Data ONTAP™ 5.2
System Administrator'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 210-02166
September 1998

Copyright information

Copyright © 1998 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 are derived from the Berkeley Net2 release and the 4.4-Lite-2 release, which are copyrighted and publicly distributed by The Regents of the University of California.

Copyright © 1980–1995 The Regents of the University of California. All rights reserved.

Portions of this product are derived from 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.

Software derived from copyrighted material of The Regents of the University of California, Carnegie Mellon University, and Network Appliance are subject to the following license and disclaimer:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notices, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

   This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER

IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

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).

Trademark information

Network Appliance, NetCache, Data ONTAP, SmartSAN, SnapCopy, Snapshot, WAFL, Web Filer, FilerView, BareMetal, and SecureShare are trademarks of Network Appliance, Inc. FAServer, NetApp, and the Network Appliance logo are registered trademarks of Network Appliance, Inc.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

Table of Contents

Preface xiii

Chapter 1 **Introducing Network Appliance Filers** 1

Filer protocols and features 2

File transfer protocols. 3

Administration protocols 4

Disk and file system management features 6

RAID support 7

Single and multiple file system (volume) support 8

Hot spare and hot swap disk support 9

Quota support 10

Saving and restoring filer data 11

Using FilerView to administer the filer. 12

Chapter 2 **Routine Filer Administration**. 13

Using the administration host. 14

The root volume 16

Editing configuration files 17

Obtaining access to the filer shell 20

Halting and rebooting the filer 22

Understanding the filer default configuration 24

Default exported and shared directories 25

The /etc/rc file 29

Naming conventions for network interfaces 34

Recovering from configuration errors 36

Core files 37

Message logging 39

Configuring filer options 42

The options command 43
The vol options command 44
Detailed option information 45
Autosupport options 46
CIFS options 48
Cluster failover options 51
DNS options 52
HTTP options 53
NFS and (PC)NFS options 55
NIS options 57
RAID options 58
Volume options 59
Miscellaneous options 61
Sending automatic email to Network Appliance 63
Synchronizing filer system time 68
Using options command options to maintain filer security. 70
Software licenses 71

Chapter 3 **Managing Disks and File Systems** 75
Disk concepts 76
Volume concepts 83
Procedures for managing disks and volumes 87
Disk management tasks 88
Volume management tasks 91

Chapter 4 **Network Administration** 95
Working with large files 96
Using SNMP 97
Host name resolution 100
Routing 107

Using ifconfig to configure an interface 111
Using vif to configure a virtual interface 116
ATM PVCs 118

Chapter 5 **Sharing Files between NFS and CIFS Users** 121
File-locking interactions 122
Managing symbolic links for CIFS access 123
NFS and CIFS use of the read-only bit 126
Naming files used by both NFS and CIFS 128
Timing of directory conversion for CIFS access to NFS-created files 130

Chapter 6 **NFS Administration** 133
NFS administration covers four areas 133
Introducing the /etc/exports file 135
Rules for exporting volumes and directories 137
Default /etc/exports entries 140
Restricting access to volumes and directories 141
The exportfs command 143
The /etc/netgroup file 145
Exporting to subnets 147
About configuring a filer for (PC)NFS 149
Introducing the (PC)NFS daemon 150
Enabling the daemon 151
Setting up (PC)NFS user entries 152
Setting the default umask 154
About configuring a filer for WebNFS 156
Setting up WebNFS 157
Managing WebNFS 158
About displaying NFS statistics 160

Chapter 7 **CIFS Administration** 165
CIFS limitations 168
Limits on CIFS open files, sessions, and shares 169
Changing or viewing the filer's description 170
Adding users to the filer 172
Adding local groups to the filer. 174
Using CIFS commands with a remote shell program 177
Enabling guest access. 178
Displaying a filer's shares 180
Creating and changing a share 183
Displaying information about shares 189
Deleting a share. 192
Creating a home share for each user 194
Assigning and changing access rights 197
Displaying access rights to an NTFS file. 202
Viewing and changing UNIX permissions from Windows 203
Sending a message to all users on a filer 206
Using oplocks 207
Displaying CIFS statistics 209
Displaying CIFS session information 210
Stopping and restarting CIFS sessions 212
Reconfiguring the filer for CIFS 216

Chapter 8 **HTTP Administration**. 217
Chapter contents 217
Starting HTTP service 218
Protecting Web pages with passwords 220
Using the HTTP virtual firewall 223
Using virtual hosting 224
Specifying MIME Content-Type values 226

Translating URLs 228
Displaying HTTP connection information 231
Displaying HTTP statistics 232

Chapter 9 **Cluster Administration** 233
Introducing clusters 233
Understanding cluster concepts 235
Understanding takeover 239
Understanding giveback 242
Managing a cluster 244
Managing the cluster in normal mode 245
Managing takeover 247
Managing filers in takeover mode 249
Managing a virtual filer 251
Performing a giveback 256

