for each filer at your site.

Guest access

You can enable users who do not have an account on the filer to have access to the filer through CIFS in two ways:

- To give access to users who are *not* in a trusted domain, use a guest account.
- To give access to users who *are* in a trusted domain, use a generic user account.

Guest account

You can use the guest account to give users access to a filer as follows:

If you are using a domain controller, guests are users who are not in trusted domains.

If you are using */etc* or NIS for authentication, guests are all users who do not have an entry in */etc* (or NIS).

Use the following option:

`options cifs.guest_account` *`account_name`*

where

account_name is the name of the guest account, usually `guest` or `nobody` (a preconfigured account in the */etc/passwd* file), to provide a guest account. To disable guest access, use a blank as *account_name*.

You would set the guest account to the name of an account in the NIS passwd map or */etc/passwd* (typically `guest`, which is mapped to the UNIX account `nobody` with the same access rights as the user `everyone`).

Generic account

If you are using a domain controller, you can allow users who are in trusted domains but who do not have a password on the filer to connect to the filer. To create such an account, use the following option:

`options cifs.generic_account` *`account_name`*

where

account_name is the name of the generic account. All users in the generic account have the same rights.

Creating a directory for a CIFS user

To create a home directory for a CIFS user, use the NetApp Permissions Tool. The NetApp Permissions Tool and its documentation is available from the NetApp On the Web (NOW) web site at *http://now.netapp.com*.

After creating a home directory, you can offer the directory as a share to a user as described in "CIFS share offer of a "home" directory for each user" on page 158.

You can also use the `cifs mkdir` command. The syntax for this command is:

`cifs mkdir` *`share user path`*

where:

share is a share,

user is the login of a user, and

path is a path relative to the root directory of share.

The `cifs mkdir` command enables an administrator using a DOS or Windows machine to create for a UNIX user a directory having the appropriate UNIX-style permissions, and at the same time to make the user the directory's owner. The directory has *user* as its owner and *user*'s primary group as its group owner.

Managing shares

This section describes how to create a share, display information about shares, and delete a share. By default, two shares are created: the hidden share *C$* and */home*.

Creating a share

If you want a directory to be shared by CIFS clients, use the `cifs shares` command to define a new share. The following is the syntax for `cifs shares`:

`cifs shares -add` *`sharename path description`*

The following describes the parameters in the command:

- *sharename* is the name of the share, which is used by CIFS users to obtain access to the directory on the filer. The name must be a valid 8.3 name.

  If *sharename* already exits, the `cifs shares -add` command will overwrite the old information.



CAUTION: Do not create shares whose names end with a dollar sign ($); doing so might cause conflicts with reserved names. In particular, do not create shares called *C$* or *ADMIN$*. *C$* is a reserved share name and *ADMIN$* is an illegal share name.

- *path* is the complete path name of the shared directory on the filer.



CAUTION: Because the `cifs shares` command is case-sensitive, be sure that you use the appropriate case when entering the path name.

Use only a forward slash (/) for a path name separator.

- *description* is a string describing the purpose of the share. It is required by the CIFS protocol and is displayed in the share list in Network Neighborhood on the client. If the description contains spaces, it must be enclosed in single quote marks.

The following is an example of creating a new share:

```
cifs shares -add library /home/lib 'New file library'
```

This example creates the *library* share and defines it as */home/lib*. With the appropriate access rights, CIFS users can gain access to the */home/lib* directory, which is displayed as the *library* share on their computers. For more information about setting access rights, refer to "Managing access rights for users or groups" on page 170.



NOTE: Do not rename a share if a file in the share is currently opened by a client application. For example, you can use the `cifs shares` command to rename the *library* share to *new_lib* only if no files in */home/lib* are currently opened by any CIFS clients.

Changing the description of a share

To change the description of a share, enter the following command:

```
cifs shares -change sharename description
```

The following describes the parameters in the command:

- *sharename* is the name of the share whose description you want to change.
- *description* is a string describing the new purpose of the share. If the description contains spaces, it must be enclosed in single quote marks.

You can change the description of a share even if the share is in use. The change takes effect immediately.

Displaying information about shares

To display information about the shares that you have defined, enter the following command:

`cifs shares [`*`sharename`*`]`

The following command:

`cifs shares `*`sharename`*

displays information only for the share *sharename*.

The following command:

`cifs shares`

displays information about all shares.

The following is sample output from `cifs shares`:

```
Name          Mount Point      description
HOME          /home             Default Share
           rw- laura
           rw- bob
           rwx jane
C$            /                 Remote Administration
              rwx george
```

Share permissions are shown by triplets consisting of rwx or a dash replacing one or more of the letters, as follows:

- r indicates read permission granted, while a hyphen in the first position indicates read permission denied.
- w indicates write permission granted, while a hyphen in the second position indicates write permission denied.

- x indicates execute permission granted, while a hyphen in the third position indicates execute permission denied.

NOTE: Delete permission has been included in the write permission. Browse permission has been included in the execute permission.

Deleting a share

You can delete a share only if client applications have no open files in the share. To delete a share, use this command syntax:

```
cifs shares -delete sharename
```

The following is an example:

```
toaster> cifs shares -delete library
```

This example deletes the share created by the `cifs shares -add` command described in "Creating a share" on page 155.

CIFS share offer of a "home" directory for each user

You can make an automatic share offer to each user of a "home" directory specific to that user. The homes feature provides a way of doing this that is easy to administer without having to create individual shares for each user.

NOTE: You can also create home directories using NT User Manager. For information about using NT User Manager, consult Windows NT documentation.

On the filer, with the homes directory enabled, there is a directory for each registered user, and also one for "guest" (assuming the guest access option is set to the PC convention "guest"). Thus, if there are users called user1 and user2, the home directory contains subdirectories for user1, user2, and guest. When user1 connects to the filer and asks for its list of shares, the display shows an offer called user1, but not user2 or guest (or any other individual user).

For example, a PC client displays the shares offered to user1 as follows:

```
>net view \\filer
Shared resources at \\filer
NetApp
Share name   Type         Used as  Comment
-------------------------------------------------------------
USER1        Disk                  Home Directory
AGILE        Disk                  Agile Software
DVLP         Disk                  Engineering Development
RLSE         Disk         R:       Engineering Releases
WEB          Disk         (UNC)    Internal Web
The command completed successfully.
```

The user's home directory also appears in Network Neighborhood. The sorting shown above varies with the operating system.

A user accesses his or her home directory in the same way as any other share. That is, the user can open the share with Network Neighborhood, by mapping a drive, or by using a UNC name. The UNC name is simply

`\\filer\`*`username`*



NOTE: Using the `cifs.home_dir` option to create home directories is useful only when users do not need to read or write other users' home directories. Since each user can only access his or her own directory, when users must access the directories of other users, create an additional share to the *homes* directory and assign each user the appropriate access permissions to other users' home directories.

User name longer than the maximum-length share name

Because share names are truncated to 12 characters, the share name may show a truncated version of the user's account name. For example, consider the 13-letter name "administrator". From the filer's point of view, the home directory for "administrator"

has a name that exactly matches the account name "administrator", but use of that directory is offered under the truncated share name "administrato".

(If there were users "administrator" and "administrator1", each would see an offer for his or her own home directory, but both would see it as an offer to share the name "administrato". Thus each gets the correct directory, even though the share names appear the same.)

Procedure for setting up the home directories

The following is a step-by-step procedure for setting up home directories for CIFS users:

1. Enable the home directory option.

   Enter the following command:

   `options cifs.home_dir` *`homedirpath`*

   This creates the directory. For example, if the directory that contains the various home directories is */homes*, the command would be:

   `options cifs.home_dir /homes`

2. Set up a directory for each user.

   Within the directory specified by the `cifs.home_dir` option, create a directory for each user. Each directory name must exactly match the user's login name. Also, make each user the directory's owner.

When you display the filer's list of shares, the */homes* directory appears in the shares list, which is mapped to the share name CIFS.HOMEDIR, as shown in Figure 7-1:

```
toaster> cifs shares
          Name            Mount Point         Description
          C$              /                   Remote
          Administration
                          rwx hawleyr
                          rwx engineering{g}
          patterns        /patterns           patterns
                          rwx engineering{g}
          CIFS.HOMEDIR    /homes              Home Directories
                          rwx everyone
```

Figure 7-1. cifs shares output

Limits on CIFS open files, sessions, and shares

The filer is subject to limits on files access through CIFS as summarized in Table 7-1.

Table 7-1. CIFS limits by filer memory configuration

	64 MB	128 MB	192 MB	256 MB
Maximum number of users	128	523	1,024	2,048
Maximum number of open files	2,560	10,240	20,480	40,960
Maximum number of locked files	2,048	4,096	6,144	8,192
Maximum number of shares	1,000	1,000	1,000	1,000

Choosing the method for deciding access rights

When a CIFS user tries to obtain access to a file on the filer, the filer uses one of two methods for determining a user's file access rights: PC style and UNIX style. The two methods are offered because share-level ACLs do not work properly with the UNIX-style method but do work properly with the PC-style method. You choose the method when you use `cifs setup` after setting up CIFS for the first time. If you want to change the method, run the `cifs setup` command, which prompts you for the method. This section explains why you can choose between two methods and how you select the method based on the mix of users at your site.

Selecting the method for deciding access rights

This section provides examples comparing the advantages of PC style and UNIX style, depending on the needs of the users. The method affects only CIFS users.

PC style

If the filer uses PC style, when a CIFS user tries to obtain access to a file, the filer checks the share-level ACL before checking the UNIX-style permissions. (The filer gives each file UNIX-style permissions upon file creation.) The following is an example that illustrates how the filer combines the information from the share's ACL and the file's UNIX-style permissions to decide the users' file access rights.

In this example, the shared directory is */home/hr-users*; the group "hr" consists of "chris" and "laura." The ACL for */home/hr-users* is as follows:

```
HR-USERS    /home/hr-users    Sample HR Share
                 r-x hr
                 r-x barbara
                 rwx laura
                 rwx chris
```

The following explains the information contained in the ACL:

- All members of "hr" have read and execute rights. The group "hr" is defined in the filer's */etc/group* file.
- In addition to having read and execute rights, laura and chris can write to */home/hr-users*.
- Barbara does not belong to "hr," but she has read and execute access.
- Users outside of "hr" and excluded from the ACL have no access at all.

When a user attempts to write to a file within the share, the filer first checks whether the user is accorded the right to write by the share ACL. If a user doesn't have the right to write according to the ACL, the filer rejects the attempt to write to */etc/hr-users*. Otherwise, the filer checks to see whether the file's UNIX-style permission allows the user to write. (File permissions were inherited from the file's parent directory when the file was created.) For example, barbara's attempt to write fails because the ACL doesn't give her the write privilege. However, laura has the right to write, so her attempt to write depends on the UNIX-style permissions of the file, as follows:

- If laura is the file owner, whether she can write depends on the file's write permission that is applicable to the file owner.
- If laura is not the file owner, whether she can write depends on the file's write permission that is applicable to the group. (The group permissions apply to everyone other than the file owner. The filer disregards the GID.)

PC style offers these advantages:

- ACLs work.
- You do not need a UNIX client to manage file permissions. You can leave the file permissions at their default settings, and use

share-level acls to control the access rights of CIFS users. By default, the permissions of the */home* directory are as follows:

```
drwxrwx---
```

If your site requires private directories for CIFS users, use the NetApp Permissions Tool (available from NOW) on a PC to modify permissions for the private directories even when you use PC style. For example, if */home/hr-users* contains chris' private directory, */home/hr-users/chris_home*, to prevent users other than chris from reading or writing to it, change the permissions for *chris_home* to the following:

```
drwx------
```

UNIX style

When the filer uses UNIX style, when a CIFS user tries to obtain access to a file, the filer checks the UNIX-style permissions before checking the share-level ACL. (The filer gives each file UNIX-style permissions upon file creation.)

When the UNIX style is used, the GID of a file is significant. That is, the filer compares the GID of the file with the GIDs of a user to determine if the user belongs to a particular group. The filer applies the group permissions only to members of the group as specified in the */etc/group* file.

If you are using UNIX-style permissions, in general it is best to set the ACL to everyone and rely on UNIX permissions to control access.

Use the following guidelines to determine whether the UNIX style is suitable for your site:

- If the CIFS users can also nfs-mount directories and are familiar with UNIX-style file permissions, the users can use UNIX clients to manage the permissions of the files they create. For example, if chris creates a file named */home/hr-users/policy*, she

can set the UNIX permissions to prevent users other than herself from reading and writing to the file.

- If you are satisfied with the way UNIX systems define groups, use the UNIX style to minimize the work required for managing access rights. For example, if you want all members of the "hr" group to have the same rights, in the ACL, assign read, write, and execute rights to the group "hr." In this way, you can use UNIX permissions for each file exclusively for managing access rights; you need to add entries to ACL only if you want to limit the access rights for particular users of the "hr" group.

Changing UNIX-style permissions for files in a CIFS share

In addition to using the NetApp Permissions Tool, you can use the `CIFS chmod` command set the UNIX-style permissions of a designated file or directory within a share. The syntax is:

`CIFS chmod` *`share mode path`*

The arguments are as follows:

share The name of a CIFS share.

mode The access mode to be applied to the file or directory (as described in the tables below). You can specify the mode in either absolute format or symbolic format.

path The path, relative to the root directory of the specified share, to the file or directory whose access is to be set.

Absolute Mode. A mode specified in absolute format is an octal number. The full range of options is represented by a number having four octal digits, but for most purposes a 3-digit number suffices (with an implied leading 0).

The significance of digits in the various positions is the same as in UNIX systems, as shown in Table 7-2

Table 7-2. Absolute mode values

Value	Position			
	Execution mode[a]	Owner	Group	Others
4	Set-user-ID-on-execution	Read	Read	Read
2	Set-group-ID-on-execution	Write	Write	Write
1	Sticky bit	Execute/ Search	Execute/ Search	Execute/ Search

a. These bits may be significant when a UNIX system executes the file, but they do not affect access rights.

For example, the mode 644 gives the owner read and write permission and gives group members and others read permission.

The mode 755 gives the owner read, write, and execute permissions, and group members and others read and execute but not write permission.

The mode 0 denies every permission to all.

Symbolic mode. A mode specified in symbolic format is a comma-separated list of expressions. Each expression is formed from up to three components, as shown in Table 7-3.

Table 7-3. Symbolic mode components

Who		Operation		Permission	
u	User	+	Add	r	Read
g	Group	-	Remove	w	Write
o	Others	=	Set	x	Execute / search
a	All (that is, ugo)			s	Set-user-ID-on-execution and Set-group-ID-on-execution
				t	Sticky

For example, the mode expression `go-w` denies write permission to group and others. The mode expression `u=rwx,go=rx` gives the user (that is, the file's owner) permission to read, write, and execute, and gives members of the user's group and others permission to read and execute (but not write). It is thus equivalent to 755.

An expression that uses = but omits the permission clears all permissions. For example, the mode expression `go=` turns off all access for group and others, and is equivalent to `go-rwxst`.

An expression that does not indicate *who* applies to all. For example, the mode `+x` adds execute permission to all, and is thus equivalent to `a+x`.

Changing the ownership of files in a share

The `cifs chown` command changes the owner of a file or directory. The syntax is:

`cifs chown` *`share owner path`*

The arguments are as follows::

share The name of a CIFS share.

owner The user name of the new owner.

path The path, relative to the root directory of the specified share, to the file or directory whose access is to be set.

Managing access rights for users or groups

This section describes how to assign access rights and remove an entry in a share's ACL. It also describes how to display the rights of users and groups.

Assigning access rights

After you create a share, define the users' or groups' access rights to the share. To assign access rights, use the `cifs access` command with the following syntax:

```
cifs access share [ -g ] user|group rights
```

The `-g` flag is used only if the access rights are defined for a group.

The rights are defined as `r`, `w`, and `x`, which mean read, write, and execute, respectively. To deny a right to a user, use a minus sign (-). The following are some examples:

```
cifs access library -g engineering rwx
cifs access library joed r--
```

The group "everyone" is reserved. When it is used in an ACL, the group "everyone" means every CIFS user. For example, to give every CIFS user read, write, and execute rights to the *library* share, enter:

```
cifs access library -g everyone rwx
```

Removing an entry from a share's ACL

If a group or a user no longer exists and you want to remove the corresponding entry in an ACL, use the following command syntax:

```
cifs access -delete share user|group
```

The following are some examples:

```
cifs access -delete library engineering
cifs access -delete library joed
```

Displaying access rights to a file

To display access rights to a file,use the `cifs rights` command. It has the following syntax:

cifs rights *share user path*

where:

share is a share,

user is the login of a user, and

path is the path of a file relative to the root directory of share.

This displays the rights that the user has for the file with the path name *path* relative to the root of the share *share*, for example:

```
toaster> cifs rights C$ hawleyr /etc/rc
```

would produce the output:

```
PC style Access Control is being used
user is a member of group eng
access rights for share LINK = r-x | r-x | r-x
access rights for share ENG-USERS = rwx | rwx | rwx
access rights for share HOME = r-x | r-x | r-x
access rights for share C$ = rwx | rwx | rwx
full path = //etc/rc
File Unix permissions user= Administrator(root)
group= well
file perm =rwx | rwx | ---
PC rights RWX
UNIX rights ---
```

Using CIFS oplocks

Oplocks (opportunistic locking) enable the redirector on a client machine in certain file-sharing scenarios to perform client-side caching of read-ahead, write-behind, and lock information. A client can therefore work with a file (read or write it) without regularly reminding the server that it needs access to the file in question, thus improving performance by reducing network traffic.

Under some circumstances, if a process has an exclusive oplock on a file and a second process attempts to open the file, the first process must relinquish the oplock and access to the file. The redirector must then invalidate cached data and flush writes and locks, resulting in possible loss of data that was to be written. Oplocks are therefore best used in an environment where most operations are reads rather than writes.

Any application that has write-cached data can lose that data under the following set of circumstances:

- It has an exclusive oplock on the file.
- It is told to either break that oplock or close the file.
- During the process of flushing the write cache, the network or target system generates an error.

The cache itself does not have any error handling - the applications do. When the application makes a write to cache, it always completes. If the cache in turn makes a write to the target system over a network - it must assume that the write will complete, because if it does not the data is lost.

CIFS oplocks on the filer are on by default.

You might turn CIFS oplocks off under either of the following circumstances:

- You are using a database application whose documentation recommends that oplocks be turned off.
- You are handling critical data (you have a good networks but your company cannot afford data loss).

Otherwise, you would leave CIFS oplocks on.

You turn off CIFS oplocks with the following options command:

```
options cifs.oplocks.enable off
```

You turn on CIFS oplocks with the following options command:

```
options cifs.oplocks.enable on
```

For additional information about oplocks, consult the Microsoft Knowledge Base at *http://www.microsoft.com/kb*



NOTE: You can turn CIFS oplocks on or off at individual clients. Turning CIFS oplocks On at the filer does not override any client-specific settings. Turning CIFS oplocks Off at the filer disables all oplocks to or from the filer.

Using NetBIOS scope IDs

The NetBIOS scope ID feature provides a naming service for the NetBIOS over TCP/IP (NBT) module. The NetBIOS scope ID feature is not compatible with DNS. The scope ID is a case-sensitive character string that is added to the end of a NetBIOS name.



NOTE: Microsoft recommends that you not use NetBIOS scope ID if you are not currently using it.

You should not enable scope ID for the filer unless you are already using it for other machines because you could unintentionally isolate the filer from clients that do not have the same NetBIOS scope ID.

If you are not using DNS, NetBIOS scope ID enables you to limit traffic to only those machines with the same NetBIOS scope ID.

You can use this feature to isolate a group of computers on a network so they can communicate only with other computers with the identical NetBIOS Scope ID, or to build virtual subnetworks (groups of machines that see each only other). By default, the scope ID is a null string, but if the filer is to run in a NetBIOS scope other than the default one, its scope ID must be set to the ID of that scope. The scope ID can be changed only when CIFS is not running.

To enable NetBIOS scope ID, use the following command:

```
options cifs.scope.id scopeID
```

The default is blank: (no scope ID)

To disable NetBIOS scope ID, set the option to NULL or blank with the following command:

```
options cifs.scope.id
```

Displaying CIFS statistics

The `cifs stat` command displays statistics about CIFS operations that take place on your filer.

You can use the `cifs stat` command in two forms:

- If a time interval is specified, the command displays statistics at the specified intervals. For example, the following command displays statistics every second:

  `cifs stat 1`

  Figure 7-2 shows an example of the output of `cifs stat 1`.

```
toaster> cifs stat 1
  GetAttr    Read   Write    Lock   Open/Cl   Direct   Other
      175     142      54      70       254      643      50
      232      76     123      44       321      154      17
      152     120      34     111        12      435      76
```

Figure 7-2. cifs stat output

- If no time interval is specified, the `cifs stat` command displays CIFS statistics that have accumulated since the last reboot. The `cifs stat` display is for diagnostic purposes.

Displaying CIFS session information

To display information about connected users, and the number of shares and open files for each user, enter the following command:

```
cifs sessions
```

To limit the information to a particular user or client name, specify the user name or client name in the command. The following is an example:

```
cifs sessions joed
```

Figure 7-3 shows a sample output from the `cifs sessions` command without a user name.

```
Server Registers as 'TOASTER' in group NT-DOMAIN
WINS Server: 172.20.0.4
PC style Access Control is being used
Using domain controller NT-DOMAIN-PDC for
authentication
=====================================
PC (user)           #shares       #files
SMITHPC (jsmith)    1             1
PETERSPC (mpeters)  2             3
```

Figure 7-3. cifs sessions output

Stopping CIFS service

You can stop CIFS service by using the `cifs terminate` command. Always use this command before you reboot or power off the filer. The `halt` command automatically invokes the `cifs terminate` command

While the `reboot` command also stops CIFS service, the `cifs terminate` command provides a time delay during which users can save their open files before the disconnection. Changes that have not been saved to disk are lost if the CIFS client has an open file when it is disconnected from the filer. The default time delay is five minutes. You can specify the time delay in minutes by using the -t option. For example, to terminate CIFS service after 10 minutes, enter:

```
toaster> cifs terminate -t 10
```

The `cifs terminate` command also displays an alert message on CIFS clients that warns them of the pending shutdown. Windows 95 and Windows for Workgroup clients must have the WinPopup program configured before they can display the alert message. Figure 7-4 is an example of the message.

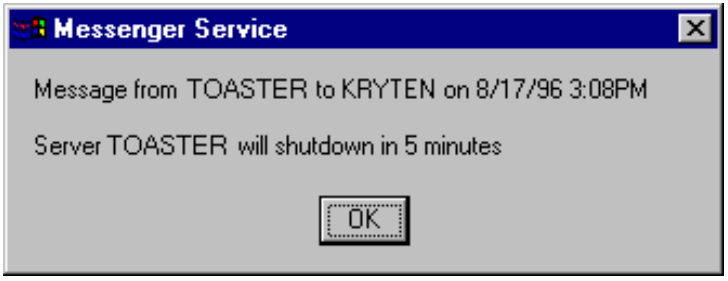


Figure 7-4. Filer shutdown message displayed on an NT client

If you want to terminate CIFS service for a particular client, specify the name of the computer in the command. For example, the following command will terminate CIFS service to a computer named "peterspc" after 10 minutes:

```
toaster> cifs terminate PETERSPC -t 10
```

Figure 7-5 shows an example of the output of the `cifs terminate` command.

```
toaster> cifs terminate
There are currently 35 CIFS users that have 37 open files
Disconnecting while files are open may cause data loss!!
How many minutes should I wait? [5]:
minutes left 4
minutes left 3
minutes left 2
minutes left 1
CIFS shutting down
toaster>
```

Figure 7-5. cifs terminate output

Pressing Ctrl-C before the end of the countdown cancels the `cifs terminate` command.



NOTE: The `cifs terminate` command disables CIFS service only between the time you enter the command and the next reboot. After each reboot, if your filer is licensed and configured to run CIFS, the filer will automatically start CIFS service. If you want to prevent CIFS from restarting after a reboot, remove the file */etc/cifsconfig.txt* from the filer or reboot without running */etc/rc*.

Restarting CIFS service

To restart CIFS service (even if you disabled */etc/rc*), use the `cifs restart` command. For example:

```
toaster> cifs restart
CIFS server is registering...
CIFS server is running.
```

Chapter 8: HTTP administration

This chapter describes how to perform the following tasks:

- starting HTTP service
- using the HTTP virtual firewall
- enabling virtual hosting
- modifying configuration files that are used by the HTTP daemon
- displaying HTTP connection information and statistics.



NOTE: You can use the filer as an HTTP server only if you have purchased the license for HTTP. Without the license, you can use an HTTP client (Web browser) only to display the filer's on-line help files and use FilerView. For more information about adding a license, refer to "Software licenses" on page 104.

Enabling HTTP service

To start HTTP service on your filer, follow these steps:

1. Enable the `httpd` daemon by entering the command:

   `options httpd.enable on`

2. Use the following command syntax to specify the root directory that contains the files and directories to be read by HTTP clients:

   `options httpd.rootdir` *`directory`*

   For example, if the root directory is */home/users/pages*, enter:

   `options httpd.rootdir /home/users/pages`

3. If you want to limit the size of the log file */etc/log/httpd.log* to other than the default of 2147483647 bytes (2 GB - 1 byte), use the following command:

   `options httpd.log.max_file_size` *`bytes`*

4. Make a copy of */etc/httpd.mimetypes.sample* and name the copy */etc/httpd.mimetypes*. If the */etc/httpd.mimetypes* file is missing, the HTTP client uses the information in */etc/httpd.mimetypes.sample*.



NOTE: If you want these options to remain active after rebooting, you must add them to the */etc/rc* file.

The procedure for enabling HTTP service is now complete, and clients can display text files under */home/users/pages* by using a Web browser. If the filer will transfer files other than text files (for example, image files and audio files), follow the instructions in "Specifying MIME Content-Type values" on page 190 to configure your filer so that the appropriate MIME Content-Type header is sent with each file transferred.

To test the filer's HTTP service, follow these steps:

1. Create an HTML file in the root directory for HTTP. For example, create a file named *myfile.html* in the root directory, which is */home/users/pages* (assuming that the HTML root directory is */home/users/pages*).

2. Start a Web browser on a client and specify the URL of the html file in the browser.

   For example, if your filer's name is "toaster" and the root directory for HTTP is */home/users/pages*, enter this URL:

   ```
   http://toaster/myfile.html
   ```

The path component of the URL is a path name relative to the HTTP root. Do not specify the complete path name to the file in the URL.



NOTE: If the URL names a directory (for example, *http://toaster/home/pages*), the filer automatically tries to transfer the file *index.html* within the directory. If *index.html* does not exist, the filer returns "Error 404. No such file or directory."

Protecting web pages with passwords

You can restrict access to a specified directory tree so that only specified users or groups have access to it. A directory tree contains web pages.

Password protection involves three configuration files:

- */etc/httpd.access*
- */etc/httpd.passwd*
- */etc/httpd.group*

The /etc/httpd.access file

/etc/httpd.access contains directives that govern authentication for each directory tree. The filer supports the following directives:

- directory
- AuthName
- require user
- require group

These directives are compatible with the Apache web server directives, but the filer ignores all other directives.

The Directory directive

This directive specifies a directory tree to be protected and encloses all other directives. The syntax of the `directory` directive is:

```
<Directory directory_name>
directive ...
</Directory>
```

The AuthName directive

This directive specifies a "realm," which is an alias for the directory and appears instead of the directory name in the browser's password dialog box when a user tries to access the directory. Whatever follows `AuthName` is the name of the realm. The name of the realm can contain spaces. The syntax of the `AuthName` directive is:

`AuthName` *`realm name`*

The require user directive

You use this directive to specify the users who can access the directory. The syntax of the `require user` directive is:

require user *user_id* [, *user_id*, ...]

The require group directive

You use this directive to specify the groups who can access the directory. The syntax of the `require group` directive is:

`require group` *`group_id`* [, *`group_id`*, `...`]

The /etc/httpd.passwd file

/etc/httpd.passwd contains user_id and encrypted-password pairs. The pairs have the format:

`user_id`:*`encrypted_passwd`*

The pairs are copied in from a machine on which the user has a password.

The /etc/httpd.group file

/etc/httpd.group contains a group_id and a list of user_ids in that group in the following format:

`group_id`: *`user_id`* [*`user_id`* `....`]

The lists are copied in from a machine that has a similar list.

Web page protection examples

The following */etc/httpd.access* file restricts access to */home/htdocs/private/bob* to only user dole:

```
<Directory /home/htdocs/private/henry>
AuthName polard Private Stuff
<Limit GET>
require user polard
</Limit GET>
</Directory>
```

Note that <Limit GET> and </Limit GET> are not supported, but the file might have been imported from an Apache or NCA server. To be used on a filer, the file does not need to be edited to remove the Limit GET directive; the filer ignores the directive and all other directives not mentioned in this chapter.

The following */etc/httpd.access* file:

```
<Directory /home/htdocs/private/funny>
AuthName Social commentary
require group stooges
</Directory>
```

in conjunction with the following entry in */etc/httpd.group*:

```
stooges: curly larry moe shemp
```

restricts access to the */home/htdocs/private/funny* directory tree to the group “stooges,” which consist of the following user IDs:

“curly”, “larry”, “moe”, “shemp”

Using the HTTP virtual firewall

The HTTP virtual firewall feature helps you maintain security on your filer. You restrict HTTP requests by marking the subnetwork interface over which they arrive as "untrusted." An untrusted interface provides only HTTP access to your filer on a read-only basis.

You would mark an interface untrusted if it meets all of the following conditions:

- You know you are going to service HTTP requests over that interface.
- You don't want to allow requests through protocols other than HTTP.
- You want to restrict access to the filer through that interface to read-only access.

By default, a subnetwork interface is trusted.

You mark an interface as untrusted or trusted by setting an option to the `ifconfig` command. For example, the following command:

`ifconfig f0 untrusted`
marks the f0 interface as untrusted.

The following command:

`ifconfig f0 trusted`
marks the f0 interface as trusted.

Using virtual hosting

Virtual hosting enables a filer to respond to requests directed to more than one IP address through a single physical interface. This means that a filer with only one physical interface can host several IP addresses. Virtual hosting enables, for example, an internet provider to host several web sites but have only one physical interface.

An HTTP server can use the destination IP address of an incoming HTTP request to find the directory that contains the HTTP pages belonging to the virtual host.

To enable virtual hosting, you direct requests by putting entries in the configuration file */etc/httpd.hostprefixes* and you map virtual host addresses to the virtual host interface with the `ifconfig` command.

Using the /etc/httpd.hostprefixes file

The */etc/httpd.hostprefixes* file has the following format:

`prefix` [*`host-name-or-address ...`*]

where

prefix is a subdirectory of the HTTP root directory that you specify with the `option http.rootdir` command.

host-name-or-address is an HTTP host name or an IP address. You can have more than one of each.

For example, the line:

`/customer www.customer.com 192.225.37.102`

means that an HTTP request which comes for the interface with address 192.225.37.102, or with an HTTP 1.1 Host: header specifying *www.customer.com,* is directed to */customer,* and the requestor cannot get a file outside the directory */customer.*

If the HTTP server receives an HTTP request that is destined for one of its virtual host IP addresses, in this example 192.225.37.102, the destination IP address is used to select the virtual host root directory from the */etc/httpd.hostprefixes* file.

Mapping virtual host addresses

To map virtual host addresses to the virtual host interface, use the `ifconfig` command as follows:

- You add a new IP virtual host address mapping with the following command:

  ```
  ifconfig vh alias address
  ```

  where *address* is an IP address.

  The use of the `vh` interface indicates to the system that you are adding a virtual host address rather than adding an IP alias address to a network interface.

- You delete virtual host addresses with the following command:

  ```
  ifconfig vh -alias address
  ```



NOTE: If you need to create a virtual subnet, with many contiguous addresses, the IP address can be a subnet address.

Specifying MIME Content-Type values

You can configure the filer to send the appropriate MIME (Multipurpose Internet Mail Extensions) Content-Type header in each response to a get request. The header shows the MIME Content-Type value of the file, which tells the browser on the client how to interpret the file. For example, if the MIME Content-Type value shows that the file being transferred is an image file and the client is configured properly, the browser can render the image by using a graphics program.

The filer determines the MIME Content-Type value of a file by mapping the file name suffix (for example, *.gif*, *.html*, *.mpg*) according to information in */etc/httpd.mimetypes*.



NOTE: On a Windows 95 or Windows NT 4.0 client, the file name */etc/httpd.mimetypes.sample* is not displayed in its entirety. By default, the Explorer displays the file name as */etc/httpd.mimetypes*. If you are using Windows, from the Explorer View menu, select Options, then the View tab and, in the dialog box, make sure that there is no check mark in the box next to "Hide MS-DOS file extensions for file types that are registered."

To modify MIME type mappings or to add MIME types, edit the entries in */etc/httpd.mimetypes*. Entries are in this form:

```
# An optional comment.
suffix   Content-Type
```

Lines preceded by the # sign are comments. The following are sample entries:

```
# My clients' browsers can now use
# PICT graphics files.
pct        image/pict
pict       image/pict
```

According to these entries, files whose names ended with *.pct*, or *.pict* are mapped to the MIME Content-Type value of "image/pict." The file name suffix is case-insensitive. The first

field in the Content-Type value describes the general type of data contained in the file; the second field is the data subtype, which shows the specific format in which the data is stored. If the browser on the client is configured to start a graphics program as a helper application, the user can view a file named *file.pict* as a graphics file on the client.

For more information about MIME, refer to RFC 1521.

Translating URLs

You can specify that the filer's response to an HTTP request be dependent on the URL. For example, you can configure the filer to redirect a particular request to a specific directory, or to prevent access to a particular directory that is specified in the URL. How the filer maps its responses to urls is defined in a configuration file named */etc/httpd.translations*.

Each entry in */etc/httpd.translations* contains up to three fields in this form:

```
rule template result
```

The `rule` field defines the response of the filer to a request. The `template` field specifies a component of a URL. The meaning of the `result` field depends on the rule as described in the following section.

Translation rules supported by the filer

This section explains the meanings of rules. It also describes the format required for each type of entry in */etc/httpd.translations*.

The map rule

The map rule specifies that if a component of a URL matches the template, the request is mapped to another directory on the same host as defined in the `result` field. For example, if the following entry is in */etc/http.translations*:

```
Map /image-bin/* /usr/local/http/images/*
```

any requests to a URL containing the */image-bin* directory will be mapped to the */usr/local/http/images* directory.

The redirect rule

The redirect rule specifies that if a component of a URL matches the template, the request is redirected to the URL defined in the `result` field. For example, if the following entry is in */etc/http.translations*:

```
Redirect /cgi-bin/* http://cgi-host/*
```

the filer redirects CGI requests to another HTTP server named "cgi-host." This is essential for calls to *cgi-bin* because the filer does not execute them.



NOTE: The result field for the redirect rule must be specified as a complete URL beginning with *http://* and the host name.

The pass rule

The pass rule specifies that if a component of a URL matches the template, the filer will accept the request, process the request as is, and disregard other rules. For example, if the following entry is in */etc/http.translations*:

```
Pass /image-bin/*
```

the filer will process the request for any URL containing */image-bin* as is, even though there is another rule specified as follows:

```
Map /image-bin/* /usr/local/http/images/*
```

If the pass rule includes the `result` field, the filer will accept the request, process the request by using the URL defined in the `result` field, and disregard other rules.

The fail rule

The fail rule specifies that if a component of a URL matches the template, the filer will deny access to that component of the URL and disregard other rules. For example, if the following entry is in */etc/http.translations*:

```
Fail /usr/forbidden/*
```

the filer will not provide access to the */usr/forbidden* directory.

How the filer processes rules

The filer processes the rules defined by */etc/http.translations* in the order they are listed, and applies the rule if the URL matches the template. However, the filer stops processing other rules once it applies a pass or fail rule.

In the `template` or `result` field of an */etc/http.translations* entry, you can use asterisks (*) as wild card characters as follows:

- In the `template` field, the wild card character matches zero or more characters, including the slash (/) character.
- In the `result` field, the wild card character represents the text expanded from the match in the `template` field. Include the wild card character in the `result` field only if a wild card character has been used in the `template` field.
- If multiple wild card characters are used, the first one in the `result` field corresponds to the first one in the `template` field, the second one in the `result` field corresponds to the second one in the template field, and so on.

The following is an example showing how a wild card character is used:

```
# Redirect all cgi requests to my cgi server
Redirect /cgi-bin/*    http://cgi-host/cgi-bin/*
```

This redirect rule specifies that all CGI requests will be redirected to another host named “cgi-host.” Suppose the filer named “toaster” receives the following requests:

```
http://toaster/cgi-bin/displayfares
http://toaster/cgi-bin/displaydates
```

the filer will expand the wild card character to “displayfares” and “displaydates” and redirect the requests to the host named “cgi-host.” To the client, the results of these requests are the same as the results of these requests:

```
http://cgi-host/cgi-bin/displayfares
http://cgi-host/cgi-bin/displaydates
```

Displaying HTTP connection information

Read the */etc/log/httpd.log* file if you are interested in the following types of information for each HTTP connection:

- IP address of HTTP client.
- The name of authorized users (if the requested page is protected) making requests. The names are in the */etc/httpd.passwd* file. If the page is not protected, dashes appear instead of a name.
- Time of connection in `dd/mm/yy:hh:mm:ss` format (gmt is used).
- Request line from connecting host (for example, get `/netapp.html`).
- STATUS code returned by the server (as defined in the HTTP 1.0 specifications).
- TOTAL bytes sent in response by the filer, not including MIME header.

Figure 8-1 shows an example of the */etc/log/httpd.log* file.

```
192.9.77.2  - - [26/Aug/1996:16:45:50] "GET /top.html" 200 1189
192.9.77.2  - - [26/Aug/1996:16:45:50] "GET /header.html" 200 531
192.7.15.6  - - [26/Aug/1996:16:45:51] "GET /logo.gif" 200 1763
198.9.200.2 - - [26/Aug/1996:16:45:57] "GET /task/top.html" 200 334
192.9.20.5  authuser [26/Aug/1996:16:45:57] "GET /task/head.html"
200 519
```

Figure 8-1. /etc/log/httpd.log example

Displaying HTTP statistics

The `httpstat` command displays statistics about HTTP operations on the filer. The `httpstat` command displays four types of statistics:

HTTP gets	Successful requests for files.
HTTP badcalls	Requests for nonexistent files.
open conn.	The number of HTTP connections currently open.
peak conn.	The largest number of simultaneous HTTP connections since the filer was booted or since the `-z` option was used.

This is the syntax for the `httpstat` command:

`httpstat [ -t|-z ][ interval ]`

Without any arguments, `httpstat` displays HTTP statistics accumulated since the last reboot or since the last time the -z argument was used. The `-z` argument resets both the gets and badcalls counters. The `-t` argument displays statistics since the last filer reboot.

You can specify the interval (in seconds) at which the filer displays the statistics.

Figure 8-2 shows an example of `httpstat` output:

```
toaster> httpstat
HTTPD statistics:
gets         badcalls         open conn.   peak conn.
451          11                5            17
```

Figure 8-2. httpstat output

Chapter 9: **Managing disks**

The filer uses RAID Level 4 to ensure data integrity even when one of the disks fails. The file system design, together with the support for RAID, optimizes filer performance and allows you to expand the filer's disk storage capacity one disk at a time.

The RAID disk structure on your filer is factory-installed; you don't need to configure the storage system before using the filer. However, when planning to optimize disk space or when replacing or adding disks, you must be familiar with how the filer uses its disk array. This chapter provides the following types of information necessary for you to get the most out of the installed disks and to plan for disk space expansion:

- functions of data, parity, and hot spare disks in the filer
- disk numbering in the filer
- usable disk space versus physical disk space
- choosing disk size when replacing or adding a disk
- increasing total disk capacity by replacing small disks
- replacing a disk with or without a hot spare
- recovering from disk failures

- managing disk space by setting disk quotas and the maximum number of files
- using the `df` command

Data, parity, and hot spare disks

In the RAID file system, different disks play different roles. Most of the disks are *data disks*, and one disk in the array is the *parity disk*. The parity disk must be the same size as or larger than the largest data disk. After a data disk failure, the filer uses the parity disk in conjunction with the other data disks to reconstruct data previously on the failed disk. While the filer can continue to function, it automatically shuts down after 24 hours to encourage you to replace the failed disk. You can change the amount of time from 24 hours to another value using the `raidtimeout` option to the `options` command.

In addition to data and parity disks, you can install one or more *hot spare disks* in the filer. A hot spare disk is not part of the RAID disk array and does not contain file system data. After a disk failure, the filer automatically rebuilds data on the hot spare and adds the hot spare to the RAID array, thus avoiding a system shutdown.

For more information on how the filer recovers from disk failures by using the parity disk or hot spare disk, refer to “Disk failures” on page 227.

How disks are numbered

Disks in the filer can be identified by different numbers. This section explains what each type of disk identifier means.

RAID disk number

When the file system is initialized, the filer assigns a RAID disk number to each parity or data disk. The RAID disk number identifies the disk's position in the RAID array. (Numbers are not assigned to hot spares.) RAID disk 0 is always the parity disk; RAID disk 1 is the first data disk, RAID disk 2 is the second data disk, and so on.

After a disk has been added, removed, moved, or reconstructed, RAID disk numbers are reassigned. A particular RAID disk number may then correspond to a different physical disk, disk sics ID, or disk ID number.

The RAID disk number is used and maintained internally by the filer software. No filer commands require the number as an argument. However, a few screen messages display RAID disk numbers to refer to disks. For more information about how to locate a disk referenced by a RAID disk number in a screen message, refer to "Locating a disk using different identifiers" on page 202.

Disk SCSI address

You can identify a disk by its SCSI address, which is a combination of the disk's SCSI adapter number and the disk's SCSI ID number on the adapter. For example, disks attached to SCSI adapter 0 are numbered 0.0 through 0.6, disks attached to SCSI adapter 9a are numbered 9a.0 through 9a.6, and so on. For more information about determining a disk's SCSI address, refer to the hardware guide for your filer model.

While no filer commands require the SCSI address as an argument, most screen messages use SCSI addresses to refer to disks. This allows you to locate the disks quickly.

Disk ID number

At boot time, the filer probes to discover all available disks, and assigns each disk a unique disk ID number, starting with disk ID 0. Disk ID numbers are independent of the SCSI ids, and change after each reboot. Some filer commands (for example, the `raid` command) require disk ids as arguments. The next section describes how to locate a disk referenced by a disk ID number.

Locating a disk using different identifiers

Use the `sysconfig -r` command to display disk information. The command output lists the three types of disk identifiers: RAID disk number, disk SCSI address, and disk ID number. The command sorts disks according to RAID disk numbers.

The `sysconfig -r` command is useful when you need to locate a disk referenced by the RAID disk number in a screen message. Figure 9-1 shows an example of the `sysconfig -r` output. In this example, you can see that the "parity disk," (RAID disk 0) has disk ID 9, and is attached to SCSI adapter 1 with SCSI ID 2 because its SCSI address is 1.2.

```
toaster% sysconfig -r
RAID Disk DISK_ID#  HA.SCSI#     Used (MB/blks)  Phys (MB/blks)
--------- --------  --------     --------------  --------------
parity        9        1.2        4000/8192000    4095/8386728
data 1        0        0.0        1000/2048000    1033/2117024
data 2        2        0.2        2000/4096000    2040/4178872
data 3        3        0.3        2000/4096000    2040/4178872
data 4        4        0.4        2000/4096000    2049/4197520
data 5        5        0.5        2000/4096000    2049/4197520
data 6        6        0.6        2000/4096000    2049/4197520
data 7        7        1.0        2000/4096000    2048/4194680
data 8        8        1.1        2000/4096000    2048/4194680
data 9        1        0.1        2000/4096000    2040/4178872
```

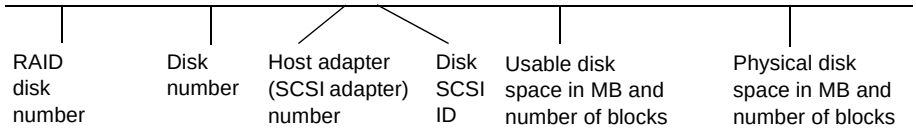


Figure 9-1. Sample sysconfig -r output

The `sysconfig -r` command is also useful when you need to determine the disk ID to use in a filer command. Suppose you want to remove the spare disk. The `raid remove` command takes a disk ID number as its argument. From the `sysconfig -r` output in Figure 9-1, you can see that the spare disk's disk ID is 10. So, before you remove the disk from the disk slot, enter the `raid remove` command with the disk ID as follows:

```
raid remove 10
```



NOTE: Make sure that any previous `raid` command has completed before you issue another `raid` command.

Usable space on each disk

This section explains why a disk's usable space can be different from its physical space. The information here applies to data, parity, and hot spare disks.

Disks from different manufacturers may differ slightly in size even though they belong to the same size category. Table 9-1 is an example showing the differing capacities of 2-GB disks manufactured by Seagate Technology and Digital Equipment Corporation, both of which are supported by the filer.

Table 9-1. Comparing disk space between disks

Seagate ST12400N	Digital DSP3210S
2040 MB (4178872 blocks)	2049 MB (4197520 blocks)

To make sure that one disk can be replaced with another in the same size category, a filer shipped with Release 2.1 or later automatically "right-sizes" its disks. That is, it rounds off the disk capacity and divides disks into 1.0-GB, 2.0-GB, and 4.0-GB classes. For example, both disks listed in Table 9-1 belong to the 2.0-GB class, but the filer only uses up to 2000 mb (4096000 blocks) on each disk. If the Digital DSP3210S fails, you can replace it with the Seagate ST12400N. All data originally on the Digital DSP3210S, which is slightly larger, can be reconstructed on the Seagate ST12400N.

Table 9-2 shows the usable disk space for each class of disk supported by the filer.

Table 9-2. Usable disk space for various classes of disks

1.0-GB disks	2.0-GB disks	4.0-GB disks	9.0-GB disks
1000 MB (2048000 blocks)	2000 MB (4096000 blocks)	4000 MB (8192000 blocks)	9000 MB (18432000) blocks

To display the space available for use and the physical size of each disk installed in the system, use the `sysconfig -r` command, which was described on page 202.

If you have upgraded your filer from a release before 2.1, the filer doesn't "right-size" the existing disks. Instead, it continues to use all the physical disk space. For example, before the upgrade, your filer contained a Digital DSP3210S disk. As shown in Figure 9-1, the physical size of this disk is 2049 mb (4197520 blocks). After the upgrade, the usable space on the disk remains 2049 mb (4197520 blocks).

If a disk has never been right-sized, it can only be replaced by another disk that *has the same number of blocks or more*. For example, to replace a Digital DSP3210S disk that was installed in the filer while it ran a pre-2.1 release, you must use a disk with at least 4197520 blocks.

While automatic disk right-sizing is not applied to existing disks in an upgraded system, it is applied to disks being *added* to the filer. Figure 9-2 illustrates how to determine the usable disk space in an upgraded system.

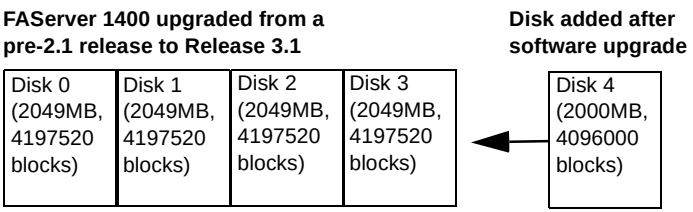


Figure 9-2. Disk space on a filer upgraded from a pre-2.1 release

After the software upgrade, the total amount of space available to the file system remains unchanged. The filer can use 2049 MB on each disk because the usable space on each disk is the same as the physical space. If you replace any disk on this system, the replacement must have at least the same number of blocks as the

old disk. If you *add* a 2.0-GB disk (Disk 4) to the filer after the software upgrade, the filer will "right-size" this disk to 2000 MB (4096000 blocks). When Disk 4 fails, you can replace it with any qualified disk in the 2.0-GB class.

In some cases the additional disks are not right-sized. If every disk in the array has the same size, and the disk being added has the same physical size as the existing disks, it is not right-sized. Figure 9-2 shows exactly this case. Here, if Disk 4 had the same physical size as the others, it would wind up with 4197520 blocks, the same as the others.

Figure 9-3 illustrates how usable disk space is computed on a new filer shipped with Release 3.1.

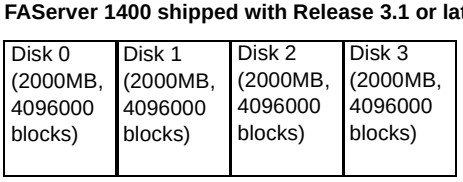


Figure 9-3. Disk space on a new filer shipped with Release 3.1 or later

Because of automatic right-sizing, the total amount of space available to the file system is 2000 MB x 4 = 8000 MB. You can use any qualified disk in the 2.0-GB class to replace any disk on this system.

Using `sysconfig -r`, shown in Figure 9-1 on page 202, you can compare the physical space and the usable space and can determine whether disks are right-sized.

Using disks of various sizes

You can use disks of various sizes in the same filer. If you use a hot spare in a filer, the hot spare should be the largest disk. The filer assigns the role of parity disk to the largest of the active disks (not counting the spare). Figure 9-4 shows a sample configuration.

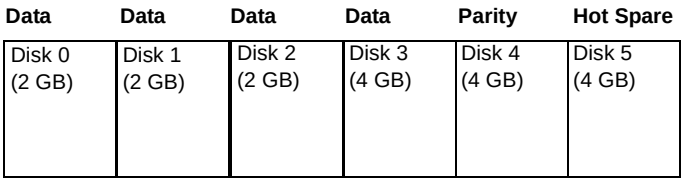


Figure 9-4. Sample disk configuration

The following subsections discuss the guidelines for choosing the size of a disk to be added to the filer or to be used as a replacement.

When replacing a failed disk

When you replace a failed disk, the physical space of the new disk must be equal to or greater than the *usable* space of the replaced one. The "Used" column of the `sysconfig -r` output shows the amount of usable space on each disk.

However, if the new disk is larger than the old disk, the system can use no more space than was available on the old disk. For example, if you replace a 1-GB disk with a 2-GB disk, the extra 1 gb on the new disk will not be available to the file system. Therefore, it is most cost-effective to replace a disk with one of the same size.

When adding a disk to the file system

When you add a disk that is larger than the current parity disk to the RAID file system, the new disk automatically becomes the parity disk. The disk used as the parity disk before the disk

addition becomes a data disk. Figure 9-5 is an example illustrating how the disk configuration changes after you add a hot spare to the file system.

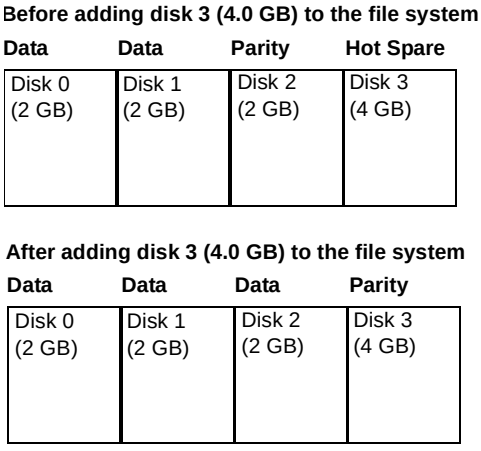


Figure 9-5. Effects of adding a disk to the file system

Because the hot spare disk (Disk 3) is larger than the parity disk when you add it to the system, the filer automatically uses it as the parity disk. Disk 2, which was previously the parity disk, becomes a data disk. The size of the file system increases by approximately 2 GB.

When installing a disk as a hot spare

A hot spare is a disk that automatically replaces a failed disk. However, a hot spare *cannot* replace a failed disk that's larger than itself. When you install a hot spare, be sure that it is at least as large as the largest file system disk. If you have multiple hot spares installed, the system uses the smallest hot spare first. For example, if one hot spare is 2-GB and another is 4-GB, the system automatically uses the 2-GB hot spare after a 2-GB file system disk has failed.

File system size limits and adding disks

When you add a disk to a disk shelf, you should be aware of the maximum file system size for your filer model. The size is checked when you use the `raid add` command to add a new disk. If adding the disk would cause the file system to exceed the maximum file system size by more than 0.5GB, the system rejects the request and writes the following message to the log:

```
Raid add not allowed; system capacity exceeded.
```

The disk you are trying to add remains a spare.

The maximum file system size for each filer is shown in Table 9-3.

Table 9-3. Maximum size of file system, by filer model

Filer models	Maximum
FAServer 450 and 1400, NetApp F200 series	54 GB
NetApp F300 series	117 GB
NetApp F500 series	243 GB
NetApp F600 series	450 GB

In the table, “GB” stands for 2^30 bytes.

Installing a new data disk

A new data disk is a disk that has never been used. To install it, follow these steps:

1. From the console, enter:

   ```
   toaster> raid swap
   ```

2. Insert the drive into the disk shelf according the instructions for your shelf.

3. Find the disk ID for the drive by entering:

   ```
   toaster> sysconfig -r
   ```

   The disk appears as a spare in the display.

4. Add the drive to the disk array by entering:

   ```
   toaster> raid add disk_id
   ```

The new data disk has now been added to the filer.

Installing multiple disks

You can follow the procedure in "Installing a new data disk" page 210 for each disk, or you can follow these steps:

1. Halt the filer with the `halt` command.
2. Install the drives into the disk shelf according to the instructions for your shelf.
3. Power up the filer.

If the disks were used by another filer or you get a "RAID tuple" error message, make sure that you erase the disk label before the filer boots with the disks installed. If the disk contains a disk label that was created by a previous filer, the filer will crash during the normal boot process. To erase the disk label, boot the filer from a boot diskette into maintenance mode, then use the `disk_erase_label` command as described in "Maintenance mode boot" on page 331.

Adding or removing hot spares

The filer supports zero or more hot spare disks. When a data disk on the filer fails, a hot spare automatically replaces it without a system shutdown.

To check whether your filer contains a hot spare, use the `sysconfig -r` command, which lists all the disks and indicates whether a disk is a parity, data, or spare disk. You can add more hot spare disks to the system at any time, as long as you have disk slots available.

The */etc/messages* file contains a log of any disk errors or failures along with any activity on the spare disk.

If your filer supports hot swapping

On filer models that support hot swapping, follow these steps to add or remove a hot spare:

1. If you are adding a hot spare, make sure that the hot spare is at least the same size as the largest disk currently in the RAID file system, then add the disk as described in "Adding a hot spare to the RAID file system" on page 213.

2. If you are removing a hot spare, hot swap the disk as described in "Removing a hot spare or an active file system disk" on page 214.

If your filer doesn't support hot swap

On filer models that don't support hot swapping, follow these steps to add or remove a hot spare:

1. If you are adding a hot spare, verify that the hot spare is at least the same size as the largest disk currently in the RAID file system.

2. Use the `halt` command to shut down the filer.

3. Turn off the filer.

4. Install or remove the hot spare according to instructions in the hardware guide for your filer model.



CAUTION: The system might hang if you do not install the disk properly.

5. Reboot the system. The system automatically recognizes the added disk as a hot spare.

Adding a hot spare to the RAID file system

To add to the RAID file system a disk that is currently installed as a hot spare, follow these steps:

1. Use the `sysconfig -r` command to determine the disk ID of the hot spare.

2. Use the `raid add` command to add the hot spare to the file system. For example, enter the following command to add a hot spare with disk ID 5:

```
raid add 5
```



NOTE: Make sure that any previous `raid` command has completed before you issue another `raid` command.

Hot swap

This section applies only to filer models that support hot swapping of disk drives. Hot swapping a drive means installing or removing a disk from the drive shelf when the filer is running, with minimal interruption to the file system.

Swapping a disk out of the filer

The procedure for hot swapping a disk out depends on the current status of the disk.



NOTE: A FAServer 450, FAServer 1300, or FAServer 1400 must have a disk at SCSI address 0.0 or 0.1 to boot, so don't leave both of these positions empty.

Removing a failed disk

If a disk has failed, enter the `raid swap` command as in the following example before removing the failed disk.

```
toaster> raid swap
System has been prepared for a disk swap. Network protocol
service will continue when a disk is removed or added to the
system.
Use the "raid unswap" command to continue service if this
command was accidentally typed or you choose not to swap a disk
at this time.
```

Then follow the instructions in the hardware guide for your filer model to remove the disk.

Removing a hot spare or an active file system disk

You might want to swap a disk out of a filer because you want to use it in another filer (if the disk is a hot spare) or because a disk is logging excessive errors.

If the disk is operational, follow these steps to swap out the disk:

1. If the disk is a hot spare, go to step a. If the disk is a file system disk, go to step b.

a. Enter the `raid remove` command to remove a hot spare. Use the `sysconfig -r` command to determine the disk ID number to be entered in the command. The `raid remove` command spins down the disk. After the disk stops spinning, the disk is ready to be removed. The following is an example of `raid remove`:

```
toaster> raid remove 1
Removal and unload of spare disk 1 has been initiated.
You will be notified when unload is complete.
toaster>
Wed Jun 01  22:10:12 GMT 1994 [raid_disk_admin]: Unload of disk
1 has completed successfully.
```

Go to step 2.

b. Enter the `raid fail` command to remove a file system disk. Use the `sysconfig -r` command to determine the disk ID number to be used in the command. The `raid fail` command spins down the disk. The following is an example of the `raid fail` command:

```
toaster> raid fail 1
Failure and unload of disk 1 has been initiated.
You will be notified via the system log when unload is complete
toaster> Wed Dec 14 23:19:46 GMT [raid_disk_admin]: Unload of
disk 1 has completed successfully.
Wed Dec 14 23:19:47 GMT [raid_stripe_owner]: Read on disk 1
failed, reverting to degraded mode.
```

After the `raid fail` command, the system operates in degraded mode, which means that a disk is missing from the RAID array. The system must reconstruct any files that use the missing disk by referring to the parity disk and the remaining members of the array. While a system in degraded mode can deliver file service, you should replace the disk as soon as possible to avoid losing data in case a second disk fails.