Chapter 10 **Snapshots** 259
Understanding snapshots 260
Snapshot commands and options 264
Understanding snapshot disk consumption 269
Managing snapshot disk consumption 273
Accessing snapshots from clients 278

Chapter 11 **Administering Qtrees** 281
Using qtrees 283
Qtree security styles 285
Qtree file access models 288
Creating a qtree 290
Modifying the security style of a qtree 291
Modifying qtree oplocks settings 292

Displaying qtree information 294

Chapter 12 **Managing Quotas and Maximum Number of Files** 295

Restricting disk usage by using disk quotas 296

Increasing the maximum number of files. 307

The df command 308

Chapter 13 **Backing Up and Restoring Files** 311

Dump and restore limitations. 314

Methods for backing up a filer 315

Introduction to filer backup. 317

The dump command syntax and options 319

Examples of the dump command. 322

Backing up from the filer. 326

Determining the number of tape volumes required. 328

How the dump command uses snapshots. 330

Backing up from a remote host. 332

Backing up the filer using NDMP 333

Methods for restoring files 337

The restore command syntax and options 339

Examples of the restore command 342

Restoring from the filer. 348

Restoring from a remote host. 349

Using a filer tape drive from another system. 350

Controlling tape devices 352

Chapter 14 **Copying One Volume to Another** 355

Introduction to the filer's commands for copying volumes 356

Requirements and recommendation for copying a volume. 358

Procedure for copying one volume to another 361

Managing a volume copy operation when it is in progress 366

Chapter 15 **System Information and Performance** 369
Displaying the Data ONTAP version 370
Displaying filer configuration information 371
Displaying volume information 373
Displaying filer statistics 374
Displaying network statistics 376
Displaying interface statistics 377
Improving filer performance 387

Chapter 16 **Troubleshooting** 389
Getting technical assistance 391
Booting from system boot diskette 392
Restarting a shut down filer 395
NVRAM problems 396
Volume problems 398
Failed mounts and stale file handles 399
Volume name problems 400
Disk problems 401
Disk failure without a hot spare disk 402
Disk failure with a hot spare disk 403
Disk errors 404
Inconsistent file system 406
Disk operations in Maintenance mode 407
Configuration problems 408
How to reset the filer password 410
How to initialize all disks and create a new file system 411
Network problems 412
NFS problems 414

Windows access problems 416
UNIX cpio problems 420
UNIX df problems 421
Cluster Messages 422
Serious error messages 428

Glossary 429

Index 437

Preface

Introduction

This guide describes how to configure, operate, and manage Network Appliance™ filers that run Data ONTAP™ Release 5.1 software. It covers all filer models.

How to Use This Guide

The following sections describe how to use this guide.

If you are...	Read...
Installing a filer for the first time	The *Getting Started Guide* for concise instructions about setting up a new filer with the default configuration. The information in the *Getting Started Guide* is sufficient for you to prepare the filer for delivering file-sharing service.
Upgrading a filer	The *Upgrade Guide*.
Looking for on-line information about commands	The man pages in the /etc/man directory of the filer default volume. The HTML version of the man pages by using a browser to access *http://filername/na_admin* and selecting the *manual pages* link (replace *filername* with the host name of your filer).

Audience

This guide is 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 are familiar with how the NFS, CIFS, and HTTP protocols are used for file sharing or transfers. This guide 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 might differ, depending on your version of UNIX.

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 pressing the Enter key.

Keyboard conventions

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.

Typographic conventions

The following table describes typographic conventions used in this guide.

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.

Convention	Type of information
`Bold monospaced` font	Words or characters you type. What you type is always shown in lowercase letters, unless you must type it in uppercase letters.

Special messages

This guide contains special messages that 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.

Introducing Network Appliance Filers 1

About Network Appliance filers

The Network Appliance filer is a dedicated, special-purpose data server that provides file service to network clients in the UNIX and Windows environments.

The filer runs the Data ONTAP software, which controls all aspects of file service. The instructions in this guide tell you how to manage the filer using the Data ONTAP command set.

Installation and upgrade instructions

If you are installing a new filer or upgrading an existing filer, use the following table to locate the installation instructions. Otherwise, continue to the next section to learn about the features of the filer and the Data ONTAP software.

If you need to...	**Read...**
Install the filer hardware	The hardware guide that came with the filer.
Configure a new filer	The *First Time Configuration Guide*.
Upgrade an existing filer to the Data ONTAP 5.2 software from a UNIX or Windows client	The *Software and Firmware Upgrade Guide*.

Filer protocols and features

Introduction

The filer supports the protocols and features described in the following table.