NOTE: You can use the `raid fail` command to remove only one disk at a time. If the system is already in degraded mode, do not execute the `raid fail` command to try to remove another disk.

2. Enter the `raid swap` command. The following is an example:

```
toaster> raid swap
System has been prepared for a disk swap. Network protocol
service will continue when a disk is removed or added to the
system.
Use the "raid unswap" command to continue service if this
command was accidentally typed or you choose not to swap a disk
at this time.
```

The `raid swap` command stalls all file activity until the next step is completed.

3. Remove the disk from the filer following the instructions in the hardware guide for your filer model. The filer then displays the following message:

```
toaster> Mon Dec 12 12:21:19 GMT [disk_config_admin]: ***
NOTICE ***
A disk has been swapped (removed or added) to a modular storage
shelf.  The system will wait 15 seconds and then check the
status of all disk drives.
```

File service resumes after the disk removal. The system waits 15 seconds and then displays the following message:

```
Mon Dec 12 12:21:35 GMT [disk_config_admin]: *** NOTICE ***
Disk unit status check has completed.
```

The hot swap procedure for removing a hot spare or file system disk is now complete.

If you want to replace the disk that has been swapped out, go to the next section. Be sure that the replacement disk is at least as large as the one removed.

Swapping a disk into the filer

You might want to hot swap a disk into a filer to replace a disk or to add a hot spare. Follow these steps for hot swapping a disk into the filer:

1. Enter the `raid swap` command. The following is an example:

```
toaster> raid swap
System has been prepared for a disk swap. Network protocol
service will continue when a disk is removed or added to the
system.
Use the "raid unswap" command to continue service if this
command was accidentally typed or you choose not to swap a disk
at this time.
```



NOTE: The `raid swap` command is required before you install a disk in the filer, even if you have already entered the command once in the preceding procedure for removing the replaced drive.

2. Install the disk according to the hardware guide for your filer model. The system then displays the following message:

```
toaster> Mon Dec 12 12:21:19 GMT [disk_config_admin]: ***
NOTICE ***
A disk has been swapped (removed or added) to a modular storage
shelf.  The system will wait 15 seconds and then check the
status of all disk drives.
```

The system waits 15 seconds as the disk is powered on, then initiates a disk spin-up. After the spin-up, the system recognizes the disk as a hot spare, and the following message appears:

```
Mon Dec 12 12:22:15 GMT [disk_config_admin]: *** NOTICE ***
Disk unit status check has found 1 new disk.
Mon Dec 12 12:22:15 GMT [raid_disk_admin]: Spare disk 1 has been
added to the system.
```

The hot swap procedure is now complete. If the filer is running in normal mode, the newly installed disk is used as a hot spare. If the system was in degraded mode prior to the swap, the filer automatically rebuilds the data of the replaced disk on the hot spare and displays a message similar to the following:

```
Wed Dec 14 23:20:02 GMT [raid_disk_admin]: One disk is missing
from the array
A spare disk is available and the missing disk will be
reconstructed on the spare disk.
```

Hot swapping a disk while dump is running

While dump is running, the filer resets all drives on the bus and rewinds all tapes in local tape drives. As a result, a hot swap could cause a dump to local tapes to fail. To prevent hot swap from interrupting a dump, the following message appears if you enter `raid swap` while the system is backing up files to any local tape:

```
** WARNING **
You can NOT swap a disk into or out of the system:
a dump is currently running on a local tape device and you must
wait until the dump is completed, or abort the dump before
reissuing the "raid swap" command.
```

You should re-enter the `raid swap` command after the dump is finished.

Disk errors during hot swap

If the filer detects a hung disk during the hot swap operation, it displays the following message and then halts:

```
*** WARNING ***
A disk has hung during a disk swap operation.  The system is
halting to allow recovery: please power off disk storage
shelves for 15 seconds to completely spindown the disks,
reapply power to shelves and then reboot the system.
```

A system reboot usually can clear transient disk errors that happen during the hot swap. Follow these steps to recover from those errors:

1. Disconnect the disk shelf used during the hot swap from the power source and wait 15 seconds. This allows all drives to spin down.
2. Reconnect the power source to the disk shelf.
3. At the system monitor prompt, type `b` to reboot the system.

After the reboot, your change to the disk array takes effect.

Contact Network Appliance Technical Support if you encounter any disk problems during a hot swap. When you call, be prepared to provide information about your system, such as the output from the `sysconfig -v` command, messages in */etc/messages*, and precisely how the hot swap was performed.

Moving a disk from one filer to another

You can install a disk that has been used by another filer. If your filer supports hot-swapping, installing the disk while the filer is running is the same as installing a new disk, as described in "Installing a new data disk" on page 210.

If you install the disk while the filer is not running, make sure that you erase the disk label before the filer boots with the disk installed. If the disk contains a disk label that was created by a previous filer, the filer will crash during the booting. For more information about how to erase the disk label, refer to "Maintenance mode boot" on page 331.

Increasing total disk capacity

This section describes how to increase total disk capacity by replacing all current disks with disks of higher capacity. This procedure involves creating a new file system and configuring the filer.



NOTE: You can increase disk capacity by adding disks if the filer's disk shelves are not full or by adding disk shelves.

The following is an example of increasing a filer's disk capacity. In this example, the file system is comprised of 2-GB disks, which are being replaced with 4-GB disks:

1. Perform a level 0 `dump` of the file system. The dump is required because during the procedure for increasing disk capacity, all data will be erased from the filer. For information about `dump`, see "Backing up the filer to tape" on page 273.
2. Follow the instructions regarding disk replacement in the hardware guide for your disk shelf.
3. Boot from floppy and install a new file system by following these steps:
   a. Insert the System Boot diskette into the disk drive.

b. At the filer prompt (for example, `toaster>`) on the console, type `reboot`. Booting from the floppy disk takes a few seconds longer than usual. Once the filer is booted, it displays the following menu:

```
1) Normal Boot
2) Boot without /etc/rc
3) Change Password
4) Install new file system
5) Maintenance mode boot

Selection (1-5)?
```

c. Type **4** and press Enter.

4. Answer the prompts as shown in the following example:

```
WARNING: there appear to be valid labels on the disks!

Zero disks and install a new file system? y
This will erase all the data on the disks, are you sure? y
Zeroing disks: takes approximately 66 minutes

A disk can be reserved and designated as a "hot spare"
which will be used to automatically replace another disk
should a disk fail in your configuration.  Note that the
spare disk will be reserved and will not be used for
your file system.

Reserve a disk for use as a "hot spare"? y
"Hot spare" disk 0 reserved.
```



NOTE: In this example, it takes about 66 minutes to reinstall the file system. The time required depends on the amount of disk space on your filer.

Information about system hardware configuration, such as the following, is displayed:

```
NetApp Release 4.1: Fri May 30 10:41:18 PST 1997
System ID: 0016777237
slot 0: System Board
        Memory Size:        64 MB
slot 0: SCSI Host Adapter 0
        6 Disks:            11.7GB
        In-Band Enclosure Services
slot 0: Ethernet Controller e0
        MAC Address:        00:a0:98:00:00:01 (Twisted pair)
slot 5: FDDI Network Adapter f5
        MAC Address:        08:00:2b:b2:39:cb (Twisted Pair)
slot 6: NVRAM
        Memory Size:        2 MB
slot 9: SCSI Host Adapter 9a
        In-Band Enclosure Services
6 Disks:            11.7GB
slot 9: SCSI Host Adapter 9b
        1 Tape:             EXB-85058SQANXR1
        In-Band Enclosure Services
```

5. Enter system configuration information. For more information about the information requested, see "General information" on page 12. The following is an example (the addresses are fictitious):

```
Please enter the new hostname: toaster
Please enter IP address for Network Interface
e0:266.266.266.263
Please enter netmask [255.255.255.0]:
Please enter media type for e0 {100tx-fd, 100tx, tp} [tp]:
Please enter IP address for Network Interface
e1:266.266.266.276
Please enter netmask [255.255.255.0]:
Please enter media type for e1 {100tx-fd, tp-fd, 100tx, tp}
[tp]: 100tx-fd
Please enter IP address for Network Interface fi0:
266.266.266.286
Please enter netmask [255.255.255.0]:
Please enter IP address for Network Interface
a5:266.266.266.296
Please enter netmask [255.255.0.0]: 255.255.255.0
Please enter name of the default gateway []: MyRouterBox
Please enter IP address for MyRouterBox: 266.266.266.261
The administration host is given root access to the
filer's /etc files for system administration. To allow
/etc root access to all NFS clients enter RETURN below.
Please enter name of administration host []: adminhost
Please enter IP address for adminhost: 266.266.276.264
```

After you finish entering the configuration information, the filer displays the following message:

```
Now unpack the tar file provided on the second floppy into the
root directory of the filer. When you are done, type “download”
to install the boot image on the hard disk. Then, set the
timezone and system clock using the “timezone” and “date”
commands.
add net default: gateway a1
savecore not supported
NFS server is running.
Server is in operation with the following export points:
/home   -root=adminhost
/       -access=adminhost, root=adminhost
Thu Jan  4 19:03:57 GMT [kbd_login_mgr]: Login from console
toaster>
```

6. Enter your license keys, necessary for activating filer protocols. For example:

   ```
   toaster> license nfs=ABCDEFG cifs=HIJKLMN http=OPQRSTU
   ```

7. Install the system files according to the upgrade procedure in the Read Me First for the Data ONTAP release used by your filer.

8. Use `telnet` to connect to the filer or go to the system console.

9. At the filer prompt, enter `download` to copy the executables to the filer’s boot blocks.

10. Remove the System Files diskette from the floppy drive of the client system.

11. Enter the `timezone` and `date` command at the filer prompt. The following are examples of these commands:

    ```
    toaster> timezone US/Pacific
    toaster> date 197003211234.56
    ```

12. Restore the complete file system. For information about restoring data, see “Restoring the entire file system” on page 291. If you use quota trees, be sure to create them before running `restore`.

13. Remove the System Boot diskette from the floppy drive of the filer and then reboot the filer by entering `reboot` at the filer prompt.



NOTE: Keep the replaced drives until the restore is completed.

The filer is now running with increased disk capacity.

Reducing the number of data disks

You might want to reduce the number of data disks on your filer if you have enough free space and want to use one disk as a hot spare. You can reduce the number of data disks by following these steps:

1. Perform a level 0 dump of the file system. The dump is required because during the procedure for reducing the number of disks, all data is erased from the filer. For information about dump, see "Backing up the filer to tape" on page 291.

2. Follow the instructions in the guide for your disk shelf to remove disks.

3. Reinstall the file system and restore your file system by following the procedure, beginning with step 3 on page 220, for increasing total disk capacity.

You are now operating with fewer data disks. If you want to use a disk you removed as a hot spare, follow the instructions in "Adding a hot spare to the RAID file system" on page 213.

Disk failures

The effects of a disk failure on filer operation depend on whether the filer is equipped with a hot spare.

Without a hot spare

If the filer is not equipped with a hot spare, after a disk has failed, the filer enters a state called "degraded mode." In this state, the RAID feature allows the filer to continue to run without losing data (although the filer's performance is affected). Replace the failed disk as soon as possible because a second disk failure would cause the entire file system to be lost.

When a disk fails, the filer logs one of the following warning messages in the */etc/messages* file and to the system console every hour:

```
Parity disk is broken. Halting in m hours.
Data disk n is broken. Halting in m hours.
```

(In the actual message, *n* is the disk ID number, and *m* is the number of hours before the system halts.)

Also, immediately before the system halts, a message similar to the following is sent to */etc/messages* and the console:

```
Sat Oct 29 13:26:42 PDT [statd]: When a disk is
  broken, the system shuts down automatically every 24
  24 hours to encourage you to replace the disk.
  If you reboot the system it will run for another
  24 hours before shutting down.
```

The shutdown ensures that you notice the disk failure. You can restart the filer without fixing the disk, but the filer will continue to shut itself off at the specified intervals until the problem is repaired.

By default, the filer shuts down after 24 hours. You can change this time interval using the `option raidtimeout` command; the argument is the time in hours that the system runs before automatic shutdown.

The system shuts down after the specified period if it is running in degraded mode. A filer is in degraded mode if:

- one disk in the file system has failed, or
- the batteries on the nvram card are low (if the filer is a PCI-based system).

With a hot spare

If a hot spare was reserved, the filer also enters degraded mode after a single disk failure. But in the meantime, in the background, the filer rebuilds the missing data on the hot spare in the background with minimal interruption to file service. The filer logs this activity in the */etc/messages* file and does not automatically shut down. If you power off the filer while it is in degraded mode, it stops data reconstruction. Once you power the filer back on, it restarts the data reconstruction process from the beginning.

Except for a slight loss in performance while data is rebuilt on the spare disk, the failure of a single disk is transparent to the user. The filer exits degraded mode and returns to normal operation once it has finished reconstructing data.

Network Appliance recommends that you replace the failed disk with a new hot spare once the filer has finished reconstructing data. This way, the filer continues to have a hot spare that can be used in case another disk fails.

The `sysconfig -r` command, described on page 202, displays which disk is reserved as the hot spare. In addition to disk failure and hot spare replacement activity, the */etc/messages* file logs any failure in a periodic check of the hot spare.

RAID reconstruction speed

You can control the speed of RAID data reconstruction by entering the following command:

```
options raid.reconstruct_speed speed
```

where *speed* is a number ranging from 1 (slowest) to 10 (fastest). Because RAID reconstruction consumes CPU time, sometimes increasing the speed of reconstruction slows down the filer's network operations. The default speed is 4, which means approximately 40% of the CPU time is used for RAID reconstruction.

When RAID reconstruction is in progress, use the `sysstat` command to check the system load on the filer. If the load is light, increase the speed of RAID reconstruction to maximize CPU utilization. For more information about the `sysstat` command, refer to "Displaying filer statistics" on page 313.

RAID scrubbing

RAID scrubbing is a procedure the filer uses to increase data availability. The filer scans each disk in the RAID group for media errors. If media errors are found, the filer tries to fix them by reconstructing the data from parity, and rewriting the data. raid scrubbing protects the system from a failure of two disks, because without scrubbing, RAID would not be able to recover from a disk media error encountered while the system was running in degraded mode.

The scrubbing process runs only while the filer is in normal mode. That is, the filer scans the disks only when all disks are operational. Although scrubbing slows the filer somewhat, network clients may not notice the change in the filer's performance because scrubbing starts automatically at 1:00 A.M. on Sunday when most systems are lightly loaded.

By default, RAID scrubbing is enabled. You may want to disable scrubbing if you have a recurring problem that scrubbing encounters. For example, there may be an unrecoverable error on a disk that you cannot fix before the next scrub. The following commands disable and enable scrubbing:

```
options raid.scrub.enable off
options raid.scrub.enable on
```

You can also manually start and stop scrubbing regardless of the current value ("on" or "off") of the `raid.scrub.enable` option. The following are the commands for starting and stopping scrubbing manually:

```
raid scrub start
raid scrub stop
```

Messages from the scrubbing process are sent to the system error logging daemon. The following are sample messages that may appear. In each message, the *n* represents a stripe or block number.

If the filer finds an inconsistent parity during scrubbing, it prints the following messages:

```
Inconsistent parity on stripe n
Rewriting bad parity block on stripe n
```

If the filer finds a media error on the parity disk, it prints the following message:

```
Rewriting bad parity block on stripe n
```

If the filer finds a media error on a data disk, it prints the following message:

```
Rewriting bad block from parity on disk n, block n
```

If the filer finds more than one bad block, it prints the following message:

```
Multiple bad blocks found on stripe n
```

The following are sample messages upon the completion of RAID scrubbing:

```
Scrub found n parity inconsistencies
Scrub found n media errors
Disk scrubbing finished...
```

Managing disk space

This section describes how to manage disk space on the filer by imposing disk quotas and by increasing the maximum number of files in the file system. This section also describes how to use the `df` command to monitor the current disk space usage.

Restricting disk usage by imposing disk quotas

Filer disk quotas restrict disk space and the number of files used by a given user or group. Quotas can also restrict disk space and the number of files used in a quota tree. A quota tree is a directory under the root directory (*/*) created by the `quota qtree` command.

Format of the /etc/quotas File

To set up disk quotas, create entries in the */etc/quotas* file. Here is a short sample */etc/quotas* file; the entries are explained later in this section:

```
#Quota Target          type            disk      files
/home/user/jdoe        user            500M      10K
21                     group           750M      75K
/proj1                 tree            750M      75K
writers                group@proj1     750M
*                      user            50M       10K
```



NOTE: Keep a record of your */etc/quotas* file in a safe place and update it as you change it in case you must do a restore without having access to your file system.

The Quota Target field. The Quota Target field contains the user, group, or tree that you are restricting. You can assign more than one quota to a user or group, but only one quota to a tree. The entries can be in any order.

You specify a user with one of:

- a file or subdirectory whose UID matches the user

- the user's user name as defined in the */etc/passwd* file or the NIS password map.
- the user's UID

The methods are equivalent, and inform the filer of the UID of the target. A file or directory is used only as the source of a UID; there are no special quota implications for that file or directory. The UID of the user must not be 0.

You specify a group with one of:

- a file or subdirectory whose GID matches the group
- the group's name
- the group's GID

The methods are equivalent. A file or directory is used only as the source of a GID; there are no special quota implications for that file or directory. The GID of the group must not be 0.

You specify a quota tree by entering its path name. Before you put the quotas into effect with the `quota on` command, use the `quota qtree` command to create any tree mentioned in the */etc/quotas* file.

Any file or subdirectory you use in the Quota Target field will be referenced repeatedly throughout the life of the system, so if you use a path name, choose a path name that will last for as long as the user or group account remains on the system. The user's or the group's home directory is a good choice for the path name.

You use an asterisk (*) to specify a default for the type that you want to restrict. Defaults apply to new users and groups that are created after the default entries have taken effect and to users and groups that are not explicitly mentioned in */etc/quotas*. You can override a default for a specific user or group by specifying a

quota for that user or group. You can assign a default only to the types “user” or “group”; you cannot assign a default to the type “tree.”

The type field. The type field specifies whether the Quota Target is a user, group, or tree. A user quota that applies just to a tree and not the entire file system is specified by user@*tree*. A group quota that applies just to a tree is specified by group@*tree*, where *tree* is the name of a tree. A quota tree is similar to a disk partition, but you can increase or decrease the size of a quota tree at any time.

The disk field. You use the disk field to specify the maximum amount of disk space that the quota target can use. In this field, k is equivalent to 1024 bytes, M means 220 bytes, and G means 230 bytes.

The files field. You use the files field to specify maximum number of files that the quota target can use. Use “K” to indicate 1024 files; 75K means 76,800 files. A blank indicates no restrictions on the number of files that the quota target can use.



NOTE: Do not put a blank in the disk field. It acts as white space. The filer would regard the following entries as equivalent:

```
#Quota Target          type           disk      files
/export                tree                     75K
/export                tree           75K
```

A detailed sample /etc/quotas file

Here is a more detailed sample */etc/quotas* file:

```
#Quota Target          type           disk      files
/home/user/jdoe        user           500M      10K
jsmith                 user           500M      10K
108                    user           500M      10K
publications           group          750M
21                     group          750M      75K
/home/group/services   group          750M      75K
/proj1                 tree           750M      75K
writers                group@proj1    150M
*                      user           50M       15K
*                      user@proj1     50M       10K
*                      group          750M      85K
*                      group@proj1    100M      75K
mhoward                user           150M      100K
mhoward                user@proj1     75M       75K
```

Any operation that creates files or writes to them must satisfy all applicable quotas. The detailed sample */etc/quotas* file specifies that:

- The owner of */home/user/jdoe*, the user whose login is "jsmith," and the user whose UID is 108 can each use 500 MB of disk space and 10,240 files.
- The group "publications" can use 750M of disk space with no quota restrictions on the number of files.
- The group whose GID is 21 and the group that owns */home/group/services* can each use 750 MB of disk space and 76,800 files.
- The quota tree "proj1" can use 750 MB of disk space and 76,800 files.
- The group "writers" can use 150 MB of disk space and an unlimited number of files in "proj1" provided that the quotas on the tree "proj1" are not exceeded.

- Any user not otherwise mentioned in this file can use 50 MB of disk space and 15,360 files. In the tree "proj1," the user can use 50 MB of disk space and 10,240 files provided that the quotas on the tree "proj1" are not exceeded and provided that the user does not use more than 50 MB of disk space and 15,360 files in the file system.

- Any group not otherwise mentioned in this file can use 750 MB of disk space and 87,040 files. In the tree "proj1," the group can use 100M of disk space and 76,800 files, provided that the group does not use more than 150 MB of disk space and 102,400 files in the file system.

- The user "mhoward" can use 150 MB of disk space and 102,400 files. In the tree "proj1," "mhoward" can use 75 MB of disk space and 76,800 files provided that the quotas on the tree "proj1" are not exceeded and provided that "mhoward" does not use more than 150 MB of disk space and 102,400 files in the file system.

- Both user and group quotas apply to the entire filer, even if the quota target in the */etc/quotas* file is specified in the form of a path name. For example, if the quota target for the user named "jdoe" is */home/user/jdoe*, the filer imposes the quotas on *all* files written by jdoe, not just the ones written to */home/user/jdoe*.

Some of the information from `quota` commands is available through snmp using the Network Appliance custom MIB. For more information about MIB, refer to "Using the Network Appliance Custom MIB" on page 109.

The quota command

The command syntax for enabling, disabling, or resizing quotas is as follows:

```
quota on | off | resize
```

Quotas go into effect when a `quota on` command is entered. The `quota on` command computes the disk usage of each quota target, so this can take a few minutes to complete for a large

number of quotas. Because the filer remembers whether quotas are on or off even after it reboots, there is no need to add a `quota` command to */etc/rc*.

The `quota resize` command updates active quotas without recalculating disk usage, and is faster than `quota off` followed by `quota on`. An active quota is one that appears in the quota report displayed by the `quota report` command, discussed in the next section, "Displaying information about quotas," on page 238. You use the `quota resize` command after changing limits for a group or user. For example, jdoe has a disk quota of 500M as shown below:

```
#Quota Target          type            disk      files
/home/user/jdoe        user            500M      10K
*                      user            10M       10K
```

If jdoe's disk quota were increased as shown below:

```
#Quota Target          type            disk      files
/home/user/jdoe        user            600M      10K
*                      user            10M       10K
```

you would use the `quota resize` command to update the quota.

The `quota resize` command usually ignores newly added quota targets. For each entry that it skips, the `quota resize` command prints the message:

`quota resize: new entry on line` *n* `in /etc/quotas`

where *n* is a line number.

If a default quota applies to the creator of a file, an active quota record is created for the owner of the file. The `quota resize` command does not skip newly added quota targets if the corresponding user or group has written to a file that is under the control of a default quota.

You can resize quotas after making changes to the */etc/quotas* file by executing the `quota off` command followed by the `quota on` command. This makes the quotas in the */etc/quotas* file active.

Active quotas appear in the report created by the `quota report` command.

A quota also becomes active when a quota target has written a file. You can make a new quota active by writing a file and giving ownership of the file to the quota target, then running the `quota resize` command. This takes less time than executing the `quota off` command followed by the `quota on` command because only the newly active quotas are recalculated.

Displaying information about quotas

The `quota report` command displays the current consumption of files and space for each user or group with a quota. With a *path* argument, `quota report` displays information about all quotas that apply to the specified file.

The `quota qtree` command, with no additional argument, lists the quota trees on the filer. This differs from the `quota report` command in that it lists all quota trees, including the ones that are not listed in */etc/quotas*. Any directory that was created by the `quota qtree` command is a quota tree, whether or not a quota is currently imposed on it.

Creating user quotas

Follow these steps to limit disk space used by a user named "jdoe":

1. Decide who should be limited by a disk quota. In this example, the user is the owner of */home/user/jdoe*. The filer will restrict disk usage of files with the same UID as */home/user/jdoe*.

2. Add the following line to the */etc/quotas* file:

```
/home/user/jdoe  user  300M  20K
```

substituting the values you want for “300M” and “20K.” You can also use the UID of jdoe or the name jdoe if you have set up a *passwd* file or map.

3. Enter the `quota on` command. If quotas are already on, enter the following:

```
quota off
quota on
```

Creating group quotas

Follow these steps to limit disk space used by a group named “service”:

1. Decide which group should be limited by a disk quota. In this example, the group is the owner of */home/user/service*. That is, the filer restricts disk usage of files with the same GID as */home/group/service*.

2. Add the following line to the */etc/quotas* file:

```
/home/group/service  group  700M  100K
```

   substituting the values you want for “700M” and “100K.” You can also use the GID of the group or the name of the group.

3. Enter the `quota on` command. If quotas are already on, enter the following:

```
quota off
quota on
```

Creating a quota tree

Quota trees limit the amount of space consumed by the user or groups that are subject to it, regardless of the user or group ownership of the tree's contents. Follow these steps to implement a quota tree:

1. Create a quota tree using the `quota qtree` command. For example:

   ```
   quota qtree news
   ```

   You can create a quota tree only in the filer's / directory. This example creates the */news* quota tree.



NOTE: You cannot impose tree quotas on directories that were created using the `mkdir` command. You can impose tree quotas only on directories that were created using the `quota qtree` command.

2. Add the following line to the */etc/quotas* file:

   ```
   /news  tree  400M  50K
   ```

3. Add other quotas as desired. For example, to limit jdoe to 10 megabytes of disk space and 10240 files in the */news* quota tree, add the line:

   jdoe user@news 10M 10K

4. Enter the `quota on` command. If quotas are on already, enter the following:

   ```
   quota off
   quota on
   ```



CAUTION: You cannot move a file into or out of a quota tree with a rename operation. As a result, the `mv` command on some UNIX systems will fail if you try to move a file into or out of a quota tree. You can always move the file by copying it, then deleting the original.

Removing quota restrictions

To remove quota restrictions, follow these steps:

1. Remove the appropriate line or lines from the */etc/quotas* file.
2. Enter the following commands:

```
quota off
quota on
```

When quotas are exceeded

When it receives a write request, the filer first checks the quota tree the file to be written is in a quota tree. If the write would exceed the tree quota tree, the filer returns an "out of disk space" error to the write request and logs the following error message:

`tid` *`tree_id`*`: tree quota exceeded`

If the quota tree is not full but the write would cause either the user or group quota to be exceeded, the filer returns a "disk quota exceeded" error to the write request and logs one of the following errors:

`uid` *`user_id`*`: disk quota exceeded`
`gid` *`group_id`*`: disk quota exceeded`

where

user_id is the user's UID.

group_id is the group's GID.

If you are about to exceed a user or group quota, the filer returns a "disk quota exceeded" error to the write request and logs one of the following errors:

`uid` *`user_id`* `tid` *`tree_id`*`: disk quota exceeded`
`gid` *`group_id`* `tid` *`tree_id`*`: disk quota exceeded`

where

user_id is the user's UID.

group_id is the group's GID.

tree_id is the tree ID.

Increasing the maximum number of files

Initially, the maximum number of files on the filer is set at one for every 32 KB of disk space. The number is increased automatically when you add a new disk. Unlike UNIX, which requires that you specify the maximum number of files in a file system when you create the file system, the filer allows you to use the `maxfiles` command to increase the number of files at any time. The only argument to the `maxfiles` command is the new maximum number of files.



NOTE: Use caution when increasing the maximum number of files because once you increase this number, you can never reduce it. As new files are created, the file system consumes the additional disk space required to hold the inodes for the additional files; there is no way for the filer to release that disk space. An inode is a data structure containing information about files.

To determine how many files are in the file system, use the `df -i` command to see how many inodes have been used, or use the `maxfiles` command without an argument.

Information generated by the `maxfiles` command is also available through snmp using the Network Appliance custom MIB. For more information, see "Using the Network Appliance Custom MIB" on page 109.

The df command

To verify the amount of free disk space on the filer, execute the `df` command on the filer. Information generated from the `df` command is also available through snmp, using the Network Appliance custom MIB, which is described in e "Using the Network Appliance Custom MIB" on page 109.

With the `-i` option, the command displays the number of used inodes and the number of available inodes. The following is an example of the `df -i` command:

```
toaster> df -i
Filesystem    iused      ifree    %iused   Mounted on
/           240843     121525     66%     /
```

The total amount of disk space shown in the `df` output is less than the sum of available space on all disks installed in the RAID array. Consider the following sample `df` command issued on a system with seven 2-GB disks installed:

```
toaster> df
Filesystem   kbytes       used    avail  capacity  Mounted on
/            9886448   8426196  1460252     85%    /
/.snapshot   1098492    427008   671484     39%    /.snapshot
```

When you add up the numbers in the “kbytes” column, you’ll notice that the sum, 10,984,940 KB (about 11 GB), is significantly less than the total amount of disk space installed (14 GB). The following are the reasons for the discrepancy:

- The parity disk, which is a 2-GB disk in this example, is not reflected in the output of the `df` command.
- As with the UNIX FFS (Fast File System), the filer reserves 10 percent of the total disk space for efficiency, which `df` does not count as part of the file system space.

The filer doesn’t limit the size of a file system. The maximum size of the file system on the filer is restricted only by the amount of disk space installed. For individual files, the maximum size is 4 GB minus one byte. (Many NFS clients restrict the file size to 2 GB.)

Using the df command with quota trees

When you enter a `df` command with a path name on a client, the command returns the amount of free space in the file system containing the path name. For example:

```
client% df /n/toaster/engineering/jdoe
Filesystem           kbytes    used   avail capacity  Mounted on
toaster:/           2097151 1646923  450228    79%    /n/toaster
```

However, if you have defined quota trees on the filer, the information about available space could be misleading because the actual space available could be less. For example, if */engineering* is a quota tree with a maximum disk space of 1800 mb, the actual space available to the */n/toaster/engineering* directory will be less than that in the `df` command output shown above.

If you have quota trees on the filer, Network Appliance recommends that you mount each quota tree separately. For example, if the filer named toaster has two quota trees, */engineering* and */marketing*, mount *toaster:/engineering* and *toaster:/marketing* on two mount points (for example, */n/toaster/engineering* and */n/toaster/marketing*). In this way, the filer takes the quota trees into consideration when responding to a `df` command from a client and returns the amount of free space in each quota tree as opposed to the space available in the entire file system.

Chapter 10: **Snapshots**

A snapshot is a read-only copy of the entire file system—it reflects the state of the file system at the time the snapshot was created. Any client of a filer can access snapshots to recover old versions of files (for example, files that have been accidentally changed or deleted). The snapshot feature allows users to restore their own files without system administrator intervention, because files in snapshots can be viewed and copied by those who have permission to do so with the original files.

Snapshots also simplify tape backup. The filer `dump` command automatically creates a snapshot of the active file system, if necessary, before backing up the data to tape (it is not necessary if you are backing up an existing snapshot). Because a snapshot is a read-only copy of the file system, it does not change even when files in the active file system are changing. As a result, `dump` can make a safe and consistent backup without requiring the system to be taken off line.

The filer uses a copy-on-write technique to create snapshots quickly without consuming any disk space. Snapshots begin to consume extra space only as blocks in the active file system are modified and written to new locations on disk. For more information on the copy-on-write technique used by snapshots, refer to "How snapshots work" on page 252.

The filer creates and deletes snapshots automatically at preset intervals. You can also create and delete snapshots manually. The filer can have up to 20 different snapshots at one time.

Snapshot files carry the same permissions and inode numbers as the original files, keeping the integrity of the security system intact. An inode is a data structure holding information about files in a UNIX file system. There is an inode for each file and a file is uniquely identified by the file system on which it resides and its inode number on that system.



NOTE: On UNIX systems, the inode number for a file in a snapshot is also the inode number for the corresponding file in the active file system. This can confuse some programs, such as the GNU `diff` program, that think that the snapshot and the original are the same file. If you copy one of these files to another file, you will have files with different inode numbers.

Snapshot commands

The commands related to snapshots are listed in Table 10-1.

Table 10-1. Commands for snapshots

Command	Meaning
`snap list` [a]	Lists all available snapshots.
`snap create` *`name`*	Creates a snapshot with a specified name.
`snap delete` *`name`*	Deletes a specified snapshot.
`snap rename` *`from to`*	Renames a snapshot.
`snap reserve`	Reserves a percentage of the disk space for snapshots.
`snap sched`	Schedules automatic snapshots.

a. Some of the information generated by this command is available through SNMP using the Network Appliance custom MIB. The custom MIB is described in Chapter 4, "Routine filer administration."

Automatic snapshot creation

The filer uses the snap sched command to create snapshots automatically and to keep them on-line for a predetermined amount of time. There are three types of automatic snapshots: hourly, nightly (at midnight), and weekly (every Sunday night at midnight).

There are three types of snapshots:

- Weekly - the filer creates these every Sunday night at midnight. Weekly snapshots are called *weekly.n*, where *n* is an integer. *weekly.0* is the most recent weekly snapshot, and *weekly.1* is the next most recent weekly snapshot. When a new weekly snapshot is created, the value of *n* is adjusted for all weekly snapshots. The higher the value of *n*, the older the snapshot.

- Nightly - the filer creates these every midnight except when a weekly snapshot is scheduled to occur at the same time; if the number of weekly snapshots is nonzero, and it's the day of the week that weekly snapshots occur, no nightly snapshot is created. Nightly snapshots are called *nightly.n*, where *n* is an integer. *nightly.0* is the most recent nightly snapshot, and *nightly.1* is the next most recent nightly snapshot. When a new nightly snapshot is created, the value of *n* is adjusted for all nightly snapshots. The higher the value of *n*, the older the snapshot.

- Hourly - the filer creates these on the hour at specified hours, except at midnight, if a nightly or weekly snapshot is scheduled to occur at the same time; this occurs either if the number of nightly snapshots in the schedule is nonzero, or if the number of weekly snapshots in the schedule is nonzero, and it's the day of the week that weekly snapshots occur. Hourly snapshots are called *hourly.n*, where *n* is an integer. *hourly.0* is the most recent hourly snapshot, and *hourly.1* is the next most recent hourly snapshot. When a new hourly snapshot is created, the value of *n* is adjusted for all hourly snapshots. The higher the value of *n*, the older the snapshot.

Figure 10-1 shows a sample snap sched command.

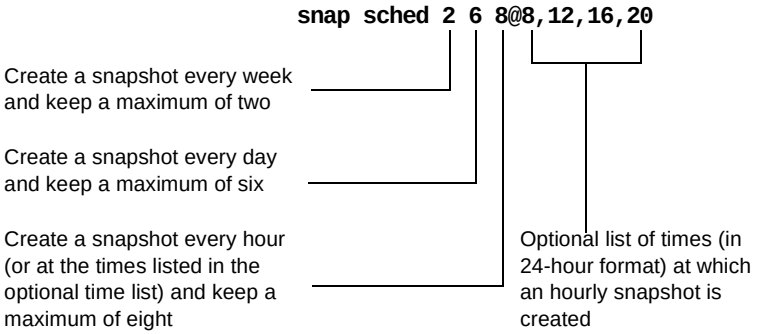


Figure 10-1. An example of the snap sched command

The first argument of the snapshot command indicates how many weekly snapshots to keep, the second how many nightly snapshots to keep, and the third how many hourly snapshots to keep. A zero in any of the three positions disables snapshots for that interval. The argument for hourly snapshots can include an optional list of numbers indicating the hours at which the filer creates the snapshots in 24-hour time. If the argument is omitted, the file creates an hourly snapshot every hour.

The default snapshot schedule is:

```
snap sched 0 2 6@8,12,16,20
```

The following snapshot schedule:

```
snap sched 2 6 8@8,12,16,20
```

creates these snapshots:

- weekly snapshots, and keeps the two most recent
- daily snapshots, and keeps the six most recent

- hourly snapshots at 8:00 A.M, noon, 4:00 P.M., and 8:0 0 P.M., and keeps the eight most recent

The list below shows the snapshots that would be created by this snapshot schedule in 1998 (when January 11 is a Sunday).

```
% ls -lu .snapshot
total 64
drwxrwsrwx  2 root 4096 Jan 14 12:00 hourly.0
drwxrwsrwx  2 root 4096 Jan 14 08:00 hourly.1
drwxrwsrwx  2 root 4096 Jan 13 20:00 hourly.2
drwxrwsrwx  2 root 4096 Jan 13 16:00 hourly.3
drwxrwsrwx  2 root 4096 Jan 13 12:00 hourly.4
drwxrwsrwx  2 root 4096 Jan 13 08:00 hourly.5
drwxrwsrwx  2 root 4096 Jan 12 20:00 hourly.6
drwxrwsrwx  2 root 4096 Jan 12 16:00 hourly.7
drwxrwsrwx  2 root 4096 Jan 14 00:00 nightly.0
drwxrwsrwx  2 root 4096 Jan 13 00:00 nightly.1
drwxrwsrwx  2 root 4096 Jan 12 00:00 nightly.2
drwxrwsrwx  2 root 4096 Jan 10 00:00 nightly.3
drwxrwsrwx  2 root 4096 Jan 09 00:00 nightly.4
drwxrwsrwx  2 root 4096 Jan 08 00:00 nightly.5
drwxrwsrwx  2 root 4096 Jan 11 00:00 weekly.0
drwxrwsrwx  2 root 4096 Jan 04 00:00 weekly.1
```

This schedule keeps the eight most recent hourly snapshots (created at 8 A.M., 12 noon, 4 P.M.., and 8 P.M.. every day), the six most recent daily snapshots, and the two most recent weekly snapshots. Whenever the filer creates a new snapshot of a particular type, it deletes the oldest one and renames the existing ones. On the hour, for example, the filer deletes *hourly.7*, renames *hourly.0* to *hourly.1*, and so on. Notice that the nightly snapshot schedule jumps from January 12 to January 10 because there is a weekly snapshot on January 11.



NOTE: On a UNIX client, if you use `ls -l` instead of `ls -lu` to list the snapshot creation times, the times will not necessarily all be different. The times listed by `ls -l` reflect the modification times of the directory at the time of each snapshot, and are not related to the times at which the snapshots are created.

Changes resulting from a `snap sched` command are effective until the next system reboot. To change the schedule permanently, edit the */etc/rc* file as described in Chapter 4, "Routine filer administration."

You can also create snapshots at predefined times instead of using the hourly, daily, and weekly schedules. For example, if you want to create two snapshots each week, you can set up a `cron` job on the administration host to run twice each week at an appropriate time to execute this `snap create` command:

```
rsh toaster snap create filename
```



NOTE: Do not include a slash (/) in the name of a snapshot in the `snap create` command because the snapshot will not be accessible to NFS clients (unless you rename the snapshot file using the `snap rename` command to a name with no slash in it).

How snapshots work

When the filer creates a snapshot, it doesn't actually copy disk blocks; instead, it identifies all blocks in the file system as belonging to the snapshot as well as to the active file system. Consider a particular file named *foo* in a newly created snapshot. Because the snapshot was just created, the snapshot version of *foo* has exactly the same contents as the version in the active file system. The same blocks on disk store both versions, so the snapshot version of *foo* consumes no disk space.

Later, if *foo* is deleted, the blocks holding the data for *foo* are no longer part of the active file system, but they are still part of the snapshot. Deleting *foo* from the active file system, then, does not free up any disk space. Figure 10-2 illustrates how disk space is used before and after foo is removed.

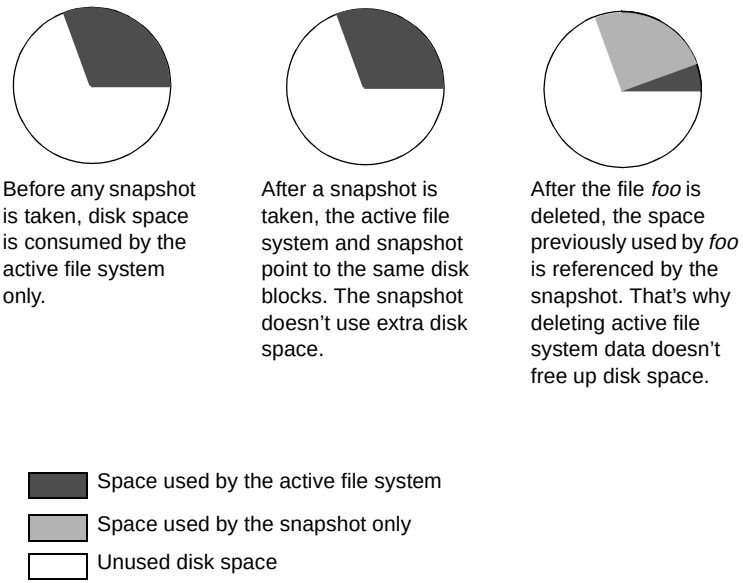


Figure 10-2. How snapshots use disk space

Changing the contents of *foo* creates a similar situation. New data written to *foo* cannot be stored in the same disk blocks as the current contents because the snapshot is using those blocks to store the old version of *foo*. Instead, the new data is written to new disk blocks, so there are two separate copies of *foo* on disk—a new copy in the active file system and an old one in the snapshot. This technique of duplicating disk blocks only as they are modified is called *copy-on-write*.

In summary, when the filer creates a snapshot, it doesn't use any disk space at all, but as files in the active file system are changed or deleted, the snapshot uses more and more space. How often files are changed and deleted determines the number of snapshots the filer can create and the length of time the snapshots can be kept.

In many environments, most data in the file system remains unchanged from day to day. For instance, a user with a 10-MB home directory might change only 100 KB to 500 KB on a typical day. When files change slowly, snapshots can be kept on-line for days or even weeks before they begin to consume unacceptable amounts of disk space.

In other environments, data in the file system changes quickly. If a large percentage of the file system changes every day, there might not be room to keep snapshots for even a few hours.

Understanding snapshot disk consumption

A confusing aspect of snapshots is the amount of disk space they currently consume and the amount of disk space they are likely to consume. The following sections explain with examples how to determine the amount of disk space used by snapshots.

Disk consumption by multiple identical snapshots

Suppose a snapshot contains a 1-mb file that hasn't changed since the filer created the snapshot. If that file is removed from the active file system, the snapshot then consumes 1 mb of disk space. The same version of that 1-mb file may be referenced by several different snapshots: *hourly.0*, *hourly.1*, and *hourly.2*. If these snapshots all contain the 1-mb file that hasn't changed since the filer created those snapshots, only 1 mb of disk space is consumed by the snapshots even though all three snapshots contain the file.

How df displays snapshots

To provide information about snapshot disk utilization, the `df` command on the filer treats snapshots as a partition different from the active file system. The following is an example of the `df` command:

```
toaster> df
Filesystem            kbytes    used   avail capacity
/                    3000000 2000000 1000000      65%
/.snapshot           1000000  500000  500000      50%
```



NOTE: The numbers in this example have been rounded off to make the example easier to understand.

The filer has 4 gb of disk space with 1 gb (or 25%) reserved for snapshots (the idea of reserving space for snapshots is described in more detail later). That leaves 3 gb for the active file system, and 2 gb of the file system is in use. It is important to understand that the `/.snapshot` line counts data that exists only in a snapshot. Because data that also exists in the active file system

needs to be stored on disk anyway, it would be misleading if the filer charged the space to snapshots. In the example, half of the 1 gb reserved for the snapshot is actually used.

How the snapshot reserve works

Given the way snapshots consume disk space, why reserve space for them? Why not just let them use whatever space they need? To understand the answer, consider what would happen if the filer created a snapshot when the disks were full. Removing files from the active file system wouldn't create any free space, because everything in the active file system would also be referenced by the newly created snapshot. The filer would have to delete the snapshot before it could create any new files.

By default, the snapshot reserve is 20% of disk space. For information about how to adjust the amount of the snapshot reserve, refer to "Changing the snapshot reserve" on page 262.

Consider what happens if the filer creates a snapshot when the active file system is full and there is still space remaining in the snapshot reserve. The `df` output might show:

```
Filesystem            kbytes    used   avail capacity
/                    3000000 3000000       0     100%
/.snapshot           1000000  500000  500000      50%
```

If you remove 100 mb of files, the data belonging to these files will no longer be part of the active file system, so the space will be charged to the snapshots instead. The new `df` output is as follows:

```
Filesystem            kbytes    used   avail capacity
/                    3000000 2900000  100000      97%
/.snapshot           1000000  600000  400000      60%
```

Notice that the filer reassigns 100 mb of space from the active file system to the snapshot reserve. Because there was reserve space for snapshots, removing files from the active file system freed up space for new files.

Even with the snapshot reserve, the job of administering snapshot disk space consumption is important. There is no way to prevent snapshots from consuming disk space greater than the amount reserved for them. Consider what would happen in our example file system if all files in the active file system were deleted. Before the deletion, the `df` output was:

```
Filesystem            kbytes    used   avail capacity
/                    3000000 3000000       0     100%
/.snapshot           1000000  500000  500000      50%
```

After removing all of the data in the file system, the `df` command generates the following:

```
Filesystem            kbytes    used   avail capacity
/                    3000000 2500000  500000      83%
/.snapshot           1000000 3500000       0     350%
```

The entire 3 gb in the active file system moved into snapshots, along with the 500 mb that were in snapshots before, making a total of 3.5 gb of snapshot data. This is 2.5 gb more than the space reserved for snapshots. Because the active file system cannot use that space, the space shows up as used by the active file system even though no files are actually there.

Whenever snapshots consume more than 100% of the snapshot reserve, the system is in danger of becoming full. In this case, you can create files only after you remove enough snapshots. For instance, if 500 mb of data were added to the active file system in the example above, a `df` command would generate the following information:

```
Filesystem            kbytes    used   avail capacity
/                    3000000 3000000       0     100%
/.snapshot           1000000 3500000       0     350%
```

As soon as the filer creates a new snapshot, every block in the system will be referenced by some snapshot, so no matter how many files you remove from the active file system, there will still not be room to add any more. The only way to recover from this

situation is to remove enough snapshots to free more disk space. Refer to "Displaying snapshot statistics" on page 259 for information about how to use the `snap list` command to determine which snapshot to delete.

Managing snapshot disk consumption

This section describes how to schedule snapshots to suit your environment, how to determine a reasonable snapshot reserve, and how to adjust the amount of space snapshots use.

Scheduling snapshots

The best way to manage the amount of space consumed by snapshots is to use `snap sched` to adjust the schedule of snapshot creation. The following are some suggested strategies for scheduling and retaining snapshots:

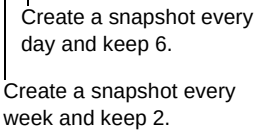
- If users rarely lose files or typically notice lost files right away, use the default snapshot schedule, which keeps snapshots for only a couple of days:

```
snap sched 0 2 6@8,12,16,20
```

Create a snapshot every day and keep 2.

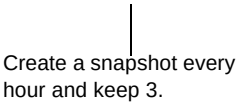
- If users commonly lose files and need to restore them, Network Appliance recommends that you delete the snapshots less often than you would in the preceding example. On many systems only 5% or 10% of the data changes each week, so the snapshot schedule of six nightly and two weekly snapshots consumes 10% to 20% of disk space. Considering the benefits of snapshots, it is worthwhile to reserve this amount of disk space for snapshots. The following is the recommended snapshot schedule:

```
snap sched 2 6 8@8,12,16,20
```



- For file systems that change very quickly, reduce the number of snapshots scheduled. For example, if the entire file system on a filer is filled and emptied each day (for example, a file system storing large temporary files for a cad application), it might not make sense to use daily or weekly snapshots at all. On a very active system, Network Appliance recommends that you schedule snapshots every hour and keep them for just a few hours. For example:

```
snap sched 0 0 3
```



  This schedule doesn't consume much disk space, and it lets users recover files in recent snapshots as long as they notice their mistake within a couple of hours.

Displaying snapshot statistics

The `snap list` command includes statistics about the disk space consumption of each snapshot in the system. This command makes it possible to see how much space each snapshot is using, and it can also help determine an appropriate snapshot reserve. Here is an example of output from `snap list`:

```
toaster> snap list
%/used       %/total    date          name
----------  ----------  ------------  --------
0% ( 0%)    0% ( 0%)    Jan 19 08:01  hourly.0
1% ( 1%)    1% ( 1%)    Jan 19 00:01  nightly.0
2% ( 2%)    2% ( 2%)    Jan 18 20:01  hourly.1
3% ( 2%)    2% ( 2%)    Jan 18 16:01  hourly.2
3% ( 2%)    3% ( 2%)    Jan 18 12:01  hourly.3
5% ( 3%)    4% ( 3%)    Jan 18 00:01  nightly.1
7% ( 4%)    6% ( 4%)    Jan 17 00:00  nightly.2
8% ( 4%)    7% ( 4%)    Jan 16 00:01  nightly.3
10%( 5%)    9% ( 4%)    Jan 15 00:01  nightly.4
```

- The "%/used" column shows space consumed by snapshots as a percentage of disk space being used. The first number is cumulative, for all snapshots listed so far, and the second is for the specified snapshot alone.

- The "%/total" column shows space consumed as a percentage of total disk space in the system.

The "%/used" is more useful for planning the snapshot reserve because it is more likely to remain constant as the file system fills.

The following explains in detail the percentages in the `snap list` output:

- The "%/used" column:

  The first value is equal to:

$$\frac{\text{cumulative snapshot space} \times 100\%}{\text{cumulative snapshot space} + \text{file system space}}$$

The second value is equal to:

$$\frac{\text{this snapshot} \times 100\%}{\text{this snapshot + file system space}}$$

- The "% / total" column:

  The first value is equal to::

  $$\frac{\text{cumulative snapshot space} \times 100\%}{\text{total disk space on the system}}$$

  The second value is equal to:

  $$\frac{\text{this snapshot} \times 100\%}{\text{total disk space on the system}}$$

"Cumulative snapshot space" is the total space used by this snapshot and all other more recent snapshots (the ones listed above this snapshot in the `snap list` output).

This example shows a system that keeps five nightly snapshots and four hourly ones (that is, the system uses `snap sched 0 5 4@8,12,16,20`). The `snap list` output shows that the overhead for snapshots is only 10%, so the default snapshot reserve of 20% seems to be a waste of disk space in this case. Assuming that this pattern of change holds up, a reserve of 12% to 15% should provide a safe margin to ensure that removing files will free up space when the active file system is full.

The values in parentheses (the values that show the space used by an individual snapshot) are useful in identifying a particular snapshot to remove when the file system is full. However, deleting a particular snapshot doesn't necessarily release the total amount of space indicated, because other snapshots may be referring to the same blocks. Refer to "Freeing up disk space by deleting a snapshot" on page 262 for further information on how to select a snapshot file for deletion if you want to reclaim disk space.

If you do not want the total amount of space consumed by all snapshots to exceed a certain percentage of the used disk space, use the cumulative values in the `snap list` output to determine which snapshots to delete. In the example above, if you don't want more than 5% of used disk space to be spent by snapshots, delete all snapshots listed below *nightly.1* in the `snap list` output (that is, *nightly.2*, *nightly.3*, and *nightly.4*). After deleting the snapshots, *nightly.1* and all the other more recent snapshots consume 5% of the used disk space.

Changing the snapshot reserve

The snapshot reserve can be used only by snapshots, not by the active file system.

The default snapshot reserve is 20% of the available disk space. To change the reserve, enter a command in the following format:

`snap reserve` *`percent`*

For example:

```
snap reserve 25
```

With no arguments, the `snap reserve` command displays the percentage of disk space reserved for snapshots.



NOTE: Snapshots can exceed the snapshot reserve space.

Freeing up disk space by deleting a snapshot

This section describes how to use the `snap list` output to determine which snapshot file to delete to free up the most disk space.

In the sample `snap list` output in "Displaying snapshot statistics" on page 259, the cumulative disk space used by snapshots gradually increases from top to bottom. For example, in the "%/used" column, the cumulative space used by *hourly.1* is 2% and the cumulative space used by *hourly.2* is 3%. This is not always the case. Consider a filer with a 100-mb file system that has not changed since the first snapshot was taken. The `snap list` command on this filer displays the following:

```
%/used      %/total     date          name
--------    ---------   ------------  ---------
0% (0%)     0% (0%)     May 05 16:00  hourly.0
0% (0%)     0% (0%)     May 05 12:00  hourly.1
0% (0%)     0% (0%)     May 05 08:00  hourly.2
```

The cumulative space used by snapshots does not increase because no changes have been made to the file system. But if you had deleted 20 mb from the file system before the filer took the *hourly.0* snapshot, the `snap list` command would have displayed the following:

```
%/used       %/total      date          name
--------     ---------    ------------  ---------
0%  (0%)      0% (0%)     May 05 16:00  hourly.0
20% (20%)     1% (1%)     May 05 12:00  hourly.1
20% (20%)     1% (1%)     May 05 08:00  hourly.2
```

Notice that in the "%/used" column, the cumulative values for *hourly.1* and *hourly.2* are both 20%, but the cumulative value for *hourly.2* is not 40%. This is because both snapshots point to the same 20 mb of data (the data that you just deleted).

The cumulative values for *hourly.1* and *hourly.2* would be different if you deleted and created data between snapshots in this way:

1. Deleted 20 mb of data and created 20 mb of new data after *hourly.2*.
2. Deleted the 20 mb of data created in step 1 after *hourly.1*.

After the data deletions and additions, the `snap list` command would display the following:

```
%/used       %/total      date          name
--------     ---------    ------------  ---------
0%  (0%)      0% (0%)     May 05 16:00  hourly.0
20% (20%)     1% (1%)     May 05 12:00  hourly.1
33% (20%)     2% (1%)     May 05 08:00  hourly.2
```

In this scenario, *hourly.1* and *hourly.2* each consume 20% of the used disk space (20 mb out of 100 mb). But this time they reference different data blocks. Cumulatively, they consume a total of 40 mb, which is about 33% of the disk space used (120 mb, which is 40 mb used by snapshots plus 80 mb in the file system).

Before trying to conserve space by deleting a large snapshot file, examine the cumulative values in the `snap list` output. If two adjacent snapshot files show little difference in the cumulative values, most of the data referenced by the snapshots is the same. In this case, removing one of the snapshots doesn't free up much space.

If you find snapshots confusing and hard to manage, use the default snapshot schedule and the default snapshot reserve because these settings are appropriate for most environments. As the file system grows, check how much space the snapshots consume. If it seems high, decrease the amount of time that snapshots are kept or increase the snapshot reserve. As you use snapshots, continue to watch the statistics change over time. The statistics will help you gain a better understanding of how snapshots work.

Accessing snapshots from clients

Snapshots can be accessed by any user with the appropriate permissions. Every directory in the filer's active file system contains a directory named *.snapshot*, through which you can access old versions of files in that directory. How users gain access to snapshots depends on the file-sharing protocol used.

Obtaining access to snapshots from an NFS client

Figure 10-3 shows the tree structure of a client with the filer's file system mounted on */n/toaster/home*.

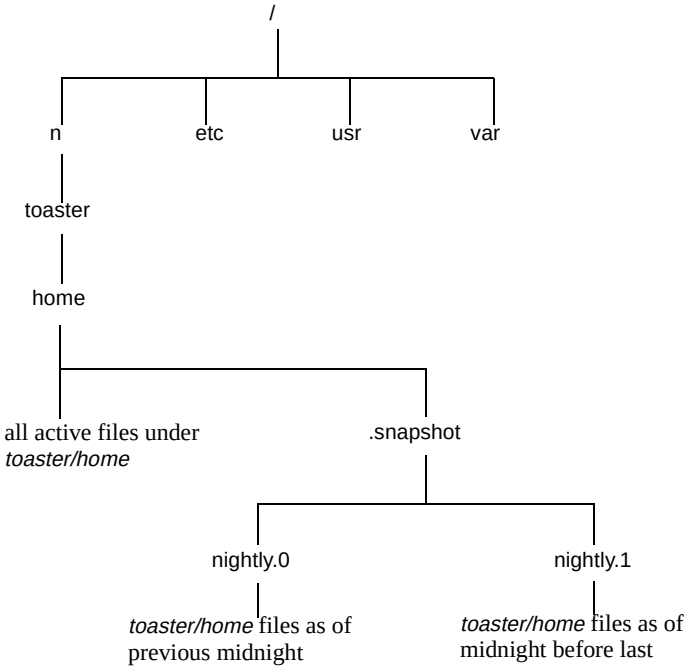


Figure 10-3. Filer toaster home directory tree

In the example in Figure 10-3, a client with a filer whose name is "toaster" mounted at */n/toaster/home* can obtain access to snapshots by way of */n/toaster/home/.snapshot*. This might seem to contradict the explanation of snapshot access in the preceding section, because it shows a *.snapshot* directory only at the mount point instead of in every directory in the tree.

Actually, the *.snapshot* directory in the mount point is "real" to make the `pwd` command work, whereas the *.snapshot* directory in all other directories is "magic." A "magic" directory is one that can be accessed when it is referenced by name but does not show up in a directory listing. For example, at the mount point of a filer file system, a directory listing looks like this:

```
% ls -a
.          .snapshot     dir1      foo
..         bar           dir2
```

The same command entered in a directory below the mount point does not show the *.snapshot* directory, but an `ls` of *.snapshot* does:

```
% ls -a
.             text1     text2
..            blivit
% ls .snapshot
hourly.0      hourly.4  nightly.0 nightly.4
hourly.1      hourly.5  nightly.1 nightly.5
hourly.2      hourly.6  nightly.2 weekly.0
hourly.3      hourly.7  nightly.3 weekly.1
```

If *.snapshot* were to show up in every directory, it would cause many commands to work improperly. For instance, all recursive removes would fail because everything below *.snapshot* is read-only. Recursive copies would copy everything in the snapshots as well as files in the active file system, and a `find` command would generate a list much longer than expected.