Read...	To learn about...
“File transfer protocols” on page 3	File transfer protocols
“Administration protocols” on page 4	Administration protocols
“Disk and file system management features” on page 6	Disk and file management
“Using FilerView to administer the filer” on page 12	Administering a filer using a graphical interface

File transfer protocols

File transfer protocols the filer supports

The filer supports the following file transfer protocols:

- Network File Service (NFS)
- Common Internet File System (CIFS)
- Hypertext Transfer Protocol (HTTP)

NFS protocol

The NFS protocol enables UNIX and (PC)NFS clients to mount filer file systems to local mount points.

Client users can read files from and write files to directories on the filer to which they have been granted access.

CIFS protocol

The CIFS protocol enables Windows 95, Windows NT, and Windows for Workgroups computers to view filer files and directories that have been designated for sharing through the CIFS protocol.

Users of Windows 95, Windows NT, and Windows for Workgroups computers who have been granted access to shared directories can read files from and write files to those directories.

HTTP protocol

The HTTP protocol enables users of World Wide Web (Web) browsers to display files that are stored on the filer.

Administration protocols

Administration protocols the filer supports

The filer supports the following administration protocols:

- Simple Network Management Protocol (SNMP)
- Domain Name Service (DNS)
- Network Information System (NIS)
- Windows Internet Name Service (WINS)
- Routing Information Protocol (RIP)
- Remote Magnetic Tape protocol (RMT)
- telnet protocol
- Remote Shell protocol (rsh)

SNMP protocol

The SNMP protocol enables network management from a centralized administration host.

DNS protocol

The DNS protocol enables the filer to query DNS servers for IP addresses that correspond to host names.

NIS protocol

The NIS protocol enables the filer to query NIS servers to authenticate user names, passwords, group names, and netgroup access permissions.

WINS protocol

The WINS protocol enables the filer to query WINS servers to resolve CIFS names.

RIP protocol

The RIP protocol enables the filer to query RIP servers to determine the default gateway for the network.

RMT protocol

The RMT protocol enables remote hosts to remotely access the filer tape drive.

telnet protocol

The `telnet` protocol enables administrators to log in to the filer from remote hosts using telnet client software.

After administrators log in to the filer, they can monitor and manage the filer from a remote host.

rsh protocol

The `rsh` protocol enables administrators to send commands from designated remote hosts to the filer.

Disk and file system management features

Disk and file system features the filer supports

The filer supports the disk and file system management features listed in the following table.

Read...	To learn about...
"RAID support" on page 7	Redundant Array of Independent Disks (RAID)
"Single and multiple file system (volume) support" on page 8	Single and multiple file systems
"Hot spare and hot swap disk support" on page 9	Hot spare and hot swap disks
"Quota support" on page 10	Quotas
"Saving and restoring filer data" on page 11	Backup, restore, and snapshots

RAID support

About RAID

RAID technology protects data against loss due to disk failure. When a disk in a RAID group fails and is replaced, the filer uses the undamaged disks in the RAID group to reconstruct the data that was stored on the damaged disk.

RAID level 4 support

The filer supports RAID level 4.

RAID groups

RAID technology is implemented by assigning multiple disks to RAID groups.

Minimum RAID group size

You must assign at least two disks—a data disk, and a parity disk—to a RAID group.

Maximum RAID group size

You can assign up to 52 disks to a RAID group.

Maximum number of RAID groups

The filer supports up to 32 RAID groups.

Single and multiple file system (volume) support

What the filer supports

The filer supports both single and multiple file systems, known as volumes.

Single volume support

The default filer configuration is a single volume. When the filer is initially set up, the default volume can contain up to 28 disks.

A single volume configuration minimizes administrative overhead.

Multiple volume support

After initial setup, you can create additional volumes.

A multiple volume configuration increases administrative overhead, but provides greater control over data access.

Hot spare and hot swap disk support

What the filer supports

The filer supports

- hot spare disks
- hot swapping of disks

Hot spare disks

Hot spare disks are those disks that are installed but not assigned to a RAID group. Hot spare disks do not contain data.

If a hot spare disk is available when a disk in a RAID group fails, the filer adds a hot spare disk to the RAID group and reconstructs the failed disk's data on it.

The filer can also expand the size of a RAID group automatically by adding a hot spare disk to the RAID group when it needs more storage.

Hot swapping of disks

Hot swapping a disk means adding, removing, or replacing it while the filer is running.

Support for hot swapping disks increases filer availability by enabling you to perform many disk maintenance tasks without shutting down the filer.

Quota support

About quotas

Quotas enable you to restrict disk space usage.

How quotas are specified

Quotas are specified by users and groups.

Scope of quotas

Quotas apply to a volume. On a filer that has multiple volumes, each volume can have an independent set of quotas.

Saving and restoring filer data

Features the filer supports

The filer supports the following features for saving and restoring copies of the filer data:

- snapshots
- backup and restore
- Network Data Management Protocol (NDMP)

Snapshots

Snapshots are data backups that are stored on the filer. The filer creates snapshots automatically based on a schedule you specify. You can schedule snapshots to occur weekly, nightly, and hourly.

Users can retrieve copies of their files from snapshots without your assistance.

Backup and restore

Backups are copies of filer volumes that are stored on tape. You can restore to the filer data that is backed up on tape. You can back up and restore files while the filer is on-line.

NDMP

The NDMP protocol is used by third-party backup management programs. Support for NDMP enables data on the server to be backed up and restored from remote hosts using NDMP-compliant software.

Using FilerView to administer the filer

About FilerView

You can administer the filer using the Network Appliance FilerView™ Web-based administration tool for Data ONTAP.

FilerView enables you to perform most administration tasks from any client on your network that has a compatible Web browser. FilerView uses Java applets to perform tasks that otherwise require you to enter commands at the filer console or edit configuration files.

You can use FilerView even if you did not purchase a license for the HTTP protocol.

For more information about using FilerView, refer to the *Getting Started Guide*.

Routine Filer Administration 2

About this chapter

This chapter describes routine filer administration procedures that you need 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.

Other protocol-independent administration procedures

Use the following table to locate procedures for managing other filer features that are independent of file-sharing protocols.

Read...	If you want to manage...
Chapter 3, "Managing Disks and File Systems"	Disks, file systems (volumes), or RAID groups.
Chapter 10, "Snapshots"	Snapshots.
Chapter 12, "Managing Quotas and Maximum Number of Files	Quotas or the maximum number of files the filer stores.
Chapter 13, "Backing Up and Restoring Files"	Backups.

Using the administration host

About the administration host

The filer recognizes a single client computer as the administration host. The administrator who set up the filer specified the name of the administration host using the `setup` program.

Administration host privileges

The filer granted root permissions to the administration host after the `setup` procedure was completed. The following table describes the administration host's privileges.

If the administration host is...	You can...
an NFS client	◆ Mount the filer root directory and edit configuration files from the administration host. ◆ Enter filer commands by using a remote shell program such as `rsh`.
a CIFS client	Edit configuration files from any CIFS client as long as you connect to the filer as root or "Administrator."

Administration host entry in the /etc/hosts.equiv file

The `setup` procedure placed the administration host name in the */etc/hosts.equiv* file automatically.

Administration host as the mail host

The administration host is the mail host for the filer. The filer configures the administration host to send periodic email messages to Network Appliance Technical Support about the status of your filer automatically.

For the administration host to send email, it must provide a server for the SMTP protocol, such as the `sendmail` program, or the Microsoft Exchange server.

Designating a different mail host

You can designate another host at your site to be the mail host at any time. Refer to “Sending automatic email to Network Appliance” on page 63 for more information about how the filer sends automatic email messages. Refer to “Use the options command to configure autosupport” on page 65 for information about how to specify a different mail host.

Requirements for using an NFS client as the administration host

If you plan to use an NFS client to manage the filer, the client must

- support a text editor that can display and edit text files containing lines ending with the newline character
- support the `telnet` and `rsh` commands
- be able to mount directories using the NFS protocol

Requirements for using a CIFS client as the administration host

If you plan to use a CIFS client to manage the filer, the client must support the `telnet` and `rsh` commands.

The root volume

About the root volume

Every filer has a root volume. It is the volume from which the filer reads configuration files. During `setup`, the filer creates a default volume, named *vol0*, and designates it as the root volume.

Designating the root volume

If you add volumes to your filer, you choose a volume to be the root volume during the multivolume configuration process.

For more information about designating the multiple volume configuration process, read Chapter 3, "Managing Disks and File Systems."

About the volume name prefix

Volume names begin with the following prefix:

/vol/

Syntax to refer to the root volume from NFS clients

To refer to the root volume when mounting the root volume to an NFS client, use the following syntax:

/vol/vol0

Editing configuration files

What editor to use

The filer does not include a local editor. You must use an editor from a client to change filer configuration files.

Where configuration files reside

Configuration files reside in the */etc* directory in the filer's root volume.

Choosing an NFS or a CIFS client

The following table describes which procedure to use to edit configuration files from an NFS client or a CIFS client.

If the client is...	Read...
an NFS client	"Editing files from an NFS client" on page 17
a CIFS client	"Editing files from a CIFS client" on page 18

The procedure for modifying configuration files is different depending on whether you edit the files from an NFS client or a 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 the *Getting Started Guide*.

Editing files from an NFS client

The following table describes how to edit configuration files from an NFS client.

Step	Action	
1	**If the NFS client is...**	**Then...**
	the administration host	Mount the filer root volume to the host.
	not the administration host	1. Mount the filer root volume to the administration host. 2. From the administration host, edit the */etc/exports* file on the root volume to grant root permission to the client. 3. Use the filer console, a `telnet` client, or the `rsh` command to issue the following command to the filer: **`exportfs`** 4. Mount the filer root volume to the client.
2	From the client, use a text editor to edit the files in the */etc* directory.	