Obtaining access to snapshots from a CIFS client

To CIFS clients, the snapshot directory appears only at the root of a share. For example, if a share named dir has been defined for the */home/dir* directory, only the */home/dir/.snapshot* directory will be visible to the clients. The name of this directory is displayed as *~snapshot* on clients that support long file names and as ~snapsht on clients that support only short file names.



NOTE: The snapshot directory is visible in that it appears in a directory listing or File Manager display if you configure your client operating system to show hidden files.

In each directory within the share, a snapshot directory exists but is not visible to clients. For example, if the client operating system supports long file names, the applications on that operating system can use the snapshot at each level of the share by using *.snapshot*, *~snapshot*, or *~SNAPSHT* as the directory name. You cannot, however, display the directory name in any listing.

Determining snapshot versions

The best way to find all file versions preserved in snapshots is to use `ls`. This example shows how to find all versions of *foo*:

```
% ls -l foo .snapshot/*/foo
-rw-r--r--  1 smith 0 Jan 14 09:40foo
-rw-r--r--  1 smith 0 Jan 13 18:39.snapshot/nightly.0/foo
-rw-r--r--  1 smith 0 Jan 12 19:17.snapshot/nightly.1/foo
```

The version of *foo* in the active file system was last modified on January 14, but the old versions available in the snapshots were modified on January 13 and January 12. Although users can use standard UNIX commands to examine the saved versions of *foo*, they cannot modify or remove these older versions because everything beneath *.snapshot* is read-only.

Determining access times

When the filer creates a snapshot, the access time of each file in the snapshot is updated to the snapshot creation time. Thus, you can use the `ls -lu` command, which shows the access times of files, to determine when snapshots were created. The following is an example of the `ls -lu` command:

```
% ls -lu foo .snapshot/*/foo
-rw-r--r--  1 smith 0 Jan 14 09:40foo
-rw-r--r--  1 smith 0 Jan 14 00:00.snapshot/nightly.0/foo
-rw-r--r--  1 smith 0 Jan 13 00:00.snapshot/nightly.1/foo
```

Effects of snapshots on quotas

Quotas do not count space consumed by snapshots. If snapshots were included in the quota calculations, users could end up in a state where they could not create any new files until all snapshots containing their old files expired.

Snapshot options

The following options for the `options` command affect snapshots:

`nosnap` Temporarily disables automatic snapshots. By default, this option is disabled and automatic snapshots take place regularly.

`nosnapdir` Makes the *.snapshot* directory that is normally present at client mount points or the root of the CIFS share invisible. It also turns off access to the *.snapshot* directory and all *.snapshot* directories under the mount point or the root of the CIFS share. By default, this option is disabled.



NOTE: The dump command does not work properly if the `nosnapdir` option is on.

Chapter 11: **Dump and restore**

This chapter describes:

- using the filer's `dump` and `restore` commands to back up files to tape (creating a system backup) and restore files from tape
- using a filer's tape drive from another system
- using NDMP support for third-party backup applications

The term "system backup" means backing up files from the filer to tape, not the creation of on-line, read-only copies of the entire file system (snapshots), which is described in Chapter 10, "Snapshots."

The term "active file system" means the file system on the filer excluding snapshots.

If your filer is running *only* NFS, you can also do a system backup using UNIX commands. If your filer is running *only* CIFS, you can also do a system backup using CIFS utilities.



CAUTION: If your filer delivers both NFS and CIFS service, use only the filer's `dump` and `restore` commands. If you use any other backup utilities, file system information such as 8.3 names, permissions, or DOS attributes might be missing or incorrect for the restored files. Furthermore, symbolic links and other types of files not known to CIFS clients might not be restored if backup and restore utilities are used on a CIFS client.

Backing up the filer to tape

The most efficient and reliable way of backing up the file system is to use the filer's `dump` command, which creates backup files compatible with the SunOS 5.*x* / Solaris 2.*x* `ufsdump` format. You can use `dump` on a directory in the active file system (in which case `dump` creates a snapshot and dumps that) or on a snapshot.

If you are using NFS only, you can back up the filer using commands such as `tar` and `cpio` from a client system. However, use `tar` and `cpio` with caution because some versions have bugs that cause them to fail with file systems that contain long path names, unusual file names, or hard links.



NOTE: If a file has a UID or GID longer than 16 bits, the filer uses the Solaris 2.5.1 scheme for putting that file on the `dump` tape. Files and directories with UIDs or GIDs longer than 16 bits restored on systems that use Solaris earlier than 2.5.1 might not be restored with the correct UIDs or GIDs.

Features of the filer's dump command

The `dump` command has the following characteristics:

- It can write to standard output, to a remote tape device on a host supporting `rmt` (remote magnetic tape protocol), or to a local tape device attached to the filer.
- It can back up an active subtree of the file system, which can be a file, directory, or the entire file system.
- It supports backing up a large file system or subtree to multiple files on a single tape or multiple files on multiple tapes. Each file on the tape is called a "volume." With this feature, you can back up a file system to a remote host even if the file system size exceeds the maximum size allowed for a single tape file on the remote host.

Planning for system backups

Before initiating the dump command, it is important for you to consider the following and write down the requirements and limitations of your environment:

- If you dump to a remote host, check the maximum size of the dump file supported by the host. Some UNIX systems (for example, systems running SunOS 4.1.1 or earlier) don't support dump files larger than 2 GB. In the dump command, you can specify the dump file size in blocks to avoid exceeding the limit imposed by the remote host.

- If you dump to a tape attached to the filer, the dump file can be greater than 2 GB. If you use a host other than the filer for restoring the files, you must determine whether the host supports dump files larger than 2 GB. If it does not, limit the size of the dump file by specifying the appropriate dump file size in the dump command.

- Verify the blocking factor supported by the system on which the dump tape will be restored; some systems, including the filer, don't support blocking factors larger than 63 KB. If you dump to a local tape drive, the filer restricts the blocking factor to 63 KB. However, if you dump to a remote host that allows a higher blocking factor, do not exceed the maximum blocking factor supported by the system to be used for restore.

- If you plan on restoring quota trees, make sure that every quota tree is listed in the */etc/quotas* file. If you want to restore your quota trees with your file system, you must recreate the quota tree before proceeding with the restore. Listing all of them in the */etc/quotas* file provides an accurate list of the quota trees to recreate. For information about quota trees, see "Creating a quota tree" on page 240.

CAUTION: Plan on verifying that dumps have completed successfully. For example, if you are executing a dump through the `rsh` command, and the dump needs more tapes than are available in your tape device, dump will display the following messages (among others):

```
DUMP: ran out of tape, and can't ask questions when
run via "rsh".
DUMP: The ENTIRE dump is aborted.
```

If you did not check whether the dump succeeded, and kept putting in new tapes and executing additional dumps, you would have tapes containing no dumps, and you might lose data.

Dump and restore for large file systems

Network Appliance recommends that individual dumps be segmented. If you segment the dump, individual dumps can complete sooner.

To restrict dump trees to a predetermined size, you can use quota trees when you are planning or modifying your file system to make sure that directories to be dumped do not grow beyond a predetermined size. For information about quota trees, see "Creating a quota tree" on page 240.

Determining the number of tapes required

To figure out how many tapes you need for backing up a subtree, follow these steps:

1. Verify the size of the dump tape to determine the number of tapes required. Consult your tape drive documentation for information about compression. If you dump to a remote host, check the maximum capacity of the tape drive on that host. When calculating the number of tapes needed, remember that `dump` can split a dump file into multiple volumes on multiple tape drives.

   To determine the capacity of the tape drive attached to the filer, enter the `sysconfig -t` command. The following is an

example for a filer containing a Quantum DLT7000 tape drive. The drive is attached to SCSI adapter 2 and has SCSI ID 4.

```
toaster> sysconfig -t
Tape drive (2.4)  Quantum DLT7000
rst0l  -  rewind device,         format is: 81633 bpi 20 GB
nrst0l  -  no rewind device,     format is: 81633 bpi 20 GB
urst0l  -  unload/reload device, format is: 81633 bpi 20 GB
rst0m  -  rewind device,         format is: 81633 bpi 40 GB (w/comp)
nrst0m  -  no rewind device,     format is: 81633 bpi 40 GB (w/comp)
urst0m  -  unload/reload device, format is: 81633 bpi 40 GB (w/comp)
rst0h  -  rewind device,         format is: 85937 bpi 35 GB
nrst0h  -  no rewind device,     format is: 85937 bpi 35 GB
urst0h  -  unload/reload device, format is: 85937 bpi 35 GB
rst0a  -  rewind device,         format is: 85937 bpi 70 GB (w/comp)
nrst0a  -  no rewind device,     format is: 85937 bpi 70 GB (w/comp)
urst0a  -  unload/reload device, format is: 85937 bpi 70 GB (w/comp)
```



NOTE: Stackers and libraries that are connected to the filer through SCSI, such as the Breece-Hill library, are not visible to the `sysconfig` command.

2. To determine the sizes of various files in a file system to be backed up, use the following procedure on a client that has mounted the file system:

   a. Use the `quota report` command to list the usage in each of the quota trees to be dumped. For example:

```
client% rsh filer quota report
                    K-Bytes          Files
Type      ID      Used    Limit     Used   Limit  Path Name
----- ------  -------- --------  ------- -------  -------------
 tree      6     21608  1048576     1350   10240  /finance
 user   1086      9172  1048576      129 1048576  /home/eirik
 tree      3   3018520 10485760   403451 1048576  /marketing
 tree      4   6247100 20971520  1561775 2097152  /news
 tree      1   3018328 10485760   406421 1048576  /users
client%
```

b. If any directories to be backed up are not in the quota report (because they are not quota trees or they are unlisted quota trees), go to the client mount point of the filer root directory and enter the `du -s` command. For example:

```
client% cd /n/toaster
client% du -s etc home
4108 etc
5510100 home
client%
```

For example, if each tape you use stores up to 10 GB of data and the host on which you dump and restore supports dump files larger than 2 GB, you can dump the *marketing* and *news* subtrees to one tape volume, then load another tape and dump the rest of the file system (that is, the *etc*, *finance*, *home*, and *users* subtrees). Remember that when you dump multiple subtrees on one tape, you must use the `norewind` tape device in the `dump` command. For information about the `dump` command, see "The dump command syntax" below.

Supported tape drives

If you are using the tape drive on a client system, you can dump files to or restore files from any tape drive on the client.

If you need to add an external tape drive to your filer, you can purchase a drive from Network Appliance or another vendor. Refer to the hardware guide for your filer model for instructions on attaching an external tape drive and a list of supported tape drives; using an unsupported drive might cause the filer to hang.

The dump command syntax

Because the filer supports `rsh`, you can automate system backups via shell scripts. Backing up the file system using `rsh` offers another advantage over starting `dump` on the console: you can be sure that `dump` will not be aborted by a Ctrl-C inadvertently entered from a host connected to the filer through `telnet`.

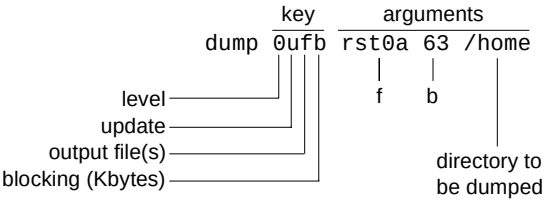
The information provided in this section applies whether the command is entered at the filer prompt or through `rsh`.

The following is the command syntax for `dump`:

`dump [ option_keys [ arguments ]] [ subtree ]`

Put a space after `dump`, then add the option keys to the dump command together with no intervening spaces, then the arguments to each option in the order of the options. You can append a subtree parameter. The option keys are listed in Table 11-1 on page 280.

A typical dump command looks like the following:



The above command dumps one directory (*/home*) on a local tape device (*rst0a*).

NOTE: The filer `dump` command is similar to the SunOS `dump` command. An important difference between the commands is that the blocking factor in the filer `dump` command is in *kilobytes*, while in the SunOS `dump` command, the blocking factor is in 512-byte units.

The *subtree* parameter specifies a file or directory that resides within the active file system or within a snapshot. If the subtree is within the active file system, the `dump` command creates a snapshot named *snapshot_for_dump.n* (where *n* is a number that is reset to 0 at boot time and then incremented by one after each *snapshot_for_dump.n*), from which it copies the data to tape. The filer automatically deletes *snapshot_for_dump.n* that it created

when it is finished with the dump. If you are doing a dump of an existing snapshot, no *snapshot_for_dump.n* is created by that dump.



NOTE: If multiple dump commands run simultaneously, each dump creates its own s*napshot_for_dump.n*. For example, these files might exist on your filer when several dump commands are taking place: *snapshot_for_dump.0*, *snapshot_for_dump.1*, and *snapshot_for_dump.2*. (Before Release 4.0, the snapshot created by dump was named *snapshot_for_dump*.)

When dumping multiple subtrees, you must also specify a dump name. For more information about dumping multiple subtrees, see "Dumping multiple subtrees" on page 283.



CAUTION: Do not try to delete a snapshot while dump is in the process of backing up the snapshot; this will cause the filer to crash. If dump is invoked on a snapshot whose name allows the snapshot scheduler to delete it (an hourly, nightly, or weekly snapshot), the snapshot scheduler can also cause a crash by attempting to delete such a snapshot. To avoid such crashes, do one of the following:

- Insure that the dump will finish before the snapshot being dumped comes up for deletion by the snapshot scheduler.
- Rename the snapshot to protect it from the snapshot scheduler.

Table 11-1 describes the option keys for the `dump` command.

Table 11-1. Option keys for dump

Key	Argument	Meaning
0 – 9		Specifies the dump level. Level 0 is a full backup. Use this option in the same way as you would on a UNIX system (for example, SunOS). If specified, this must be the first key. The default is 0.
u		Updates the */etc/dumpdates* file after a successful dump. This ASCII file contains information about each dump. Its format is identical to one created by the SunOS `dump` command.
f	*files*	Specifies the files to which `dump` writes. You can specify a comma-separated list of files, and each file is considered a dump volume. The following are legitimate file names: • local tape drive names (for example, `nrst0a`). For more information on tape drive names, refer to the *na_tape*(4) man page. • remote tape drive names, which must be specified with the host name (for example, `tapemachine:/dev/rst0,/dev/rst1`) • standard output, which is specified as a dash (–).
b	*factor*	Specifies the blocking factor in kilobytes. If you dump to local tape drives, use a blocking factor no greater than 63 KB. Make sure that the blocking factor you specify is supported on the system on which the dump file will be restored.

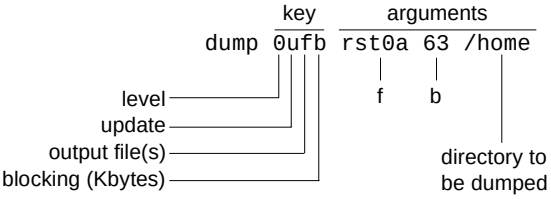
Table 11-1. Option keys for dump (continued)

Key	Argument	Meaning
B	*blocks*	Specifies the maximum size of each dump volume in 1024-byte (1-KB) blocks. When the subtree being dumped exceeds one volume, `dump` automatically writes to the next tape device specified with the `f` option. If there are no more files in the list, `dump` prompts you for a new tape to be loaded to the last device. [a] For example, to limit the dump volume size to 2 GB (so that the dump file can be restored on a UNIX system with a 2-GB limitation), specify 2097151 blocks. For information about uncompressed and compressed capacity of your tape drives, consult your tape drive documentation. Specify the dump volume size only if the limit for a dump file system on the host is 2 GB. The B option replaces the `d` and `s` options in the SunOS `dump` command, which are retained for compatibility.
d	*bpi*	Specifies the tape density in BPI and is included for compatibility with UNIX. sw
s	*size*	Specifies the size in blocks of the volume being dumped to and is included for compatibility with UNIX. Network Appliance recommends that you use the `B` option instead.
l		Specifies a dump containing named subtrees of the directory specified in the final argument of the command line. The `dump` command reads in subtree names from standard input (entered one per line with the list terminated by one blank line) and dumps only those subtrees.
n	*name*	Specifies the name of the dump for */etc/dumpdates*. Use this option when dumping multiple subtrees.

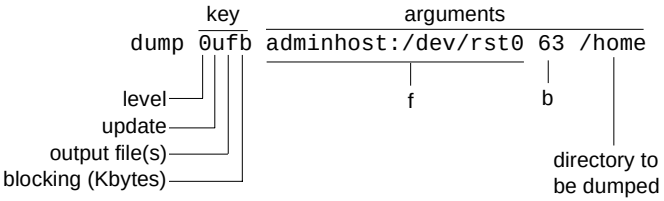
a. You can display detailed information about the capacity of a filer's tape drive using the **`sysconfig -t`** command.

The following are several examples of the dump command for backing up the */home* directory. They all use a blocking factor of 63 KB.

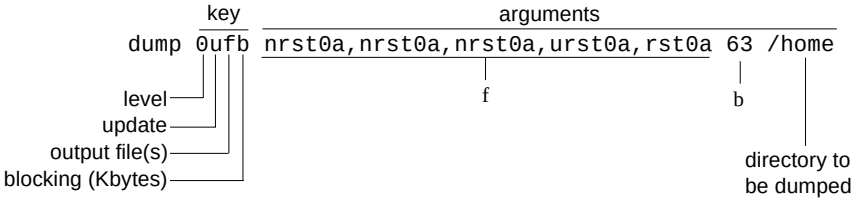
- To dump to one volume on a local tape device:



- To dump to one volume on a remote tape device:



- To dump to multiple volumes on a tape stacker or library without operator intervention:



- To back up a directory in an existing snapshot:

  dump 0unfb /home rst0a 63 /.snapshot/weekly.0/home

- To back up a directory to a local DLT 2000 tape drive:

  dump 0ufb rst0a 63 /home

- To back up an entire file system in sequential mode to a Breece-Hill library when it is estimated that four tapes will be required:

```
dump 0ufb urst0a,urst0a,usrst0a,nrst0a 63 /
```



NOTE: When you back up a directory in an existing snapshot, use the `n` switch to ensure that the dumpdates entry is correct.

For information about using the tape drive on another filer to make backup tapes, see "Using a filer's tape drive from another system" on page 294. For a listing of the device names to use for the local tape drive, run the `sysconfig -t` command.

See the *na_dump*(1) and *na_tape*(4) man pages for more information about the `dump` command.

Dumping multiple subtrees

You can dump multiple subtrees with a single `dump` command by using the `l` and `n` options. The `n` option requires the name of the dump for the */etc/dumpdates* file. The `l` option specifies that `dump` will read path names from its standard input. The following is the syntax of the command for dumping multiple subtrees:

```
dump [ option_keys ] [ arguments ] [ dumpname ]
   [ directory ]
```

After you enter the `dump` command, the filer displays some messages and a prompt for the names of the subtrees. Enter the name of each subtree as a path name relative to the directory you specified on the command line. Do not specify ".." as a subtree or specify subtrees that contain symbolic links. When you finish typing subtree names, press Ctrl-D as shown in the following example:

```
toaster> dump 0ufnl nrst0a dir.1.3.5 /home
DUMP: creating "snapshot_for_dump.0" snapshot.
creating.................................................
DUMP: Date of this level 0 dump: Tue Jun  3 12:47:14 1997
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /home to nrst0a
DUMP: mapping (Pass I) [regular files]
DUMP: Reading file names from standard input
dir1
dir3
dir5
^D
DUMP: 96% done, finished in 0:07
DUMP: 3252426 tape blocks
DUMP IS DONE
DUMP: deleting "snapshot_for_dump.0" snapshot.
deleting.................................................
```



CAUTION: When you do the level 0 dump, make a note of the subtrees you are dumping and the dump name. For subsequent incremental dumps, always specify the same dump name and the same subtrees. The dump command does not enforce the correspondence between the subtree names and the dump name. Therefore, you *must* keep track of this information yourself. If you do not specify the same subtrees for the incremental dumps, you will not be able to do incremental restores.

Restoring dump files

You can restore a dump file created by the filer's `dump` command in one of two ways:

- Using the `restore` command on the filer to restore the file from a local or remote tape drive. If you use a remote tape, the host for the tape drive must support the `rmt` protocol.
- Using the UNIX `ufsrestore` command on a client that supports the SunOS 5.*x* / Solaris 2.*x* `ufsdump` format.



CAUTION: Use the SunOS `ufsrestore` command only if your filer runs NFS exclusively. If your filer runs the CIFS protocol, do not use the SunOS `ufsrestore` command. If you do so, file system information such as 8.3 names, permissions, or DOS attributes might be missing or incorrect for the restored files.

The `restore` command is usually used in the following situations:

- The entire file system is damaged and unusable. If this happens, call Network Appliance Technical Support to see whether you can repair the file system. If the file system cannot be repaired, you need to create a new file system and restore the entire file system from tape. If you can repair the file system but some files are deleted during the repair process, you then need only restore the missing files from tape.
- If no disk slots are available for storage expansion, you can back up the entire file system, replace the current disks with disks of greater capacity, create a file system on the new disks, and restore the file system from tapes.
- If some files on the filer are corrupted, you can restore the subtree containing the files.

CAUTION: Before trying to create a new file system on your filer, call Network Appliance Technical Support for instructions. Creating a new file system destroys all data currently on your disks.

Restoring quota trees

Before using the `restore` command, you must recreate quota trees, otherwise, the `restore` command will create ordinary directories in place of the quota trees. Obtain a list of the quota trees in the system you dumped, then recreate them.

Generating a list of quota trees

You can generate a complete list of quota trees in an active file system by entering the command:

```
quota qtree
```

If you do not have access to an active file system, use the record of the */etc/quotas* file you made as described in "Restricting disk usage by imposing disk quotas" on page 232.

Recreating quota trees

You recreate each quota tree by issuing the following command:

```
quota qtree tree
```

where *tree* is a quota tree.

The restore command

This section discusses only the syntax of the `restore` command. Refer to "Locally restoring a subtree" on page 290 or "Restoring the entire file system" on page 291 for more information and step-by-step procedures for restoring files. For information about using the tape drive on another filer, see "Using a filer's tape drive from another system" on page 294.

The `restore` command is similar to the one in SunOS, except that the filer's `restore` command doesn't support the `i` argument of the SunOS `restore` command This argument enables you to

specify interactively individual files and directories to be restored from a dump tape. However, you can restore individual files and directories using the `x` argument of the filer's `restore` command.



CAUTION: You can use the filer's `restore` command only to restore dump files created by the filer. Do not try to use the filer's `restore` command to restore dump files created by other servers or workstations.

You can enter the `restore` command on a trusted host via `rsh`. In addition, you can run multiple `restore` commands by starting each from a remote shell.



NOTE: Do not run multiple `restore` commands simultaneously. Wait for the first restore to complete before starting a second restore.

The restore command syntax

The following is syntax of the `restore` command:

```
restore [function_key] [option_keys]
   [ arguments ] [ subtree ]
```

After the function key (described in Table 11-2), put all of the option keys to the `restore` command together with no intervening spaces, then the arguments to each option in the order of the options, and end with the subtree parameter.

The `r`, `R`, `t`, and `x` keys specify the functions of the restore. You can have at the most one of these in the command. They are explained in Table 11-2:

Table 11-2. Restore command function keys

Key	Meaning
r	Rebuilds the file system or subtree. If you are applying incrementals, this must be the only option.
R	Specifies a particular tape if the dump file is on a multi-volume set. Use this argument to restart `restore` if it has been interrupted.
t	Lists all the file names in the archive file that you specify with the `f` option. If you don't specify any archive, the root directory of the dump tape is listed.
x	Extracts individual files or directories from the backup tape.

You can include zero or more of the option keys listed in Table 11-3 after the function key with no intervening spaces and no space after the function key.

Table 11-3. Restore option keys

Key	argument	Meaning
b	*blocks*	Specifies the block size. Use the same block size you used when dumping the file.
f	*archive*	Specifies the name of the archive file. If you specify "-", `restore` reads from standard input.

Table 11-3. Restore option keys (continued)

Key	argument	Meaning
D	*pathname*	Specifies the full absolute pathname of a directory into which the files are restored. Without the pathname, the files are restored to the directory from which they were dumped.
h		Prevents hierarchical restoration.
m		Extract by inode number rather than file name.
s	*number*	Specifies the number of the file if multiple dump files exist on a tape. File numbering starts at 1.
v		Specifies that `restore` takes place in verbose mode. That is, `restore` displays each file name preceded with its file type.
y		Specifies that `restore` not ask whether it should abort when getting an error. That is, if there are bad blocks, `restore` skips over them and continues.

Locally restoring a subtree

This section provides an example illustrating how you can restore a subtree by issuing the `restore` command on the filer.

In this example, the name of the filer is "toaster," the subtree to be restored is */etc/home*, and the mount point for the subtree on the client is */n/toaster/home*. Also, the procedure assumes that */n/toaster/home* was the subtree dumped to the tape; it is not a directory within a subtree.

The following is the procedure for restoring */home*:

1. If you want to delete all files in the subtree before the restore, enter the following command on the client:

   ```
   client% rm -rf /n/toaster/home/*
   ```

   Then enter the `ls -a` command to list files in */n/toaster/home* whose names begin with a period (.). Use the `rm` command to remove those files except the current directory (.) and the parent directory (..).

2. Install the tape that contains the level 0 dump file for the */home* subtree in a local or remote tape drive.

   If you restore from a remote host, make sure that the filer and the remote host with the tape drive (the tape host) can perform hostname lookups. If either host relies on its */etc/hosts* file, that file must have an entry for the other host.



NOTE: The `restore` command doesn't support incremental-only restores. You must begin the restoration from a level 0 dump.

3. If you are restoring from a local tape, enter the `sysconfig -t` command on the filer to determine the name of the tape device. If you are restoring from a remote tape, specify the remote tape in this format:

   hostname:*device_name*

The following is the sample tape device name used in this procedure:

```
unixhost:/dev/rst0a
```

4. Enter the following command on the filer:

   ```
   toaster> restore rfD rst0a /home
   ```

   This command restores the dump file in rst0a to the */etc/home* directory, using the blocking used on the tape. If the dump file for the subtree is in a multivolume set, `restore` prompts you for the next volume when appropriate.

5. After the restore is finished, remove the tape from the drive. Install another tape that contains the next lowest level of incremental dump. For example, if you have dump tapes containing level 1, level 2, and level 3 dumps of */etc/home*, load the tape with the level 1 dump in the drive and repeat the `restore` command. Repeat this step until the dump file of the highest level (level 3 in this example) is restored.

6. From the client, use the `rm` command to remove the *restore_symboltable* file in the directory that you just restored. This file was created by `restore` to pass information between different levels of restore. For example:

   ```
   client% rm /n/toaster/home/restore_symboltable
   ```

The restore process is now complete.

Restoring the entire file system

This section describes how to restore the entire file system. It assumes that you have already created a new file system on your filer.

If you have backed up the entire file system to a dump file, follow the procedure in "Locally restoring a subtree" on page 290, except that you should use the appropriate mount point on the client, and use / as the directory to which the dump file is restored.

If you have never backed up the entire file system, restore each subtree that you have dumped. Suppose your filer contains two directories, */etc* and */home*, and you have backed them up as subtrees, follow these steps to restore the entire file system:

1. Mount the filer's / directory on a client. For example:

   ```
   client% mkdir /n/toaster/root
   client% mount toaster:/ /n/toaster/root
   ```

2. If you want to restore quota trees, create directories on the client using the `quota qtree` command that will be used as the top of the subtrees to be restored. For example:

   ```
   client% cd /n/toaster/root
   client% quota qtree etc
   client% quota qtree home
   ```

3. Install the tape that contains the level 0 dump file for the */etc* subtree in a local or remote tape drive.

   If you restore from a remote host, make sure that the filer and the remote host with the tape drive (the tape host) can perform host name lookups. If either host relies on its */etc/hosts* file, that file must have an entry for the other host.



NOTE: The `restore` command doesn't support incremental-only restores. You must begin the restoration from a level 0 dump.

4. If you are restoring from a local tape, enter the `sysconfig -t` command on the filer to determine the name of the tape device. If you are restoring from a remote tape, specify the remote tape in this format:

   hostname:device_name

   The following is the sample tape device name used in this procedure:

   unixhost:/dev/rst0a

5. Enter the following command on the filer:

   ```
   toaster> restore rfD rst0a 63 /etc
   ```

   This command restores the dump file in rst0a to the */etc* directory. If the dump file for the subtree is in a multivolume set, `restore` prompts you for the next volume when appropriate.