Editing files from a CIFS client

The following table describes how to edit configuration files from a CIFS client.

Step	Action
1	Connect from a CIFS client to the filer as Administrator. After `setup` finished, the default */etc/passwd* and */etc/group* files on the root volume were set up to enable you to share files on the filer as Administrator.

Step	Action
2	Display the contents of the filer's *C$* share and select a file to edit. After `setup` finished, the filer root directory was shared automatically as *C$*. The Administrator 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 the filer shell

Ways to access the command line

The filer supports a command-line interface. You can access the command line

- directly, from the system console
- remotely, using `telnet`
- remotely, using a remote shell such as `rsh`

Sharing a single telnet and console session

The console and `telnet` share a single session. Everything entered through `telnet` is echoed at the console; everything entered at the console is echoed to the `telnet` session

telnet session restriction

Only one `telnet` session can be open at a time.

Closing a telnet session

To close a `telnet` session, press Ctrl-] to log out of the filer, then press Ctrl-D to log out of `telnet`.

telnet and console password requirement

Although `telnet` and the console share the same shell session, `telnet` and the console each prompt you for a password. Both the console and `telnet` connections use the same password.

rsh support

The filer supports `rsh` with trusted remote hosts—those remote hosts listed in */etc/hosts.equiv* on the root volume.

In addition to entering `rsh` commands manually, you can use a shell script or `crontab` file to enter some commands.

Note

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

Commands accepted from rsh

The following table lists the filer commands that you can execute using `rsh`.

`cf`	`halt`	`quota`	`sysconfig`
`cifs`	`httpstat`	`raid`	`sysstat`
`date`	`mt`	`rdate`	`timezone`
`df`	`netstat`	`reboot`	`uptime`
`disk`	`nfsstat`	`restore`	`version`
`download`	`options`	`route`	`vif`
`dump`	`partner`	`snap`	`vol`
`exportfs`	`qtree`	`snmp`	`ypwhich`

Use Ctrl-C to abort the command that is running

Ctrl-C aborts whatever command is being run from the console or a `telnet` session.

Because the console and `telnet` share a single session, a command entered at the console or through `telnet` can be terminated inadvertently from either location.

To ensure that a command is not terminated by Ctrl-C, start the command from a trusted host through `rsh`.

Changing the system password

A system password is required to establish a console or `telnet` connection with the filer. The password was specified during `setup`.

You can change the system password at any time with the `passwd` command. When you enter the `passwd` command, the filer prompts you to enter the old password, if any, and then requests the new password twice.

Where to go to learn more about security

For information about ways to increase filer security in addition to password protection, use the options described in “Using options command options to maintain filer security” on page 70.

Halting and rebooting the filer

Data storage in NVRAM

The filer stores requests it receives in nonvolatile random-access memory (NVRAM). The use of NVRAM

- improves system performance
- prevents loss of data in case of a system or power failure

NVRAM event during orderly shutdown

The `halt` and `reboot` commands perform an orderly shutdown. During an orderly shutdown, the contents of NVRAM are flushed to disk.

Procedure to halt the filer

To halt the filer, enter the following command:

```
halt
```

The filer displays the following prompt:

```
ok
```

Procedure to boot the filer

Complete the following step to boot the filer.

Step	Action
1	Ensure that the **ok** prompt is displayed on the console.
2	Enter the following command: **boot**

Procedure to reboot the filer

You can halt and reboot the filer in a single operation by entering the following command:

```
reboot
```

Where the filer boots from

When the filer boots, it uses the boot diskette in its diskette drive, if there is one. Otherwise, the filer boots from its hard disk.

Use the halt command to avoid data loss

You should always execute the `halt` command before turning the filer Off for the following reasons:

- The `halt` command flushes all data from NVRAM to disk, eliminating a potential point of failure.
- The `halt` command avoids potential data loss on CIFS clients.

  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.

Caution

Never interrupt CIFS service by halting the filer without giving advance warning to CIFS users.

Before turning the filer Off, use the `halt` or `cifs terminate` command to send a warning message to CIFS users. This method gives users an opportunity to save files and exit applications within the time period that you specified before the actual shutdown.

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 capability to display messages from the filer is built into Windows NT.

For more information

For more information about `cifs terminate`, refer to "Stopping and restarting CIFS sessions" on page 212.

Understanding the filer default configuration

About the default configuration

The default configuration of a filer depends on whether the filer is running NFS, CIFS, or both; and NIS.

Although 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 in this chapter describe in greater detail how you might want to modify the default configuration to suit your needs.

Default exported and shared directories

Default directories created

When `setup` finishes, two default directories are made available for access by clients. The default directories are

- the root directory
- the */home* directory

Permissions for the default directories

The following table shows the permissions that are assigned to the default directories when `setup` finishes.