6. After the restore is finished, remove the tape from the drive. Install another tape that contains the next lowest level of incremental dump. For example, if you have dump tapes containing level 1, level 2, and level 3 dumps of */etc*, load the tape with the level 1 dump in the drive and repeat the `restore` command. Repeat this step until the dump file of the highest dump level (level 3 in this example) is restored.

7. From the client, use the `rm` command to remove the *restore_symboltable* file in the directory that you just restored. This file was created by `restore` to pass information about different levels of restore. For example:

   ```
   client% rm /n/toaster/home/restore_symboltable
   ```

8. Repeat steps 5 through 7 for the */home* subtree.

The restore process is now complete.

Using a filer's tape drive from another system

You can use a filer's tape drive to dump and restore files located on a UNIX system or on another filer. The filer with the tape drive must allow access from the other system. That is, the host or filer from which a `dump` (or `ufsdump`) or `restore` (or `ufsrestore`) command is issued must be listed in the following configuration files:

- In */etc/hosts.equiv* on the filer with the tape drive. Alternatively, you can specify both the host and user in */etc/hosts.equiv*. The filer's */etc/hosts.equiv* is a simple file containing entries in this format:

  ```
  hostname [ username ]
  ```

  For more information about */etc/hosts.equiv*, see the *na_hosts.equiv*(5) man page.

- In */etc/hosts* on the filer with the tape drive or in the dns database if the filer is using dns.

Also, the filer with the tape drive must be added to the other system's */etc/hosts*. See "Editing the client's /etc/hosts file" on page 39 for information on how to add the filer to the client's */etc/hosts*.

In the commands for dumping and restoring, specify the filer's tape drive in this format:

```
filer:device_name
```

The following examples show how to use a filer's tape drive from an SVR4 UNIX system such as Solaris 2*x*:

- To dump a file system that resides on a UNIX system:

  ```
  client% ufsdump 0uf toaster:nrst0a /dev/sd0g
  ```

- To restore a directory from a filer's tape drive to a UNIX system:

```
client% cd /filesystem
client% ufsrestore rf toaster:rst0a
```

The following examples show how to use a filer's tape drive from a UNIX system other than an SVR4 UNIX system:

- To dump a file system that resides on a UNIX system:

```
client% dump 0uf toaster:nrst0a /dev/sd0g
```

- To restore a directory from a filer's tape drive to a UNIX system:

```
client% cd /filesystem
client% restore rf toaster:rst0a
```

To use a filer's tape drive from another filer, follow the directions in "Dumping multiple subtrees" on page 283, "Locally restoring a subtree" on page 290, and "Restoring the entire file system" on page 291 to dump or restore files, using `filer:device_name` as the tape drive name in these procedures.

Backing up the filer using NDMP

You can use third-party backup management programs that support the Network Data Management Protocol (NDMP), v1 or v2, to back up the filer. The filer can be backed up by NDMP to these devices:

- tape drives, libraries, or stackers attached to the filer
- tape drives, libraries, or stackers attached to the workstation that runs the backup program
- tape drives, libraries, or stackers attached to a workstation or filer on your network

Figure 11-1 shows an example of backing up the filer to various devices using NDMP.

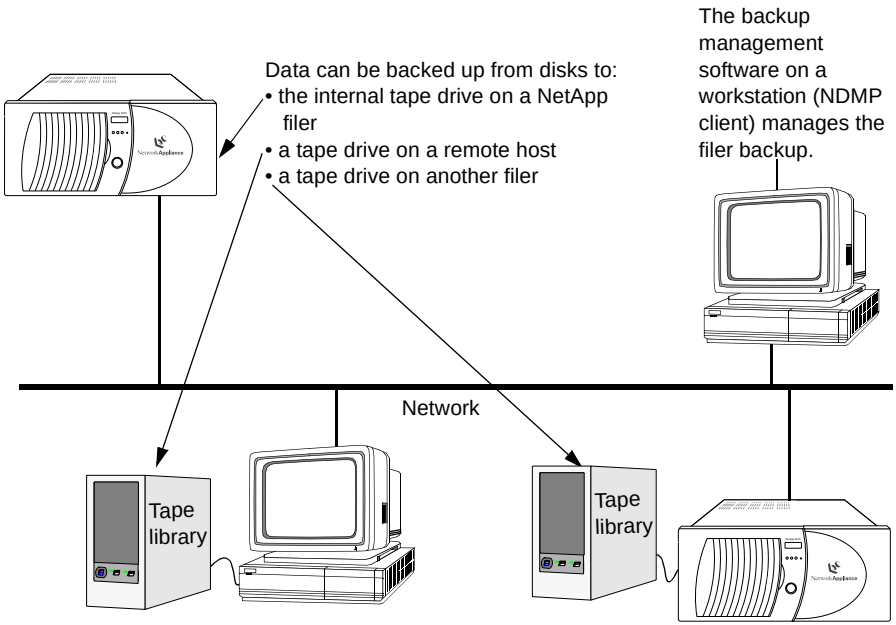


Figure 11-1. Using NDMP to back up data to tape

Use the ndmpd command for enabling NDMP support on the filer and for obtaining information about backup and restore operations that use NDMP. The following is the syntax for the ndmp command:

```
ndmpd [ on | off | status | probe [ session ] ]
```

The on option enables NDMP. The off option disables NDMP. After you enter NDMPD off, the filer continues processing existing requests but rejects new ones.

If you want the filer to run NDMP after each reboot, remember to put the NDMP on command in the */etc/rc* file.

The status option displays the current state of each of the five possible NDMP sessions. Sessions are numbered 0 to 4. In the following example, there is only one active session.

```
toaster> ndmpd status:
ndmpd ON.
Session: 0
  Active
  tape device: open
  state:      Idle
Session: 1
  Inactive
Session: 2
  Inactive
Session: 3
  Inactive
Session: 4
  Inactive
```

The probe option displays detailed information about the NDMP session specified. Using the probe option without a session parameter displays NDMP information for all five sessions. In the following example, there is only one active session.

```
toaster> ndmpd probe 0:
Session: 0
   pid:                    0xfffffc00005d1e10
   is_active:              TRUE
   eof:                    FALSE
   authorized:             TRUE
   my_sequence:            32
   tape.id:                16973824
   tape.device:            nrst0a
   tape.mode:              Read/Write
   protocol_version:       1
   record_size:            61440
   data.bu_type            nil
   data.state:             Idle
   data.operation:         None
   data.halt_reason:       N/A
   data.pause_reason:      N/A
   data.seek_position:     0*0
   data.blocks_processed:  0
   data.est_blocks_remain: 0
   data.est_time_remain:   0
   data.window_begin_byte: 0*0
   data.abort:             FALSE
```

Controlling tape devices

You can control tape devices with the `mt` command. This section describes how to do the following:

- Move a tape to the next file (fsf).
- Append a dump on a tape by moving to the end of data (eom).
- Rewind a tape.
- Take a tape off line.
- Display status information about a tape device.

The examples in this section use the device nrst0a – a no-rewind tape device, unit zero, at the highest format (density).

The mt command syntax

The syntax of the `mt` command is:

`mt [-f|-t]` *`tapedevice command`* `[` *`count`* `]`

This section discusses only the `eom`, `fsf`, `rewind`, `offline`, and `status` commands. Keep in mind the following:

- The `-f` and `-t` options are interchangeable as far as the filer is concerned. Only the `-f` option is shown here.
- For additional information on controlling tape devices and detailed information about the `mt` command, consult the *mt*(1) man page.
- For information about the format of a filer tape device name, consult the *tape*(4) man page.

Moving a tape to the end of data

You can append material such as a dump on a tape. To do so, use the `eom` command, which moves the tape to end of data (end of media if tape is full), as in the following example:

```
toaster> mt -f nrst0a eom
```

After this command, you can write to the remainder of the tape. Make sure that there is enough tape for your additional data.

Appending a dump

To skip over a previously created dump file and append a dump onto a tape device, use the `fsf` command as in the following example:

```
toaster> mt -f nrst0a fsf 1
```

Rewinding a tape

To rewind a tape, use the `rewind` command, as in the following example:

```
toaster> mt -f nrst0a rewind
```

Taking a tape off line

To rewind the tape and, if appropriate, take the drive unit off-line by unloading the tape, use the `offline` command as in the following example:

```
toaster> mt -f nrst0a offline
```

Displaying status information

To display status information about the tape unit use the `status` command, as in the following example:

```
toaster> mt -f nrst0a status
```

which displays information like the following:

```
Tape drive: Quantum DLT7000
Status: ready, write enabled
Format: 85937 bpi 70 GB (w/comp)
fileno = 0  blockno = 95603  resid = 0
```

Chapter 12: System information and performance

This chapter describes the following:

- how to use the `version`, `sysconfig`, `sysstat`, `uptime`, and `netstat` commands to display information about system configuration and statistics
- how to improve filer performance by adding hardware or setting software options

Displaying the Data ONTAP version

To display the version of data ONTAP currently running on a filer, use the `version` command. The display shows the version number and the date of the version, as shown below:

```
toaster> version
NetApp Release 4.3: Fri Sep 12 03:06:00 PDT 1997
```

Displaying filer configuration using sysconfig

The `sysconfig` command displays information about the filer's hardware configuration. The exact types of information displayed depend on the command options. (A `sysconfig` command without options is like `sysconfig -v`, but that the output is more concise. See "Displaying overall filer information using sysconfig -v" on page 306.)

Displaying disk information using sysconfig -d

The `sysconfig -d` command displays product information about each disk in the filer. Figure 12-1 is an example of the `sysconfig -d` command output.

```
toaster% sysconfig -d
HA# SCSI#   Disk Vital Product Information
--- -----  ------------------------------------------------------
 0     0   0576307
 0     1   00014266
 0     2   00023357
 0     3   00024183
 0     4   PCB=419240822541(ZG40822541 ?); HDA=0000000042964740
 0     5   PCB=419241241883(ZG41241883 ?); HDA=0000000042966220
 0     6   PCB=419232502625(ZG32502625  ); HDA=0000033428371126
 1     0   00911064
 1     1   00931549
 1     2   00009873
 1     3   00019483
```

SCSI (Host) adapter number

SCSI ID

Product information used by disk drive vendors to identify their disk drives

Figure 12-1. sysconfig -d output

Displaying RAID information using sysconfig -r

The `sysconfig -r` command displays raid configuration information about the parity disk (disk 0), data disks, and hot spare disks (if any). This information is useful for the following purposes:

- Locating a disk referenced in a screen message. Refer to "Locating a disk using different identifiers" on page 202 for more information on disk identifiers.
- Determining how much space on each disk is available to the filer. Refer to "Usable space on each disk" on page 204 for more information about disk capacity.
- Determining the status of the disk operations such as raid scrubbing, reconstruction, parity verification, adding a hot spare, and disk failure.

Figure 12-2 shows an example of `sysconfig -r` command output.

```
toaster% sysconfig -r
RAID Disk DISK_ID#  HA.SCSI#     Used (MB/blks)  Phys (MB/blks)
--------- --------  --------     --------------  --------------
parity        9        1.2        4000/8192000    4095/8386728
data 1        0        0.0        1000/2048000    1033/2117024
data 2        2        0.2        2000/4096000    2040/4178872
data 3        3        0.3        2000/4096000    2040/4178872
data 4        4        0.4        2000/4096000    2049/4197520
data 5        5        0.5        2000/4096000    2049/4197520
data 6        6        0.6        2000/4096000    2049/4197520
data 7        7        1.0        2000/4096000    2048/4194680
data 8        8        1.1        2000/4096000    2048/4194680
data 9        1        0.1        2000/4096000    2040/4178872
```

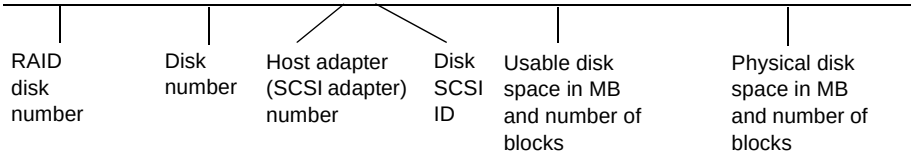


Figure 12-2. sysconfig -r output

You can obtain the information displayed by `sysconfig -r` from snmp, using the Network Appliance custom MIB. For information about snmp, see "Using SNMP" on page 108.

Displaying tape drive information using sysconfig -t

The `sysconfig -t` command displays device and configuration information for each tape drive on the system. Use this command to determine the capacity of the tape drive and the device name before you use the `dump` and `restore` commands.

Figure 12-3 shows an example of the `sysconfig -t` command output:

```
toaster% sysconfig -t
Tape drive (8.1)  Quantum DLT7000
rst0l  -  rewind device,        format is: 81633 bpi 40 GB (w/comp)
nrst0l -  no rewind device,     format is: 81633 bpi 40 GB (w/comp)
urst0l -  unload/reload device, format is: 81633 bpi 40 GB (w/comp)
rst0m  -  rewind device,        format is: 85937 bpi 35 GB
nrst0m -  no rewind device,     format is: 85937 bpi 35 GB
urst0m -  unload/reload device, format is: 85937 bpi 35 GB
rst0h  -  rewind device,        format is: 85937 bpi 50 GB (w/comp)
nrst0h -  no rewind device,     format is: 85937 bpi 50 GB (w/comp)
urst0h -  unload/reload device, format is: 85937 bpi 50 GB (w/comp)
rst0a  -  rewind device,        format is: 85937 bpi 70 GB (w/comp)
nrst0a -  no rewind device,     format is: 85937 bpi 70 GB (w/comp)
urst0a -  unload/reload device, format is: 85937 bpi 70 GB (w/comp)
```

Figure 12-3. sysconfig -t output

Displaying overall filer information using sysconfig -v

The `sysconfig -v` command displays the system's ram size, nvram size, and information about devices in all expansion slots. The following information appears for Network Appliance diagnostic purposes:

- After every SCSI device (other than EMU units), the following lines appear:

  ```
  In-Band Enclosure Services
  0 Swap Events, 0 Enclosure Events
  ```

- After every EMU description line, the following line appears:

  ```
  0 Swap Events, 0 Enclosure Events
  ```

You can specify a slot number to display information about a particular slot. Slot numbers start at 0, where slot 0 is the system board. The figures below show examples of `sysconfig -v` command displays as follows:

- Figure 12-4 shows the display for a FAServer 1400.
- Figure 12-5 shows the display for a NetApp F330 with Digital storage shelves.
- Figure 12-6 shows the display for a NetApp F630 with StorageShelf 2 units. This figure is in two parts.

```
toaster% sysconfig -v
  NetApp Release 4.3: Wed Aug 27 16:40:17 PST 1997
  System ID: 0004526346
  slot 0: System Board 50 MHz (MCC NICE)
          Memory Size:        64 MB
          CMOS RAM Status:    OK
  slot 1: SCSI Host Adapter 0 (Adaptec 1740)
       0: FUJITSU M2694ES-512  812A Size=0.9GB (2117025 blocks)
       1: SEAGATE ST12550N     0013 Size=1.9GB (4178874 blocks)
       2: SEAGATE ST12550N     0013 Size=1.9GB (4178874 blocks)
       3: SEAGATE ST12550N     0013 Size=1.9GB (4178874 blocks)
       4: DEC     DSP3210S     441C Size=1.9GB (4197520 blocks)
       5: DEC     DSP3210S     441C Size=1.9GB (4197520 blocks)
       6: DEC     DSP3210S     435E Size=1.9GB (4197520 blocks)
  slot 2: SCSI Host Adapter 1 (Adaptec 1740)
       0: SEAGATE ST12400N     8650 Size=1.9GB (4194685 blocks)
       1: SEAGATE ST12400N     8650 Size=1.9GB (4194685 blocks)
       2: SEAGATE ST15230N     0252 Size=3.9GB (8386733 blocks)
       3: SEAGATE ST15230N     0252 Size=3.9GB (8386733 blocks)
       6: EXABYTE EXB-85058SQANXR106M0
  slot 3: Ethernet Controller ns0 (KTS)
        MAC Address:    00:c0:f0:01:08:b9 (Twisted-pair Net)
  slot 6: FDDI Network Adapter fi0 (Cisco Turbo)
       MAC Address:    00:40:0b:40:8a:df (UTP,Serial# 63006882)
  slot 7: NVRAM (PEP)
          Memory Size:        2 MB
          Ni-Cad Status:      Good (8.43v)
```

Figure 12-4. sysconfig -v output on a FAServer 1400

```
toaster% sysconfig -v
NetApp Release 4.3: Wed Aug 27 16:40:17 PST 1997
     System ID: 0016777237
     slot 0: System Board 90 MHz (NetApp System Board I)
             Memory Size:        64 MB
     slot 0: SCSI Host Adapter 0 (QLogic ISP 1020)
       0: DEC  RZ28  (C) DEC435E Size=1.9GB (4110480 blocks)
       1: DEC  RZ28B (C) DEC0003 Size=1.9GB (4110480 blocks)
       2: DEC  DSP3210S     441C Size=1.9GB (4197520 blocks)
       3: DEC  RZ28  (C) DEC441C Size=1.9GB (4110480 blocks)
       4: DEC  RZ28  (C) DECD41C Size=1.9GB (4110480 blocks)
       5: DEC  DSP3210S     X442 Size=1.9GB (4197520 blocks)
       In-Band Enclosure Services
       0 Swap Events, 0 Enclosure Events
slot 0: Ethernet Controller e0
       MAC Address:        00:a0:98:00:00:01 (Twisted pair)
     slot 5: FDDI Network Adapter f5 (DEC 3002)
       MAC Address:        08:00:2b:b2:39:cb (Twisted Pair)
     slot 6: NVRAM (NetApp NVRAM I)
             Memory Size:        2 MB
             Battery 1 Status:   100% (3.27v)
             Battery 2 Status:   100% (3.27v)
     slot 9: Dual SCSI Host Adapter (NetApp SCSI Adapter I)
             SCSI Host Adapter 9a
       0: DEC  RZ28  (C) DEC435E Size=1.9GB (4110480 blocks)
       1: DEC  RZ28  (C) DEC435E Size=1.9GB (4110480 blocks)
       2: DEC  RZ28  (C) DEC435E Size=1.9GB (4110480 blocks)
       3: DEC  RZ28  (C) DEC435E Size=1.9GB (4110480 blocks)
       4: DEC  DSP3210S     X442 Size=1.9GB (4197520 blocks)
       5: DEC  RZ28B (C) DEC0003 Size=1.9GB (4110480 blocks)
       In-Band Enclosure Services
       0 Swap Events, 0 Enclosure Events
             SCSI Host Adapter 9b
              6: EXABYTE EXB-85058SQANXR10781
       In-Band Enclosure Services
       0 Swap Events, 0 Enclosure Events
```

Figure 12-5. sysconfig -v output on a NetApp F330

```
toaster% sysconfig -v
  NetApp Release 4.3: Wed Sep 17 17:33:46 PDT 1997
  System ID: 0016782403
  slot 0: System Board (NetApp System Board III rev-b)
    Firmware release:   1.6_a
    Memory Size:  256 MB
  slot 2: SCSI Host Adapter 2 (QLogic ISP 1040B)
    Firmware Version 2.26     Clock Rate 60MHz.
    1: QUANTUM DLT7000   172D
    In-Band Enclosure Services
    0 Swap Events, 0 Enclosure Events
  slot 3: Gigabit Ethernet Controller e3
    MAC Address:  00:60:cf:00:01:01 (1000fx)
  slot 4: NVRAM (NetApp NVRAM I)
    Revision:     A8
    Serial Number:      5187
    Memory Size:  8 MB
    Battery 1 Status:   100% (3.30v)
    Battery 2 Status:   100% (3.32v)
  slot 5: FDDI Network Adapter f5 (DEC 3002)
    MAC Address:  00:00:f8:4a:8c:34 (Twisted Pair)
  slot 6: 10BaseT/100BaseTX Quad-Ethernet Controller (Znyx)
    e6a  MAC Address:   00:c0:95:e0:06:fc (Twisted pair)
    e6b  MAC Address:   00:c0:95:e0:06:fd (Twisted pair)
    e6c  MAC Address:   00:c0:95:e0:06:fe (Twisted pair)
    e6d  MAC Address:   00:c0:95:e0:06:ff (Twisted pair)
slot 7: Dual SCSI Host Adapter (NetApp SCSI Adapter I)
    SCSI Host Adapter 7a (Qlogic ISP 1040B)
    Firmware Version 2.26     Clock Rate 40MHz.
    0: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
    1: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
    4: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
    5: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
    6: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
```

Figure 12-6. sysconfig -v output on a NetApp F630

```
  SCSI Host Adapter 7b (Qlogic ISP 1040B)
 Firmware Version 2.26     Clock Rate 40MHz.
 0: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 1: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 2: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 3: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 4: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 5: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 6: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 EMU 3: P/N 500408 S/N 2533758982 Firmware Rev 1.8
 0 Swap Events, 0 Enclosure Events
slot 8: Dual SCSI Host Adapter (NetApp SCSI Adapter I)
 SCSI Host Adapter 8a (Qlogic ISP 1040B)
 Firmware Version 2.26     Clock Rate 40MHz.
 0: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 1: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 2: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 3: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 4: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 5: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 6: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 EMU 0: P/N 500408 S/N 2533754136 Firmware Rev 1.8
 0 Swap Events, 0 Enclosure Events
 SCSI Host Adapter 8b (Qlogic ISP 1040B)
 Firmware Version 2.26     Clock Rate 40MHz.
 0: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 1: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 2: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 3: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 4: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 5: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 6: SEAGATE ST19171W  0023 Size=8.3GB (17783112 blocks)
 EMU 1: P/N 500408 S/N 2533754124 Firmware Rev 1.8
 0 Swap Events, 0 Enclosure Events
```

Figure 12-6. sysconfig -v output on a NetApp F630 (Continued)

Displaying overall filer information using sysconfig

If you enter `sysconfig` without any options, similar but abbreviated information is displayed. When you report a problem to Network Appliance, be sure to provide information displayed by `sysconfig -v`. This information is useful for diagnosing system problems.

Displaying filer statistics

You use the `sysstat` and `uptime` commands to display filer statistics.

The sysstat command

The `sysstat` command displays information about CPU utilization, `file` operations, read and write operations on disks and tapes, and the age of data in the cache buffer. By default, the filer displays statistics every 15 seconds; you can specify the interval (in seconds) at which statistics are displayed. Figure 12-7 shows the `sysstat` output updated every second.

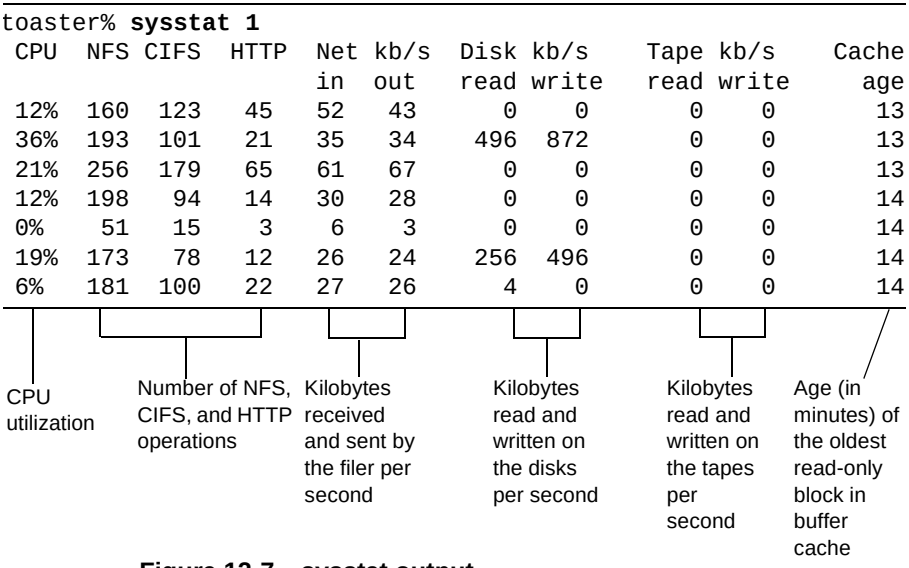


Figure 12-7. sysstat output

The `sysstat` output is particularly useful for revealing file access patterns on your filer, from which you can determine whether you should install more nvram, system memory, or disks.

Depending on the applications that access the filer, the computers on the network, and the network configuration, the following suggested actions may improve your filer's performance:

- Normally, 2 mb of nvram is sufficient for a system with one SCSI channel. However, if the "Disk kb/s write" column shows that the filer writes to disks almost every second, adding nvram may help improve system performance. If there are short spurts of writes, as shown in Figure 12-7, adding nvram will not improve system throughput. See the hardware guide for your filer model for information about how much additional nvram you can install.

- The data in the "Cache age" column indicates how fast read operations are cycling through system memory. If the values in this column are consistently below 5, the filer may benefit from more system memory. Cache age shows the age of the oldest read-only blocks in memory. A low cache age means that the filer is retrieving information from disk instead of from memory. For information about adding memory, see the next section.

- If the CPU utilization percentage in the "cpu" column is low but you are not getting the expected performance, you might need to add disks to the system.

- If CPU utilization is high, you might need to add another filer.

- If the network traffic, shown in the "Net kb/s" column, is at or near the capacity of the network interfaces (1 mb per Ethernet adapter card and 10 mb per FDDI adapter), you might need to add another network adapter card to the system.

The uptime command

The `uptime` command prints the current time, the length of time the system has been up, and the total number of NFS operations the system has performed since it was last booted. A example of the display is:

```
toaster> uptime
8:54am up 2 days 22:23, 3122520 NFS ops
```

Improving filer performance

The following list describes configuration procedures that might improve your filer's performance:

- Limiting directory file size

  An extremely large directory file can use up most of the filer's CPU cycles when a user enters an `ls` command in the directory. The limit on directory file size is set by the `options wafl.maxdirsize` command. The default limit (10 mb) should prevent the system from hanging. A directory of this size accommodates approximately 300,000 files with short file names.

  This option takes the maximum number of kilobytes as its argument. When you reset the maximum directory size, the argument that you supply is rounded up to the next highest 4-kilobyte boundary. If a user tries to create a file that would cause the directory to grow larger than the specified size limit, the user's command fails.

- Balancing NFS traffic on network interfaces

  To balance network traffic among different interfaces, attach multiple interfaces on the filer to the same physical network. For example, if two Ethernet interfaces on the filer named "toaster" are attached to the same network where four clients reside, specify in */etc/fstab* on client1 and client2 that these clients mount from *toaster-0:/home*, and specify in */etc/fstab* on client3 and client4 that these clients mount from *toaster-1:/home*. This scheme can balance the traffic among interfaces if all clients generate about the same amount of traffic.

  The filer always responds to an NFS request by sending its reply on the interface on which the request was received.

  For information about editing */etc/fstab*, see "Editing the client's /etc/fstab file" on page 39.

- Avoiding access time update for inodes

If your applications do not depend on having the correct access time for files, you can disable the update of access time (atime) on an inode when a file is read. To prevent updates, turn on the `no_atime_update` option. Consider turning on this option if your filer has extremely high read traffic (for example, on a news server used by an Internet provider) because it prevents inode updates from contending with reads from other files.



CAUTION: If you are not sure whether your filer should maintain an accurate access time on inodes, leave this option at its default (off) so that the access time is updated.

- Putting the file id in the file handles reported by the filer in "big-endian" format (applicable to NFS only).

  Turning on the `nfs.big_endianize_fileid` option improves performance on directory lookups for clients that use the file id in the file handle as a hash key in certain ways. It is recommended that you enable this option if your NFS clients are mainly running HP-UX or IRIX.

NOTE: If you turn on the `no_atime_update` option, all NFS clients that have mounted directories from the filer will have to unmount and remount them, otherwise they will get "stale file handle" errors on all references to files already opened on the filer until they unmount and remount all directories.

- Enabling minimal read-ahead.

  If the file access patterns of your clients are random (nonsequential), turning on minimal read-ahead might improve performance. By default, the filer uses aggressive read-ahead, which enhances sequential access (more commonly used by UNIX clients and applications). To specify minimal read-ahead, turn on the `minra` option. By default, the option is off and the filer does very aggressive read ahead.

Displaying network statistics using netstat

The `netstat` command displays network-related data in various output formats. Figure 12-8 shows an example of `netstat` output.

```
toaster> netstat
Active TCP connections
Active TCP connections
Local Address Remote Address   Swind Send-Q Rwind Recv-Q State
toaster-0.23  198.95.230.12.   39323    101  8760      0 ESTABLISHED
```

Figure 12-8. netstat output

The `-i` and `-I` options show the state of all network interfaces or one specific interface, respectively. Figure 12-9 shows an example of `netstat -i` output:

```
toaster> netstat -i
Name Mtu  Network     Address   Ipkts Ierrs Opkts Oerrs Collis Queue
e0   1500 266.276.286 toaster-0 0    349660 10801 0     260    0
lo   1536 127         local2    0      5302 0     0     0      0
```

Figure 12-9. netstat -i output

Figure 12-10 shows an example of `netstat -I` output:

```
toaster> netstat -I e0
Name MTU Network   Address   Ipkts Ierrs Opkts Oerrs Collis Queue
e0   1500 192.9.25 toaster-0 12984 0     12848 0     0      0
```

Figure 12-10. netstat -I output

The `netstat -r` command shows the filer's routing table. Figure 12-11 shows an example of `netstat -r` output.

```
toaster> netstat -r
Routing tables
Internet:
Destination      Gateway           Flags  Refs      Use Interface
default          nwo-227           UG        1      138 e0
198.95.227       link#1            UC        0        0 e0
198.95.227.1     link#1            UHL       1       24 e0
nwo-227          8:0:20:79:f9:79   UHL       1        0 e0
198.95.227.255   link#1            UHL       1     3696 e0
```

Figure 12-11. netstat -r output

For information about troubleshooting network problems, see "Networking problems" on page 332. For more information about the `netstat` command, see the *na_netstat*(1) man page.

Chapter 13: **Troubleshooting**

This chapter describes the following types of filer problems:

- restarting a down filer
- nvram problems
- disk problems
- configuration problems
- networking problems
- nfs problems
- filer software problems

Hardware problems are described in the hardware guide for your filer model. See the appropriate hardware guide for information about system LEDs, LCD error messages (if your filer has an LCD), and system startup messages.

Restarting a down filer

The following procedure describes what to do if your filer unexpectedly shuts down:

1. Write down the messages displayed on the console, and the message on the LCD (if your filer has an LCD).

2. If the console is ... Do this ...

Showing the mon> prompt	Enter b (for boot) to reboot the filer.
Showing the ok prompt	Enter boot to reboot the filer.
Not responding	Reset the filer by turning it off, leaving it off for 30 seconds, and turning it back on.

3. If the filer does not boot, follow these steps to boot the filer from the System Boot disk:

   a. Place the System Boot disk into the diskette drive.

      If the console is showing the `mon>` prompt, enter `b` to reboot the filer. If it is showing the `ok` prompt, enter `boot`. If the console doesn't respond, reset the filer by turning it off, leaving it off for 30 seconds, and turning it back on.

      Booting from the diskette takes a few seconds longer than booting from the hard disk.

      Once the filer is booted, it displays the following menu:

```
1) Normal Boot
2) Boot without /etc/rc
3) Change Password
4) Install new file system
5) Maintenance mode boot

Selection (1-5)?
```

4. Type **1** and press Enter.

5. If you cannot boot the filer from the System Boot disk, call Network Appliance Technical Support.

NVRAM problems

This section describes two types of nvram problems.

NVRAM battery failure

This section applies only to the NetApp F220, NetApp F330, and NetApp F540. If the filer detects low batteries on the nvram card, it gives advanced warning and automatically shuts down every 24 hours (the time interval can be changed by the `raidtimeout` option in the `options` command).

If the filer is in degraded mode and the nvram batteries are low, it will fail to boot. Replace the nvram batteries immediately.

Very rarely do nvram battery failures cause data loss. When the filer executes the `halt` command, it flushes the contents of nvram to disk, and then turns off the nvram so that no battery power is used. However, if you turn off a FAServer without using the `halt` command and it remains off for more than two weeks, the batteries may fail and data may be lost. (The other filer models can retain the nvram data for a much longer time.)

Inconsistent NVRAM contents

The filer performs a number of checks to ensure that the nvram contents are consistent. If the contents are inconsistent, it discards the inconsistent contents and dumps core. The file requests received during the last few seconds before the filer shuts down will be lost. This does not cause the file system to become inconsistent, but files written during the last 10 seconds before shutdown might contain old or incorrect data. Also, because the parity of some recently written stripes may be incorrect, the filer must do a parity check on the whole raid array. The parity check (and any correction) is performed on-line; that is, the filer conducts the test in the background while continuing otherwise normal operation.

This kind of data error can happen only when you boot a system after a failure or after you turn off a filer without using the `halt` command.

Disk problems

How disk failures affect the filer depends on whether a hot spare disk is installed.

Disk failure without a hot spare

If a hot spare is not installed, a disk failure causes automatic shutdown after a specified period.

Single disk failure

If a disk fails, the filer continues to run without losing any data but has a somewhat degraded performance. Replace the disk as soon as possible, because a second disk failure could cause the filer to lose the entire file system.

The filer logs a warning message in the */etc/messages* file every hour after a disk has failed. To ensure that you notice the failure, the filer automatically shuts itself off in 24 hours, by default, or in a period set with the `raidtimeout` option. You can restart the filer without fixing the disk, but it will continue to shut itself off periodically until you repair the problem.

You should check the */etc/messages* file once a day for important messages. Checking this file can be automated with a script that periodically searches the file and then alerts you.

On-line rebuild

After you replace a disk drive, the filer detects the new drive when the system boots, starts file service, and rebuilds the missing data in the background with minimum interruption to service.

Double disk failure

If a second disk fails while the filer is operating in degraded mode, or if two disks fail at once, the file system will be damaged and cannot be repaired. You will need to reinstall filer software and restore data from a backup tape.

Disk failure with a hot spare

If a hot spare is installed, when a failure is detected, a hot spare automatically comes on line as a replacement. The missing data is rebuilt on the hot spare in the background, so that the interruption to file service is minimized. The activity is logged in the */etc/messages* file, but there is no automatic shut down. However, if a second drive should fail, there is automatic shutdown in 24 hours as determined by the `raidtimeout` option.

The `sysconfig -r` command, described on page 202, shows which disk has been reserved as the hot spare. In addition to disk failure and hot spare replacement activity, the */etc/messages* file logs any failure in a periodic check of the hot spare. For more information, see the *na_sysconfig*(1) and *na_messages*(5) man pages.

After the filer is finished rebuilding data on the hot spare, replace the failed disk with a new hot spare as soon as possible so that there is always a hot spare available in the system.

Inconsistent disk labels

The filer writes a label on each disk indicating its position in the raid disk array. If the labels become inconsistent, the filer will not boot. This problem occurs only if you mistakenly insert a disk that has a label generated on a different filer without first erasing the disk label.

Inconsistent file system

Normally, the file system never becomes inconsistent. However, an inconsistent file system can be a result of combined disk and nvram failure. Contact Network Appliance Technical Support for help if your file system has become inconsistent.

Configuration problems

Configuration problems usually occur in one of the three configuration files: */etc/rc*, */etc/exports*, or */etc/hosts*. This section describes common configuration problems.

Filer not accessible from the administration host

If you can access the filer from the console but not from the administration host, the filer's */etc/hosts* file may have an ip address that is unreachable. Follow these steps to fix this problem:

1. Log in to the filer from the system console.
2. At the filer prompt, enter the following commands:

```
toaster> ifconfig e0 mediatype type IP_address netmask netmask
toaster> exportfs -i -o root=adminhost_IP_addr /
toaster> nfs on
```

   In the actual command, use the interface name appropriate to your filer model. Also, use the information in the Configuration Information table in the *Quick Reference Guide* for ip*_address*, *netmask*, and *adminhost*_ip*_address*.

3. Mount the root file system from the administration host.
4. Edit the configuration files as described in Chapter 4, "Routine filer administration."

Damaged or missing /etc/rc file

If the */etc/rc* file is accidentally deleted, the filer will automatically run setup the next time it is booted. If the system does not respond to network requests after a bootup, check the console to make sure that the system is not waiting for your input.

If the filer cannot boot from the hard disk because of damaged configuration files, you can boot it from the System Boot diskette. In this case, you must manually initialize the filer and correct the configuration. See “Recovering from configuration file problems” on page 328 for information about booting from diskette.

Booting from diskette

You may need to reboot the filer from the System Boot disk to correct configuration problems, recover from a lost password, or correct certain disk configuration problems.



NOTE: Do not try to boot the filer with a pre-4.0 System Boot disk. The attempt to boot fails because pre-4.0 software cannot recognize system data from a filer running Release 4.0 or later. If you need to run an earlier release on the filer, follow the instructions in the *Read Me First* to downgrade your filer.

Recovering from configuration file problems

The following procedure for recovering configuration errors in */etc/rc* is applicable only if your filer runs NFS but not CIFS.



NOTE: Booting the filer without the */etc/rc* file automatically disables cifs service. You cannot use the following procedure using CIFS.

1. Place the System Boot diskette into the disk drive.
2. Reboot the system using one of these methods:
   - If the console is showing the `mon>` prompt, enter `b` (for boot). If the console is showing the ok prompt, enter `boot`.
   - If the console is unresponsive, reset the filer by turning it off, leaving it off for 30 seconds, and turning it back on again.

Booting from the diskette takes longer than booting from the hard disk. Once the filer is booted, it displays the following menu:

```
1) Normal Boot
2) Boot without /etc/rc
3) Change Password
4) Install new file system
5) Maintenance mode boot

Selection (1-5)?
```

3. Type `2` and press Enter.

4. At the filer prompt, enter the following two commands:

```
toaster> ifconfig e0 mediatype type IP_address netmask netmask
toaster> exportfs -i -o \
toaster> access=adminhost_IP_addr,root=adminhost_IP_addr nfs on
```

5. In the actual commands, use the interface name appropriate to your filer model. Use the information in the Configuration Information table in the *Quick Start Guide* for *server-IP-address*, *server-netmask*, and *adminhost-IP-address*.

6. Mount the root file system from the administration host.

7. Edit the configuration files as described in Chapter 2, "Getting started."

8. Remove the diskette from the disk drive and reboot the filer to test the new configuration files.

Although you can correct network problems using various keyboard commands, Network Appliance recommends that you correct the */etc/rc* file so that it initializes the system correctly. Having a tested and working */etc/rc* file assures that the system can reboot itself if there is a power outage or system software failure.

Lost password

If you have forgotten your filer password, reset the password by using the System Boot disk. To avoid security problems, take care to limit access to the System Boot disk. The following is the procedure for resetting the filer password:

1. Place the System Boot diskette into the disk drive.
2. Reboot the system using one of these methods:
   - If the console is showing the filer prompt (for example, `toaster>`), type **`reboot`**.
   - If the console is showing the `mon>` prompt, enter **`b`** (for boot). If the console is showing the `ok` prompt, enter `boot`.
   - If the console is unresponsive, or if you can't access it because you have lost the password, reset the filer by turning it off, leaving it off for 30 seconds, and turning it back on again.

   Booting from the diskette takes longer than booting from the hard disk. Once the filer is booted, it displays the following menu:

   ```
   1) Normal Boot
   2) Boot without /etc/rc
   3) Change Password
   4) Install new file system
   5) Maintenance mode boot

   Selection (1-5)?
   ```

3. Type **3** and press Enter.
4. When the filer asks you for a new password, enter it at the prompt.

   The system prints the following message:

   ```
   Password Changed
   Hit Return to reboot:
   ```

5. Remove the diskette from the disk drive and reboot the filer by pressing Enter.

Option for installing a new file system

In the menu displayed after you boot the filer from floppy, option 4, "Install new file system," replaces the existing file system with a new one. An example of the prompts displayed when using this option is shown below:

```
WARNING: there appear to be valid labels on the disks!

Zero disks and install a new file system? y
This will erase all the data on the disks, are you sure? y
Zeroing disks: takes approximately 66 minutes

A disk can be reserved and designated as a "hot spare"
which will be used to automatically replace another disk
should a disk fail in your configuration.  Note that the
spare disk will be reserved and will not be used for
your file system.

Reserve a disk for use as a "hot spare"? y
"Hot spare" disk 0 reserved.
```

Maintenance mode boot

In the menu displayed after you boot the filer from floppy, option 5, "Maintenance mode boot," provides commands for troubleshooting hardware or configuration problems. These commands are `disk_list`, `disk_check`, and `disk_erase_label`:

- The `disk_list` command displays detailed device information for each disk on the system, such as its *disk_id*, scsi adapter, scsi id, drive type, and firmware revision level.
- The `disk_check` command checks access to a particular drive, issuing read requests to the drive. The drive will be active for approximately 15 seconds. During this time, you can watch the drive's activity LED to verify that the disk is accessed.

- The `disk_erase_label` command erases a disk label on the specified disk drive. Use the `disk_erase_label` command carefully because once a disk label is erased, raid will treat the disk as an uninitialized disk, no longer recognizing it as a member of a raid array. Typically, this command is used to remove an old raid label on a previously used disk that is now being added to an existing raid group. For example, use this command to erase the disk label when you move a disk from one filer to another.

Before booting the filer in maintenance mode, call Network Appliance for instructions.



NOTE: You must power cycle the filer to exit maintenance mode.

Networking problems

You can detect networking problems by going to the filer console and using the `ping` command, which checks whether the filer can communicate with other hosts on the network and that other hosts can communicate with the filer. If the filer should be able to reach a host but `ping` does not respond with the message indicating the host is alive, follow these steps:

1. Check that the network cable is tightly connected to the proper interface connector.

2. Use the `ifconfig` command to verify that the ip address and netmask are set correctly and the up and running flags are displayed.

3. Use `arp -a` to confirm that the filer has the correct ip-to-Ethernet address map for the host you try to reach.

4. Use the `netstat -i` command to check for excessive errors on the interface(s). If you see excessive input errors (ierrs) or output errors (oerrs), check the network connections on both ends of the connection. Bad transceivers or network hubs can sometimes introduce errors into the network. Collisions reported by `netstat` are a concern only if the filer detects a

substantial percentage of collisions as compared to the total packet throughput. The goal is to keep collisions below 5 percent, but a network can operate properly (but slowly) with collision rates as high as 30 percent.

5. Run `netstat -r` to examine the routing tables.

6. Run `routed status` to find out the status of the default router.



NOTE: An improperly set up network router can also cause network problems. If a router is not working correctly or not configured with the filer's address, clients or hosts on the other side of the router cannot access the filer through that router.

7. If you are using the filer on a cifs network, and you are experiencing difficulty accessing the filer using name-based ip operations (for example, `ping toaster`), create static mappings on your wins servers for each of the filer's interfaces.

If you encounter other problems, contact Network Appliance Technical Support for help.

Trouble using UNIX cpio

If you copy large amounts of data using UNIX `cpio`, some files might be copied incorrectly. This happens in UNIX versions of `cpio` that still use a 16-bit inode definition number. Large file systems require a 32-bit number. The problem generally occurs only on file systems with a large number of hard links.

Some versions of the `cpio` utility work by copying each file with hard links once and then recreating the hard links. Trouble occurs because the inode number is assumed to be 16 bits. If another file has a matching low-order 16 bits, an internal number collision occurs, which `cpio` does not recognize. `cpio` then overwrites the first file and creates files that no longer contain the original data.

Check with your UNIX provider to see if your version of `cpio` has this problem. If you use SunOS 4.*x*, ask for the Sun patch 100556-01. This patch works around the problem by requiring that the files have the same UID, GID, mode, mod time, inumber, and device, before concluding that they are the same file.

False UNIX df file space display

Some UNIX versions of the `df` command have file system limits considerably smaller than the file system size supported by the filer. This could cause the UNIX `df` command to show an incorrect amount of filer disk space in use or remaining. However, the disk space you installed in your filer is fully available and you can use it.

To avoid incorrect display of disk space on a client system, enable the `df_2gb_lim` option, as described in "Configuring filer options" on page 82. For additional information, refer to the *na_options*(1) man page.



NOTE: Some DOS, Windows, or MacOS clients might have a similar display problem; in these cases, enable the `df_2gb_lim` option.

At all times, the `df` command entered on the filer correctly shows the amount of disk space used and remaining.

If a directory in a quota tree is mounted, and a client issues a `df` command on something under that mount point, the command shows the smaller of the client's file system limit or the filer disk space. This makes the quota tree look fuller according to the client `df` command than it actually is. The `quota report` command shows the correct usage within that quota tree.

NFS problems

nfs problems are indicated when the filer and the client can communicate with each other using the `ping` command, and the client can connect to the filer using `telnet`, but the client cannot mount files from the filer. When nfs problems occur, follow these steps:

1. Make sure that the filer can correctly look up the client host name.
2. Make sure that nfs service has been turned on using the `nfs on` command.

Sometimes a client can see the filer but gets a `Permission Denied` message when requesting a mount. If this happens, follow these steps:

1. Make sure that you have defined the file systems correctly in the filer's */etc/exports* file and that you have run the `exportfs` command on the filer.
2. On certain clients the mount request does not come from the root user using a privileged port. The filer denies such mount requests by default to ensure secure access. To grant such mount requests, issue the following `options` command:

```
toaster> options mount_rootonly 0
```

To make this change permanent, add the above command to your */etc/rc* file.

Software problems

If your filer has a serious problem, it will issue a system panic message similar to the one below:

```
PANIC: system hung (NS0)!
```

This message indicates either a problem with the hardware or a bug in the system software. If you get a system panic message, call Network Appliance Technical Support immediately.



NOTE: The panic class of this message is significant; the actual text of the message varies with circumstances. Be sure to write down the actual panic message and send it to Network Appliance Technical Support when reporting a filer problem.

Getting technical help

Before contacting your service provider or Network Appliance Technical Support, make sure that you have the following information available:

- The system serial number.
- The system configuration as reported by `sysconfig -v` (if the filer still responds to the command).
- Any diagnostic messages that have been reported. (Diagnostic messages can be found in the */etc/messages* file.)
- For hardware problems on the FAServer 1300, FAServer 1400, and FAServer 450, the EISA configuration output information and the FAServer configuration test message.

Network Appliance provides telephone, fax, and email support.

Table 13-0. Contacting Network Appliance Technical Support

Telephone	FAX	email
For U.S. customers: 888 463-8277 For international customers: +1 408 367-3000 (ask for Technical Support)	(408) 367-3151	support@netapp.com

When requesting support using fax or email, be sure to include the information requested above in the text of the message, along with a complete description of the problem. Include listings or printouts of the */etc/messages* file, if applicable.

Appendix A: **Time zone names**

Enter one of the time zone names during the setup procedure. Be sure to include slashes if they are part of the name.

Africa

Africa / Abidjan	Africa / Accra	Africa / Addis_Ababa
Africa / Algiers	Africa / Asmera	Africa / Bamako
Africa / Bangui	Africa / Banjul	Africa / Bissau
Africa / Blantyre	Africa / Brazzaville	Africa / Bujumbura
Africa / Cairo	Africa / Casablanca	Africa / Conakry
Africa / Dakar	Africa / Dar_es_Salaam	Africa / Djibouti
Africa / Douala	Africa / Freetown	Africa / Gaborone
Africa / Harare	Africa / Johannesburg	Africa / Kampala
Africa / Khartoum	Africa / Kigali	Africa / Kinshasa
Africa / Lagos	Africa / Libreville	Africa / Lome
Africa / Luanda	Africa / Lumumbashi	Africa / Lusaka
Africa / Malabo	Africa / Maputo	Africa / Maseru
Africa / Mbabane	Africa / Mogadishu	Africa / Monrovia

Africa (Cont.)

Africa / Nairobi	Africa / Ndjamena	Africa / Niamey
Africa / Nouakchott	Africa / Ouagadougou	Africa / Porto-Novo
Africa / Sao_Tome	Africa / Timbuktu	Africa / Tripoli
Africa / Tunis	Africa / Windhoek	

America

America / Adak	America / Anchorage	America / Anguilla
America / Antigua	America / Aruba	America / Asuncion
America / Atka	America / Barbados	America / Belize
America / Bogota	America / Boise	America / Buenos_Aires
America / Caracas	America / Catamarca	America / Cayenne
America / Cayman	America / Chicago	America / Cordoba
America / Costa_Rica	America / Cuiaba	America / Curacao
America / Dawson	America / Dawson_Creek	America / Denver
America / Detroit	America / Dominica	America / Edmonton
America / El_Salvador	America / Ensenada	America / Fort_Wayne
America / Fortaleza	America / Glace_Bay	America / Godthab
America / Goose_Bay	America / Grand_Turk	America / Grenada
America / Guadeloupe	America / Guatemala	America / Guayaquil
America / Guyana	America / Halifax	America / Havana
America / Indiana	America / Indianapolis	America / Inuvik

America (cont.)

America / Iqaluit	America / Jamaica	America / Jujuy
America / Juneau	America / Knox_IN	America / La_Paz
America / Lima	America / Los_Angeles	America / Louisville
America / Maceio	America / Managua	America / Manaus
America / Martinique	America / Mazatlan	America / Mendoza
America / Menominee	America / Mexico_City	America / Miquelon
America / Montevideo	America / Montreal	America / Montserrat
America / Nassau	America / New_York	America / Nipigon
America / Nome	America / Noronha	America / Panama
America / Pangnirtung	America / Paramaribo	America / Phoenix
America / Port_of_Spain	America / Port-au-Prince	America / Porto_Acre
America / Puerto_Rico	America / Rainy_River	America / Rankin_Inlet
America / Regina	America / Rosario	America / Santiago
America / Santo_Domingo	America / Sao_Paulo	America / Scoresbysund
America / Shiprock	America / St_Johns	America / St_Kitts
America / St_Lucia	America / St_Thomas	America / St_Vincent
America / Swift_Current	America / Tegucigalpa	America / Thule
America / Thunder_Bay	America / Tijuana	America / Tortola
America / Vancouver	America / Virgin	America / Whitehorse
America / Winnipeg	America / Yakutat	America / Yellowknife

Antarctica

Antarctica / Casey	Antarctica / DumontDUrville	Antarctica / Mawson
Antarctica / McMurdo	Antarctica / Palmer	Antarctica / South_Pole

Asia

Asia / Aden	Asia / Alma-Ata	Asia / Amman
Asia / Anadyr	Asia / Aqtau	Asia / Aqtobe
Asia / Ashkhabad	Asia / Baghdad	Asia / Bahrain
Asia / Baku	Asia / Bangkok	Asia / Beirut
Asia / Bishkek	Asia / Brunei	Asia / Calcutta
Asia / Chungking	Asia / Colombo	Asia / Dacca
Asia / Damascus	Asia / Dubai	Asia / Dushanbe
Asia / Gaza	Asia / Harbin	Asia / Hong_Kong
Asia / Irkutsk	Asia / Ishigaki	Asia / Istanbul
Asia / Jakarta	Asia / Jayapura	Asia / Jerusalem
Asia / Kabul	Asia / Kamchatka	Asia / Karachi
Asia / Kashgar	Asia / Katmandu	Asia / Krasnoyarsk
Asia / Kuala_Lumpur	Asia / Kuching	Asia / Kuwait
Asia / Macao	Asia / Magadan	Asia / Manila
Asia / Muscat	Asia / Nicosia	Asia / Novosibirsk
Asia / Omsk	Asia / Phnom_Penh	Asia / Pyongyang
Asia / Qatar	Asia / Rangoon	Asia / Riyadh
Asia / Saigon	Asia / Seoul	Asia / Shanghai

Asia (Cont.)

Asia / Singapore	Asia / Taipei	Asia / Tashkent
Asia / Tbilisi	Asia / Tehran	Asia / Tel_Aviv
Asia / Thimbu	Asia / Tokyo	Asia / Ujung_Pandang
Asia / Ulan_Bator	Asia / Urumqi	Asia / Vientiane
Asia / Vladivostok	Asia / Yakutsk	Asia / Yekaterinburg
Asia / Yerevan		

Atlantic

Atlantic / Azores	Atlantic / Bermuda	Atlantic / Canary
Atlantic / Cape_Verde	Atlantic / Faeroe	Atlantic / Jan_Mayen
Atlantic / Madeira	Atlantic / Reykjavik	Atlantic / South_Georgia
Atlantic / St_Helena	Atlantic / Stanley	

Australia

Australia / ACT	Australia / Adelaide	Australia / Brisbane
Australia / Broken_Hill	Australia / Canberra	Australia / Darwin
Australia / Hobart	Australia / LHI	Australia / Lindeman
Australia / Lord Howe	Australia / Melbourne	Australia / NSW
Australia / North	Australia / Perth	Australia / Queensland
Australia / South	Australia / Sydney	Australia / Tasmania
Australia / Victoria	Australia / West	Australia / Yancowinna

Brazil

Brazil / Acre	Brazil / DeNoronha	Brazil / East
Brazil / West		

Canada

Canada / Atlantic	Canada / Central	Canada / East-Saskatchewan
Canada / Eastern	Canada / Mountain	Canada / Newfoundland
Canada / Pacific	Canada / Saskatchewan	Canada / Yukon

Chile

Chile / Continental	Chile / EasterIsland

Etc

Etc/GMT	Etc/GMT+0	Etc/GMT+1
Etc/GMT+2	Etc/GMT+3	Etc/GMT+4
Etc/GMT+5	Etc/GMT+6	Etc/GMT+7
Etc/GMT+8	Etc/GMT+9	Etc/GMT+10
Etc/GMT+11	Etc/GMT+12	Etc/GMT0
Etc/GMT-0	Etc/GMT-1	Etc/GMT-2
Etc/GMT-3	Etc/GMT-4	Etc/GMT-5
Etc/GMT-6	Etc/GMT-7	Etc/GMT-8
Etc/GMT-9	Etc/GMT-10	Etc/GMT-11
Etc/GMT-12	Etc/GMT-13	Etc/GMT-14
Etc/Greenwich	Etc/UCT	Etc/Universal
Etc/UTC	Etc/Zulu	

Europe

Europe / Amsterdam	Europe / Andorra	Europe / Athens
Europe / Belfast	Europe / Belgrade	Europe / Berlin
Europe / Bratislava	Europe / Brussels	Europe / Bucharest
Europe / Budapest	Europe / Chisinau	Europe / Copenhagen
Europe / Dublin	Europe / Gibraltar	Europe / Helsinki
Europe / Istanbul	Europe / Kiev	Europe / Kuybyshev
Europe / Lisbon	Europe / Ljubljana	Europe / London
Europe / Luxembourg	Europe / Madrid	Europe / Malta
Europe / Minsk	Europe / Monaco	Europe / Moscow
Europe / Oslo	Europe / Paris	Europe / Prague
Europe / Riga	Europe / Rome	Europe / San_Marino
Europe / Sarajevo	Europe / Simferopol	Europe / Skopje
Europe / Sofia	Europe / Stockholm	Europe / Tallinn
Europe / Tirane	Europe / Vaduz	Europe / Vatican
Europe / Vienna	Europe / Vilnius	Europe / Warsaw
Europe / Zagreb	Europe / Zurich	

GMT

GMT	GMT+1	GMT+2
GMT+3	GMT+4	GMT+5
GMT+6	GMT+7	GMT+8
GMT+9	GMT+10	GMT+11
GMT+12	GMT+13	GMT-1
GMT-2	GMT-3	GMT-4
GMT-5	GMT-6	GMT-7
GMT-8	GMT-9	GMT-10
GMT-11	GMT-12	

Indian (Indian Ocean)

Indian / Antananarivo	Indian / Chagos	Indian / Christmas
Indian / Cocos	Indian / Comoro	Indian / Kerguelen
Indian / Mahe	Indian / Maldives	Indian / Mauritius
Indian / Mayotte	Indian / Reunion	

Mexico

Mexico / BajaNorte	Mexico / BajaSur	Mexico / General

Miscellaneous

Arctic/Longyearbyen	CET	CST6CDT
Cuba	EET	Egypt
Eire	EST	EST5EDT
Factory	GB	GB-Eire
Greenwich	Hongkong	HST
Iceland	Iran	Israel
Jamaica	Japan	Kwajalein
Libya	MET	MST
MST7MDT	Navajo	NZ
NZ-CHAT	Poland	Portugal
PRC	PST8PDT	ROC
ROK	Singapore	Turkey
UCT	Universal	UTC
WET	W-SU	Zulu

Pacific

Pacific/Apia	Pacific/Auckland	Pacific/Chatham
Pacific/Easter	Pacific/Efate	Pacific/Enderbury
Pacific/Fakaofo	Pacific/Fiji	Pacific/Funafuti
Pacific/Galapagos	Pacific/Gambier	Pacific/Guadalcanal
Pacific/Guam	Pacific/Honolulu	Pacific/Johnston
Pacific/Kiritimati	Pacific/Kosrae	Pacific/Kwajalein
Pacific/Majuro	Pacific/Marquesas	Pacific/Midway
Pacific/Nauru	Pacific/Niue	Pacific/Norfolk
Pacific/Noumea	Pacific/Pago_Pago	Pacific/Palau
Pacific/Pitcairn	Pacific/Ponape	Pacific/Port_Moresby
Pacific/Rarotonga	Pacific/Saipan	Pacific/Samoa
Pacific/Tahiti	Pacific/Tarawa	Pacific/Tongatapu
Pacific/Truk	Pacific/Wake	Pacific/Wallis
Pacific/Yap		

System V

SystemV/AST4	SystemV/AST4ADT	SystemV/CST6
SystemV/CST6CDT	SystemV/EST5	SystemV/EST5EDT
SystemV/HST10	SystemV/MST7	SystemV/MST7MDT
SystemV/PST8	SystemV/PST8PDT	SystemV/YST9
SystemV/YST9YDT		

US (United States)

US/Alaska	US/Aleutian	US/Arizona
US/Central	US/East-Indiana	US/Eastern
US/Hawaii	US/Indiana-Starke	US/Michigan
US/Mountain	US/Pacific	US/Pacific-New
US/Samoa		

Aliases

GMT	Greenwich, UCT, UTC, Universal, Zulu
CET	MET (Middle European Time)
US/Eastern	Jamaica
US/Mountain	Navajo

Appendix B: Configuration Record

For each filer, make a copy of the form on the next page, then fill it out and store it in a safe place. Store the passwords and license codes for each filer in a very secure place.