This directory...	From this client...	Has these permissions...
The root directory	NFS	◆ full permissions for the root user on the administration host ◆ no permissions for any other user or host ◆ no permissions
	CIFS	◆ read and write permissions to all files for the Administrator user when logged in to the filer using the root password ◆ no permissions for other users
The */home* directory	NFS	permissions associated with individual users and with groups through UNIX security database
	CIFS	permissions assigned by the filer administrator for the HOME$ share

Accessing the directories

The following table shows how to access the default directories. Replace *filer* with the host name of your filer.

To access...	From this client...	Do this...
The root directory	NFS	Mount */filer/vol/vol0*.
	CIFS	Map a drive to *filer**C$*.
The */home* directory	NFS	Mount */filer/vol/vol0/home*.
	CIFS	Map a drive to *filer**HOME$*. Or Use the Network Neighborhood icon to locate the filer and locate the *HOME$* share.

Contents of the etc directory

The root directory contains an *etc* directory in which the filer configuration files are stored. You can modify the configuration files from the administration host.

The following table describes the files in the *etc* directory. Note that some of the configuration files might not exist on your filer if you use the filer for CIFS or NFS only.

File name	Contents	File-sharing protocol
.cifs.cat	Domain information (only if the filer is a member of a domain)	CIFS only
cifsconfig.txt	CIFS commands that the filer used for configuration The filer maintains this file automatically whenever you enter a `cifs` command, which can cause changes you make manually to be lost. To edit this file, terminate CIFS service by using the `cifs terminate` command, edit and save the file, then reboot the filer.	CIFS only

File name	Contents	File-sharing protocol
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 (group identification numbers), and members' names Not used if you use NIS to authenticate groups.	CIFS only
netgroup	Network groups	NFS only
nsswitch.conf	The order in which the filer contacts name services	All
passwd	Users' names, UIDs (user identification numbers), and primary GIDs Not used if you use NIS to authenticate users.	CIFS and (PC)NFS
rc	Script of commands to be executed when the filer is initializing	All
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 Not used if you use NIS to authenticate users.	CIFS and (PC)NFS

How the /home directory is used

How the */home* directory is used depends on the file-sharing protocol used by the client. The following table describes how NFS and CIFS clients use the directory.

For NFS clients	For CIFS clients
The */home* directory is exported with read and write permissions to all NFS clients, and with root access to the administration host. Clients can mount directories after the filer host name is added to the filer */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 to prevent users from deleting each other's files. This is a safe configuration for networks that use UNIX permissions to provide security for network-accessible files. You can clear the sticky bit with the `chmod` command.	The */home* directory can be shared by CIFS users as the home share immediately after `setup` finishes. Before users can read and write files in the home share, you must follow the procedures in "Creating a home share for each user" on page 194. By default, CIFS users can write and delete their own files in */home*. For more information about access rights, refer to "Assigning and changing access rights" on page 197.

The /etc/rc file

How the filer uses the /etc/rc file

The filer executes the commands in the */etc/rc* file on the root volume at boot time to configure the filer.

If your filer is licensed to run the CIFS protocol, the */etc/rc* file must be present at boot time for the CIFS protocol to be enabled. No CIFS-specific information is entered in */etc/rc* as a result of the `setup` procedure.

All the commands in the */etc/rc* file are executable from the command line—there are no commands that are restricted to being executed from within the */etc/rc* file.

Procedure for editing the /etc/rc file

The following table describes how to make changes in the filer configuration that take effect every time the filer is booted.

Step	Action
1	Make a backup copy of the */etc/rc* file.
2	Edit the */etc/rc* file. **Note** Do not add CIFS commands in */etc/rc*. See “Editing configuration files” on page 17 for instructions about editing files from NFS and CIFS clients.
3	Save the edited file.
4	Reboot the filer to test the new configuration. If the new configuration does not work as desired, repeat Steps 2 through 4.

Default /etc/rc file contents

The best way to understand the commands used in */etc/rc* on the root volume is to examine the following sample */etc/rc* file:

```
#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
savecore
exportfs -a
nfs on
```

Explanation of default /etc/rc contents

The following table explains the commands in the sample */etc/rc* file.

Command	Explanation
`hostname toaster`	Sets the filer’s host name.

Command	Explanation

`ifconfig e0 'hostname'-0`
`ifconfig e1 'hostname'-1`
`ifconfig f0 'hostname'-f0`
`ifconfig a5 'hostname'-a5`
	Sets the IP address for the filer's Ethernet, Gigabit Ethernet, FDDI (Fiber Distributed Data Interface), and ATM (Asynchronous Transport Mode) interfaces with a default network mask. (The network mask is set during `setup`, as described in the *Getting Started Guide*.) 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 on the root volume; you might prefer to enter IP addresses directly in */etc/rc* on the root volume. 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 34 for more information about interface names.
	
	If you change the filer's host name, you must modify the */etc/hosts* file on the root volume 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`
	
	The preceding command specifies the default router. You can add route commands to */etc/rc* on the root volume to set static routes for the filer. The network address for LocalRouter must be in */etc/hosts* on the root volume.