The fields and their contents are described in the following sections. For additional information, refer to Chapter 2, "Getting started."

Filer information

Field	Contents
Filer host name	The name of the filer
Time zone	A time zone name from the list in Appendix A, "Time zone names." The default is Greenwich Mean Time (GMT).
DNS Domain Name	The name of a DNS domain, like *netapp.com*
NIS Domain Name	The name of an NIS domain, like *netapp.com*
Interface information	The name, IP address, and subnet mask of each interface on the filer.

Other systems information

Field	Contents
Default Gateway	The name and IP address of the default gateway for routing. The filer does not need a default gateway
Administration host	The name of a client that you will use to do such things as: • Send automatic e-mail to Network Appliance Technical Support. • Enter filer commands using `rsh`. • Mount the filer's root directory (/). If you do not enter the administration host name and address, any NFS client has root access to the filer.
DNS server	The name and IP address of DNS servers.

CIFS information

Field	Contents
WINS Server address	The addresses of any WINS servers for the filer.
File permission style	Whether the filer is using PC (Windows) or UNIX permissions.
Authentication mode	Whether you are using a domain controller or the */etc/passwd* file for authentication. If the filer is in a Workgroup, the filer uses the */etc/passwd* file.
Domain name	The name of the filer's domain.
Workgroup name	The name of the filer's workgroup.

Filer	**Filer host name:**			**Time zone:**	
	DNS domain name:			**NIS domain name:**	
	Interface	**IP address**	**Subnet mask**	**Interface**	**IP address**	**Subnet mask**

Other Systems		**Name**	**IP address**
	Default gateway:		
	Administration host:		
	DNS Server:	**N/A**	
	DNS Server:	**N/A**	
	DNS Server:	**N/A**	

CIFS	**WINS Server address:**		
	WINS Server address:		
	WINS Server address:		
	WINS Server address:		
	File permission style:	**PC** ❑	**UNIX** ❑
	Authentication mode:	**Domain controller** ❑	*/etc/passwd* ❑
	Domain name:		
	Workgroup name:		

Appendix C: **Glossary**

ACL	Access control list, which contains the users' or groups' access rights to each share.
adapter card	A SCSI card, network card, hot swap adapter card, serial adapter card, or VGA adapter that plugs into a filer expansion slot.
administration host	A client computer that is used to manage a filer via `telnet` or `rsh`. This is abbreviated as "adminhost" in this guide.
appliance	A device that performs a well-defined function and is easy to install and operate.
ASUS board	One type of system board used in earlier versions of FAServers.
authentication	A security step performed by a domain controller for the filer's domain, or by the filer itself, using its */etc/passwd* file.
autosupport	Filer daemon that triggers e-mail messages from the customer site to Network Appliance or another specified e-mail recipient when there is a potential filer problem.

big-endian	A binary data format for storage and transmission in which the most significant bit (or byte) comes first.
client	Computer that share files on a filer.
console	A terminal that is attached to a filer's serial port and is used to monitor and manage filer operation.
copy-on-write	Technique for creating snapshots without consuming excess disk space. Using this technique, snapshots consume space only as blocks are modified and written to new locations on disk.
degraded mode	The operating mode of a filer when a disk is missing from the RAID array or the batteries on the NVRAM card are low. Replace the missing disk or batteries as soon as possible.
disk ID number	A number assigned by the filer to each disk when it probes the disks at boot time.
disk shelf	A shelf that contains disk drives and is attached to the filer.
EISA	Extended Industry Standard Architecture, which is the bus architecture used in FAServers.
EISA Configuration Utility	Utility located on the EISA system diskette for configuring FAServers.
Ethernet adapter	An Ethernet interface card.
expansion card	A SCSI card, NVRAM card, network card, hot swap card, or console card that plugs into a filer expansion slot.
expansion slot	Slots on the system board in which you insert expansion cards.
FAServer	A Network Appliance filer that uses the EISA (Extended Industry Standard Architecture) bus architecture.

FDDI adapter	An FDDI interface card.
FDDI-fiber	An FDDI adapter that supports a fiber-optic cable.
FDDI-TP	An FDDI adapter that supports a twisted-pair cable.
GID	Group identification number.
group	A group of users defined in the filer's */etc/group* file.
hot spare disk	A disk installed in the filer that can be used to substitute for a failed disk. Before the disk failure, the hot spare disk is not part of the RAID disk array.
hot swap	Adding, removing, or replacing a disk while the filer is running.
hot swap adapter	An expansion card on the FAServer 1400 that makes it possible to add or remove a hard disk with minimal interruption to file system activity.
inode	A data structure containing information about files on a filer and in a UNIX file system. There is an inode for each file and a file is uniquely identified by the file system on which it resides and its inode number on that system.
interrupt switch	A switch on some filer front panels used for debugging purposes. Use of this switch is reserved for authorized service personnel only.
magic directory	A directory that can be accessed by name but does not show up in a directory listing. The *.snapshot* directories, except for the one at the mount point or at the root of the share, are magic directories.

mail host	Client host responsible for sending automatic e-mail to Network Appliance when certain filer events occur.
maintenance mode	An option when booting a filer from floppy disk. Maintenance mode provides special commands for troubleshooting hardware and configuration.
MIME	Multipurpose Internet Mail Extensions, a specification that defines the mechanisms for specifying and describing the format of Internet message bodies. An HTTP response containing the MIME Content-Type header allows the HTTP client to invoke the application that is appropriate for the data received.
NDMP	Network Data Management Protocol. A protocol that allows Network Appliance filers to communicate with backup applications, and provides capabilities for controlling the robotics of multiple tape backup devices.
network adapter	An Ethernet, FDDI, or ATM adapter.
NICE board	One type of system board used in a FAServer.
NVRAM cache	Nonvolatile RAM in the filer, used for logging incoming write data and NFS requests. Improves system performance and prevents loss of data in case of a filer or power failure.
NVRAM card	Adapter card that contains the filer's NVRAM cache.
parity disk	Disk on which parity information is stored for the RAID-4 disk drive array. Used to reconstruct data in failed disk blocks or on a failed disk.

PCI	Peripheral Component Interconnect, which is the bus architecture used in newer filer models.
pcnfsd	Filer daemon that permits PCs to mount filer file systems. The corresponding PC client software is called (PC) NFS.
PDC	Primary Domain Controller. The domain controller that has negotiated to be (or been assigned as) the primary authentication server for the domain.
POST	Power-on self tests run by the filer right after the power is turned on.
PVC	Permanent Virtual Circuit, which is a link with a static route defined in advance, usually by manual setup.
quota tree	A directory on which you can impose tree quotas, created by the `quota qtree` command.
RAID	Redundant Array of Independent Disks. A technique that protects against disk failure by computing parity information based on the contents of all the disks in the array. The filer uses RAID Level 4, which stores all parity information on a single disk.
RAID scrubbing	The process in which the system reads each disk in the RAID group and tries to fix media errors by rewriting the data to another disk area.
SCSI adapter	Expansion card that supports the SCSI devices: disk drives and tape drives.
SCSI ID	The number of a disk drive on the SCSI chain (0-6).

SCSI address	The full address of a disk, consisting of the disk's SCSI adapter number and the disk's SCSI ID (for example, 9a.1).
serial adapter	Expansion card for attaching a terminal as the console on some filer models.
serial console	An ASCII or ANSI terminal attached to a filer's serial port. Used to monitor and manage filer operations.
share	A directory or directory structure on the filer that has been made available to network users and can be mapped to a drive letter on a CIFS client.
snapshot	An on-line, read-only copy of the entire file system that protects against accidental deletions or modifications of files without duplicating file contents. Snapshots allow users to restore files and allow the system administrator to back up the filer to tape while the filer is in use.
SVC	Switched Virtual Circuit. A connection established via signaling. The user defines the end points when the call is initiated.
system board	Board that contains the filer's CPU, expansion bus slots, and system memory.
tree quota	A type of disk quota that restricts the disk usage of a directory created by the `quota qtree` command. Different from user and group quotas that restrict disk usage by files with a given UID or GID.
UID	User identification number.
VCI	Virtual Channel Identifier. A unique numerical tag defined by a 16 bit field in the ATM cell header that identifies a virtual channel over which the cell is to travel.

VGA adapter	Expansion card for attaching a VGA terminal as the console.
VPI	Virtual Path Identifier. An eight-bit field in the ATM cell header that indicates the virtual path over which the cell should be routed.
WAFL	Write Anywhere File Layout. The file system designed for the Network Appliance filer to optimize write performance.
WINS	Windows Internet Name Service
Workgroup	A collection of computers running Windows NT and/or Windows for Workgroups that are grouped for browsing and sharing.

Index

Numerics

8.3 format for file names 57

A

access control list (ACL)
    assigning rights to users 35, 170
    being checked before UNIX permissions 163
    removing an entry from 170
access rights, assigning for CIFS shares 35, 170
administration host
    accessibility problems from 326
    in configuration setup 15
    modifying configuration files 68
    requirements for 62, 63
Africa, time zone names for 339, 340
arp command 332
ATM
    network mask for 73
    PVCs
        incoming on the filer 128
        outgoing, establishing on the filer 127
        tearing down 130
Australia, time zone names for 343
authentication features, CIFS 19
authentication mode, CIFS, specifying 26
AuthName directive 185
automatic e-mail to Network Appliance
    configuring 98–100
    contents of the messages 97
    designating the contact person 76
    using administration host for 62
automounting 39
autosupport daemon
    configuring with the options command 98–100
    how it works 96
autosupport options 84–85
autosupport.doit option 84, 100
autosupport.enable option 84, 98
autosupport.from option 84, 99
autosupport.mailhost option 84, 99
autosupport.noteto option 84
autosupport.to option 85, 99

B

background rebuild 3
backing up the filer 273–283
base host names 39

block size for the restore command 288
blocking factor for dump command 274
boot server 5
booting
and NVRAM 4
/etc/rc script 72–75
from floppy disk 328–331
halt and reboot commands 66
time required for 2
without /etc/rc 329, 330
booting filer from boot disk 44
broadcast filer address, setting using ifconfig 124

C

C$ share 63, 68
cables, checking 332
Canada, time zone names for 344
CGI requests, redirecting 193
CIFS
access rights, files 171
adding clients 150
assigning share-level access rights 170
authentication
domain controller 32
authentication features 19
authentication mode, specifying 26
backups 271
C$ share 63, 68
choosing file access rights 163–166
connecting to filer from clients 40
creating shares 155
creating users' home directories 154
deleting a share 158
determining file access rights method 22
displaying session information 176
displaying share information 157
displaying statistics 175
domain, adding filer 20
effect of filer reboot on clients 66
/etc/group, modifying when authenticating with a domain controller 35
/etc/passwd, modifying
if authenticating with a domain controller 32
if authenticating with /etc/passwd 33
file access rights method, choosing 22–24
file access rights, choosing 163–166
file ownership, changing 169
file-naming conversions and NFS 59
filer name
changing 28
specifying 25
filer NT domain
changing 28
generic account 153
guest access 152, 153
guest account 152
home directories for users, creating 154

home directory share offer 158–161
NT domain name for filer 26
open files, limits 162
oplocks 172–173
options 85–87
password, root 24
removing an ACL entry 170
restoring files 285
rsh use to enter filer commands 151
session information, displaying 176
sessions, limits 162
setup 18–24
shares
assigning access rights 35
changing description 156
changing UNIX-style file permissions 166–167
creating 155
deleting 158
limits 162
managing 155–158
statistics, displaying 175
stopping and restarting service 177
symbolic links 50–53
workgroup
filer in 20
filer setup 27
cifs access command 170
cifs chmod command 166
cifs chown command 169
cifs mkdir command 154
cifs restart command 179
cifs rights command 171
cifs sessions command 176
cifs setup command 21
cifs shares command 155
cifs stat command 175
cifs terminate command 177
cifs.generic_account option 85
cifs.guest_account option 85
cifs.home_dir option 86
cifs.oplocks.enable option 86
cifs.scopeid option 86
cifs.symlinks.cycleguard option 53, 87
cifs.symlinks.enable option 51, 87
clients
accessing snapshots from 265–268
NFS statistics in the custom MIB 110
per-client NFS statistics 144–147
collisions 332
commands
aborted by Ctrl-C 65
accessible via rsh 65
in /etc/rc 73
configuration files
default 68–75
editing 39–40, 63
errors in 327–329
/etc/dgateways 119, 121
/etc/exports 132–134
/etc/hosts 112–114
/etc/hosts.equiv 64
/etc/netgroup 134
/etc/rc 72–75
/etc/resolv.conf 115
/etc/syslog.conf 80–82
configuration problems
booting with floppy disk for 327–329
filer accessibility 326

lost passwords 330
with /etc/rc 326–329
console session 64
copy-on-write technique 245, 253
core files 79
cpio
copying files with 334
nfs-only backups 273
crash files 74, 79
cron and crontab 101–102

D

data rebuild, on the hot spare 228, 325
date command 101
default configuration 68
default gateway 14
default route in routing table 119
default router 73, 119
degraded mode
meaning of 215
timeout period for automatic shutdown 228
when a hot spare is available 228
when a hot spare is not available 227
df command 242–244, 254–257, 334
df_2gb_lim option 89, 334
diagnostic messages 2, 80–82, 338
directories
created by snapshots 266
exporting 74, 132
Directory directive 184
disk labels, erasing 211
diskettes, safe storage of 44
disks
adding 205, 207–208
capacity, increasing 220–225
data, reducing number of 226
degraded mode 200, 215, 227–228
different sizes in same filer 207
different types in RAID 200
failures, effects of 227–228
free space, accessing information through SNMP 110
free space, displaying 242–244
freeing up space by deleting snapshots 262
hot spare 208–213
hot swapping 214–219
ID number 202
inconsistent labels 325
information in the custom MIB 110
locating by identifier numbers 202
maximum number of files 242
moving to another filer 220
new, installing 209, 210
numbering 201–202
parity disk 200, 207–208, 243
problems 324–325
quotas 103, 110, 232–241, 269
raid command 213, 215–218
RAID disk number 201, 202
RAID reconstruction 229
RAID scrubbing option 229
removing 214–217
replacing 204, 205, 207
required to reboot an FAServer 66
restricting disk usage 232–241
right-sizing 204–206
SCSI ID number 201, 202

snapshots, disk space used 253–264
sysconfig command 202
usable space on 204–206
DNS
disabling 116
domain name
specifying during setup 16
specifying with option 88
domain name in /etc/rc 74
enabled in /etc/rc 74
enabling 115
options 88
querying the name server 114
resolving names with 111
specifying during setup 16
dns.domainname option 88
dns.enable option 88
domain, NT, changing 28
double disk failures 325
dump
blocking factor 280
effect of hot swap on 218
/etc/dumpdates 280, 281, 283
levels 280
multiple subtrees 277, 283
number of tapes requires 275
size of file 274
subtrees 278
using filer tape drive from another system
SVR4 UNIX 294
UNIX (not SVR4) 295
using snapshot 245
volume size, maximum 281
dump command 273–283

E

editing configuration files 39, 63
email support 338
Enabling 115
erasing disk labels 211
errors caused by exceeding disk quotas 241
errors displayed by netstat 332
/etc/.cifsconfig.dat 27
/etc/crash directory 74, 79
/etc/dgateways 119, 121
/etc/dumpdates 280, 281, 283
/etc/exports
contents of 63, 132–134
deleting entries 134
exporting directories in 74, 132–134
/etc/fstab
balancing traffic among interfaces 316
editing 39–316
/etc/group 31
/etc/group, modifying for CIFS
when authenticating with a domain controller 32
when authenticating with /etc/passwd 35
/etc/hosts
accessing problems in 326
contents of 112
for IP address 73
on the client 39
/etc/hosts.equiv 64
/etc/httpd.access file 184
/etc/httpd.group file 185
/etc/httpd.hostprefixes file 188
/etc/httpd.passwd file 185
/etc/man directory 42
/etc/messages

activity on the hot spare 228
configuring messages written to the file 80–82
files saved 82
for disk failures 212, 324
messages from savecore 79
reporting to Network Appliance 338
warnings for disk failures 227
/etc/netgroup 134–135
/etc/nsswitch.conf 69, 111
/etc/passwd 30, 31, 33
modifying for CIFS
when authenticating with a domain controller 32
copying for (PC) NFS 139
modifying for CIFS
when authenticating with /etc/passwd 33
/etc/quotas, configuring 232–236
/etc/rc
commands in 72–74
damaged or missing 326–329
updating 72, 75
/etc/resolv.conf 115
/etc/serialnum 70
/etc/shadow 70
/etc/shadow, when using (PC) NFS 139
/etc/symlink.translations file 52
/etc/syslog.conf 80–82
/etc/zoneinfo directory 74
Ethernet
configuring the network mask during setup 13
interface names and numbers 77
network mask for 73
setting the media type 13, 124
Europe, time zone names for 346
explicit routes in routing table 119
exportfs command 74, 134
exporting directories 132–134

F

FAServer 1300, rebooting 66
FAServer 1400, rebooting 66
FAServer 450, rebooting 66
FAX support 338
FDDI
configuring the network mask during setup 13
interface names and numbers 77
network mask for 73
setting the MTU 125
file locking, differences between NFS and CIFS 49
file names
conventions 59
conversion 56
legal characters 57
maximum length 56
file names used by NFS and CIFS clients 57
file space, incorrect display 334
file system
damaged 285, 331
exporting 132–134
inconsistent 325
installing a new file system 220–225, 331
maximum number of files 242
mounting 39–40, 336
restoring the entire 291
filer login and logout 64
filer name
specifying 25

filer password
changing 65
configuring during setup 17
files
access rights, CIFS 171
copying with cpio 334
ownership
changeable by root only 103
changing, CIFS 169
same file criteria 246
setting maximum number of 242
firewall, virtual 187

G

gateway, default 14
Generic account, CIFS 153
GMT, time zone names for 346
group quotas, creating 239
guest access, CIFS 152, 153
Guest account, CIFS 152

H

halt command 4, 323
help command 43
home directories for users, creating in CIFS 154
/home directory
exporting 133
read and write permissions 133
host name in configuration setup 12
host names
administration host 15
for network interfaces 39, 77
resolving 112–118
hostname command 73
hot spare 3, 200, 207, 208–213
hot swapping a disk 3, 214–219
hourly snapshots 250
HTML on-line help files 42
HTTP
configuring the filer for 37
displaying connection information 196
displaying statistics 197
options 88–89
password protection 184–185
root directory 183
starting service 182
virtual hosting, enabling 188
httpd.admin.enable option 88
httpd.enable option 88
httpd.log.max_file_size option 88
httpd.rootdir option 89
httpd.timeout option 89
httpd.timewait.enable option 89

I

ICMP redirect messages 119
IERRS (input errors) displayed by netstat 332
ifconfig command 72, 73, 113, 124–126, 332
inconsistent file system 325
inodes, effects of maximum number of files on 242
interfaces
errors on 332
how packets are sent and received 122
using ifconfig to configure 124–126
See also ATM, Ethernet, FDDI
IP addresses
administration host 15
setting for the filer using setup 13

setting in /etc/rc file 73
setting using ifconfig 124
ip.path_mtu_discovery.enable option 93

L

label, disk 325
legal characters in file names 57
license command 104, 106
licenses 104–106
logging in to the filer 64
logging out from the filer 64
lost data from disk failures 324
lost passwords 330
ls command, listing snapshot files 266, 268

M

mail host
automatic e-mail 62, 96
designating other mail hosts 99
Makefile, NIS 114, 135
man pages for the filer 42
MANPATH environment variable 42
maxfiles command 242
media type for an Ethernet interface 13, 38, 124
messages
configuring the syslogd daemon 80–82
file of 80–82
for disk failures 324
panic 337
severity of 80
MIB, Network Appliance Custom 44, 109
MIB-II 108
MIME Content-Type, specifying 190
minra option 93, 317
mount_rootonly option 94, 103
mounting files
if there are quota trees 244
problems with 336
procedure for 39–40
restricting mount privilege to root 103
mt command 299

N

name server, adding filer names and IP addresses 39
name services, specifying the order in which contacted 111
names
filer
CIFS name, changing 28
CIFS name, specifying 25
host names 12
resolving 111–118
NDMP 296–298
ndmp command 297
NetBIOS scope IDs 174
netstat command 119, 122, 318, 332
Network Appliance Technical Support e-mail address 96
network interfaces
balancing traffic among 316
configuring 124–126
naming conventions 77
networks
connections, checking 332
connections, verifying with ping 38

displaying statistics (netstat command) 318
how filer sends and receives traffic 122
management services, using SNMP 108–110
network mask
configuring during setup 13
configuring using ifconfig 124
in /etc/rc file 73
problems with 332
protocols supported 4
statistics 332
using ifconfig to configure 124
New Zealand, time zone names for 343
NFS
displaying per-client statistics using the nfsstat command) 144
file-naming conversions and CIFS 59
how interfaces respond to packets 122
problems with 336
statistics in custom MIB 109
statistics, displaying (nfsstat command) 143
turning on 75
NFS and (PC) NFS options 89–91
nfs command 75
NFS over UDP requests 122
nfs.big_endianize_fileid option 90, 317
nfs.per_client_stats.enable option 90, 144
nfs.tcp.enable option 90
nfs.v3.enable option 90
nfs.webnfs.enable option 90, 141
nfs.webnfs.rootdir option 91
options nfs.webnfs.rootdir 142
nfs.webnfs.rootdirset option 91, 142
nfsstat command 143–146
nightly snapshots 248–250
NIS
changing domain name 118
disabling 118
domain name
specifying during setup 16
specifying with option 91
domain name set in /etc/rc 74
enabled in /etc/rc 74
enabling
during setup 16, 117
without using setup 118
maps supported 117
options 91–92
propagating changes
/etc/hosts on filer 114
/etc/netgroup on filer 135
/etc/passwd on filer 139
/etc/shadow for (PC) NFS on filer 139
nis.domainname option 91
nis.enable option 92
no_atime_update option 94, 317
Non-Volatile RAM (NVRAM)
cache 4
failures in 323
inconsistent NVRAM contents 323
NVRAM batteries 66, 228, 323
when to increase 314
nosnap option 92, 269
nosnapdir option 93, 269
NT domain

filer name 26
filer, changing 28
numbering network interfaces 77

O

on-line help, viewing 42
oplocks 172–173
options
autosupport.doit 84, 100
autosupport.enable 84, 98
autosupport.from 84, 99
autosupport.mailhost 84, 99
autosupport.noteto 84
autosupport.to 85, 99
cifs.generic_account 85
cifs.guest_account 85
cifs.home_dir 86
cifs.oplocks.enable 86
cifs.scopeid 86
cifs.symliks.enable 87
cifs.symlinks.cycleguard 53
cifs.symlinks.enable 51
df_2gb_lim 89, 334
dns.domainname 88
dns.enable 88
httpd.admin.enable 88
httpd.enable 88
httpd.log.max_file_size 88
httpd.rootdir 89
httpd.timeout 89
httpd.timewait.enable 89
ip.path_mtu_discovery 93
minra 93, 317
mount_rootonly 94, 103
nfs.big_endianize_fileid 90, 317
nfs.per_client_stats.enable 90, 144
nfs.tcp.enable 90
nfs.v3.enable 90
nfs.webnfs.enable 90, 141
nfs.webnfs.rootdir 91
nfs.webnfs.rootdirset 91, 142
nis.domainname 91
nis.enable 92
no_atime_update 94, 317
nosnap 92, 269
nosnapdir 93, 269
pcnfsd.enable 91, 138
pcnfsd.umask 91
raid.reconstruct_speed 92
raid.scrub.enable 92, 230
raidtimeout 92, 200
root_only_chown 94, 103
rps.status 94
telnet.hosts 95, 103
wafl.maxdirsize 95, 316
options cifs.symlinks.cycleguard 87
options command
making changes permanent 82
options to 82
See also options

P

packets, responses to 122
panic files 79
panic message 337
parity disk 3, 200, 207–208
partitions 2
passwd command 65
password
changing with passwd command 65
CIFS root 24
configuring during setup 17
HTTP pages 184–185
lost 330

used for shell session 64
(PC) NFS 138–139
pcnfsd daemon 138–139
pcnfsd.enable option 91, 138
pcnfsd.umask option 91
performance, improving 314
permissions
for snapshots 3, 246
on exported directories 133
ping command 38, 332
power
turning off 66
turning on 9
protocol licenses 104–106
protocols, network 4
PVCs, ATM
establishing on a remote ATM host 128, 130
establishing on interconnecting ATM switches 129
incoming, establishing on the filer 128
outgoing, establishing on the filer 127

Q

quota command 236
quota tree, creating 240
quotas, disk
effects of snapshots on 269
information available through SNMP 110
information available through the custom MIB 110
procedures for using 232–241
when changing file ownership 103

R

RAID (Redundant Array of Independent Disks)
accessing information through SNMP 110
and disk labels 325
and on-line rebuild 3
definition 3
disk number 201
displaying free space in 243
displaying information about disks 304
information in the custom MIB 110
options 92
reconstruction speed 229
scrubbing 229
raid command 213–217, 230
raid.reconstruct_speed option 92
raid.scrub.enable option 92, 230
raidtimeout option 92, 200
rdate command 101
read ahead, minimal 317
read-only bit 54–55
reboot command 4, 66
rebooting
and NVRAM 4
automatic e-mail to Network Appliance 79
downed filer 322
FAServers 66
from floppy disk 328–332
halt and reboot commands 66
scripts for 72
to update configuration 75
reconstruction, disk 229
remote shell (rsh)
accessing the filer from the administration host 64

for entering filer commands 65
requests, replies to 122
require group directive 185
require user directive 185
resolving host names 112
restore command 285–293
restore_symboltable 291
restoring files, remote access to a filer's tape drive
SVR4 UNIX 295
UNIX (not SVR4) 295
root password, CIFS 24
root_only_chown option 94, 103
route command 119
routed command 120
routed daemon
enabled by default 15
/etc/dgateways file 121
purpose of 120
router
default 73, 119
effects on ping command 38
problems with 333
Routing Information Protocol (RIP) 120
routing table, filer 119
rps.status option 94
rsh. See remote shell (rsh)

S

savecore command 74, 79
security
mounting /home 134
options for 103
permissions on /home 133
system password 65
setting up the filer 10–38, 68
setup 10–29, 72
setup command 18
setup program 10
shadow file 70
shares
assigning access rights 35
managing 155–158
single disk failures 324
size of dump file 274
snap command 247–251
snapshot_for_dump snapshot 278
snapshot_for_dump.n snapshot 278
snapshots
accessing 265–266
automatic 74, 248–250
commands for 247
definition 3, 245
deleting to free up space 262
directory name displayed on CIFS clients 267
disk consumption by 254–264
effects on disk quotas 269
information in the custom MIB 110
ls command 266, 268
magic directories 266
making snapshot directory invisible 269
operation of 252–253
options 92–93
reserving space for 255–257, 259–260, 262
scheduling 257
temporarily disabling 269
types 248
SNMP
commands
examples of 108–109
in /etc/rc 75
configuring the agent 108

custom MIB 109
snmp command 108–109
software licenses 104–106
South America, time zones for 344
sticky bit 70
subnets, exporting to 136–137
SVTX bit 70
symbolic links, CIFS 50–53
sysconfig command 202, 205, 304–309, 338
syslogd 80
sysstat command 313–314
system boot diskette 44, 322, 328
system crashes 74, 79
system diskettes, purposes of 44
System files floppy disk 44
system message files 80–82
system panics 79, 337
system software, reinstalling 44

T

tape drives
controlling 299–301
displaying information about 306
remote access to filer's tape drive 294
tapes, number required by dump 275
tar and nfs-only backups 273
TCP/IP, how interfaces respond to packets 123
telnet connection to the filer
administration host 62, 63
limiting host access 95, 103
telnet.hosts option 95, 103
time
setting on the filer 101
synchronizing the filer with another system 101
time zones
configuring during setup 16
names of 339–350
setting 74
timezone command 74
troubleshooting
configuration problems 326–331
downed filer 322
help for 338
network problems 332
NFS problems 336
software problems 337
trusted host 65

U

United States, time zone names for 16, 350
URL, how filer translates 192
user 154
user quotas, creating 238

V

virtual firewall 187
virtual host addresses, mapping 189
virtual hosting, enabling 188

W

wafl.maxdirsize option 95, 316
Web NFS 141–142
weekly snapshots 248–253
WINS server addresses 22
workgroup, CIFS, filer setup 27

Y

ypwhich comand 118