`routed on`	Starts the routing daemon. See "Routing" on page 107 for more information about routing.

Command	Explanation
`options dns.domainname netapp.com` `options dns.enable on`	These options set the DNS domain.
`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 on the root volume describes the time zone you want. A list of time zone names is in the *Getting Started Guide*.
`savecore`	Saves the core file from a system panic, if any, in the */etc/crash* directory on the root volume. Core files are created only during the first boot after a system panic.
`exportfs -a`	Exports all directories specified in the */etc/exports* file on the root volume. 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 the root volume on a filer without an NFS license, the command fails and the following messages appear:
	
	`NFS service is not licensed.`
	`(Use the "license" command to license it.)`
	`NFS server is NOT running.`

Changing SNMP commands in /etc/rc

Regardless of whether SNMP is enabled at your site, add the following commands to */etc/rc* on the root volume:

```
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 enables 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 “Sending automatic email to Network Appliance” on page 63.

Naming conventions for network interfaces

Interface types the filer supports

The filer supports the following interface types:

- Ethernet
- Gigabit Ethernet
- FDDI
- ATM

How interfaces are numbered

Interface numbers are assigned based on the slot in which the interface card is installed. For more information about how network interfaces are numbered, refer to the hardware guide for your filer model.

How multiple ports are identified

Some Ethernet interface cards support four ports. These cards are referred to as quad-port interfaces. The filer uses a letter to refer to each port on a quad-port interface. The following table shows the relationship of port numbers to letters.

Port number	Letter
1	a
2	b
3	c
4	d

How interfaces are named

The following table shows how interfaces are named, and how interface names are combined with card slot numbers and port letters to make their names unique.

Interface type	Letter used in name	Examples of names
Ethernet (single)	e	e0 e1

Interface type	Letter used in name	Examples of names
Ethernet (quad-port)	e	e0a e0b e0c e0d e1a e1b
FDDI	f	f0 f1
ATM	a	a0 a1

About using interface names in scripts

When you write or modify shell scripts that involve interface names, remember the interface naming conventions that the filer uses. Because the slot in which an interface card is installed is a part of the interface name, different filers might have different interface names.

Filer host names

The first time the setup program runs, the filer creates a host name for each installed interface by appending the interface name to the host name.

Host name examples

Example 1: A filer named toaster with a single Ethernet interface in slot 0 and a quad-port Ethernet interface in slot 1 has the host names given in the following table.

Interface	Host name
Single-port Ethernet card in slot 0	toaster-0
Quad-port Ethernet card in slot 1	toaster-1a toaster-1b toaster-1c toaster-1d

Example 2: A filer named toaster with an FDDI interface in slot 1 would have the host name `toaster-f1`.

Recovering from configuration errors

Reasons to follow a special recovery procedure

Certain configuration errors can require you to follow a special recovery procedure because

- The filer does not have a local text editor.
- Problems with interface configuration can make the filer inaccessible to clients from which the */etc/rc* file can be edited.

Procedure when the filer does not boot

If configuration errors prevent the filer from booting from the hard disk, try booting from the diskette. For more information, see "Booting from system boot diskette" on page 392.

Procedure when administration host cannot access the filer

If your filer becomes inaccessible from the administration host after you change the */etc/rc* file, use the procedure in the following table to recover from the error.

Step	Action
1	Enter commands on the console to configure the interface with the correct address.
2	Enter the `exportfs` command to export the filer root directory to the administration host.
3	Edit the filer */etc/rc* file from the administration host.
4	Reboot the filer. If the changes do not correct the problem, repeat Steps 1 through 4.

Core files

About core files

When a hardware or software failure causes the filer to crash, the filer creates a core file that Network Appliance Technical Support can use to troubleshoot the problem.

Core file storage in /etc/crash

On the first boot after a system crash, the filer stores the core file in the */etc/crash* directory on the root volume.

What the savecore command does

The `savecore` command, which is included in the default */etc/rc* file on the root volume

- Produces a *core.n* file. The *n* in the file name is a number.
- Displays a message on the system console.
- Logs a message in */etc/messages* on the root volume.

The core.n file contents

The *core.n* file includes more information that might be essential for determining the cause of the crash; it is often necessary for investigating problems involving the file system.

Core dump space needed

A core dump file contains the contents of memory and NVRAM. Core dumps are written to a set of areas at the beginning of all the disks. The core dump area on each disk has a fixed size of approximately 20,447,232 bytes. Therefore, a filer or appliance with a large amount of memory can have an insufficient amount of core dump disk space to store a full core dump.

The following table shows how many disks are needed to store a full core dump for the amount of memory you might have in a specific filer or appliance.

Memory (in MB)	Disks Needed
32	2
64	4

Memory (in MB)	Disks Needed
96	5
128	7
160	9
192	10
224	12
256 or more	14

Automatic technical support notification

If you followed the instructions for the system setup procedure in the *Getting Started Guide*, your filer sends email automatically to Network Appliance Technical Support upon each system reboot.

Manual technical support notification

If you disabled automatic email, you should contact Network Appliance Technical Support when your system creates a core file.

Message logging

About message logging

The filer maintains messages in the */etc/messages* file on the root volume.

The level of information that the filer records in the */etc/messages* file is configurable.

About the syslogd daemon and the /etc/syslog.conf file

The message logging daemon, `syslogd`, uses the */etc/syslog.conf* configuration file on the filer root volume to determine how to log system messages.

You can configure `syslogd` to direct system messages to the console, to a file, or to a remote system based on their severity and origin.

By default, all system messages (except for those of severity level debug) are written to the console and to the */etc/messages* file on the root volume.

The /etc/syslog.conf file format

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

`facility.level action`

The facility parameter

The `facility` parameter specifies the subsystem from which the message originated. The following table describes the `facility` parameter keywords.

Keyword	Description
`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
*	messages from all facilities

The level parameter

The `level` parameter describes the severity of the message. The following table describes the `level` parameter keywords arranged in order from highest to lowest severity.

Level	Description
`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)
*	specifies all levels

The action parameter

The `action` parameter specifies where messages should be sent. Messages for the specified level or higher are sent to the message destination. The following table describes the possible actions and gives examples of each action.

Action	Example
Send messages to a file specified by a path.	*/etc/messages*
Send messages to a host name preceded by an @ sign.	*@adminhost*
Send messages to the console.	*/dev/console* or *

Example line from /etc/syslog.conf

The following example causes all kernel messages of levels `emerg`, `alert`, `crit`, and `err` to be sent to the */etc/messages* file:

```
err.kern /etc/messages
```

The /etc/messages file restart schedule

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

Checking the /etc/messages file daily

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.

Sample /etc/syslog.conf file

The following example shows a customized */etc/syslog.conf* file:

```
# 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
```

For more information

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

Configuring filer options

Commands to use to set options

The filer recognizes two commands, `options` and `vol options`, to set options.

"The options command" on page 43 describes the syntax for the options command.

"The vol options command" on page 44 describes the syntax for the vol options command.

The options command

What the options command does

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

Syntax of the options command

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

`options [ option [value] ] ...`

If you omit a value for an option, the command displays the current value of the option.

The following table describes the variables.

Variable	Description
option	the name of the option
value	the value of the option

Example of the options command

The following command enables the `pcnfsd` daemon:

```
options pcnfsd.enable on
```

The vol options command

vol options command configures volume-level behavior

You use the `vol options` command to configure volume-level behavior. You can use this command only with volume options, which are listed in "Volume options" on page 59.

Changes made with the `vol options` command are persistent between reboots—you do not need to add them to the /etc/rc file.

Syntax of the vol options command

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

`vol options` *`volname option`* [*`value`*]

If you omit a value for an option, the command displays the current value of the option.

The following table describes the variables.

Variable	Description
`volname`	the name of the volume that the option applies to
`option`	the name of the option
`value`	the value of the option

Example of the vol options command

The following command sets the maximum size of a RAID group in the volume named myvol to 12:

```
vol options myvol raidsize 12
```

Detailed option information

About setting detailed information

Options are described in detail in other sections of this guide and in the man pages. The *na_options*(1) man page contains information about `options` command options. The *na_vol*(1) man page contains information about the `vol options` command options.

The sections that follow provide a brief description of each option grouped by function. The default value is below the option name.

Option values

The following conventions apply 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."

The values for On and Off are not case-sensitive. If you do not supply a parameter in the `options` command, the command prints the current values of all available options.

Autosupport options

What the autosupport options do

The autosupport options control whether and how the filer sends automatic status messages.

For more information about autosupport, see “Sending automatic email to Network Appliance” on page 63.

The autosupport.doit option

Default: None

Description: 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.

The autosupport.enable option

Default: On

Description: Enables the `autosupport` daemon, which sends automatic email messages to report the status of the filer.

The autosupport.from option

Default: autosupport

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

The autosupport.mailhost option

Default: administration_host

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

The autosupport.noteto option

Default: None

Description: Specifies up to five recipients of an automatic short email message. Use a comma-separated list with no spaces.

The autosupport.to option

Default: autosupport@netapp.com

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

CIFS options

What the CIFS options do

The CIFS options control CIFS features on the filer.

The cifs.generic_account option

Default: pcuser

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

The cifs.guest_account option

Default: None

Description: When you leave `cifs.guest_account` blank, a CIFS user can log in to the filer without an account in the password database, provided that a domain controller authenticates the user.

When you set `cifs.guest_account` 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 more information, see "Enabling guest access" on page 178.

The cifs.home_dir option

Default: None

Description: 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.

The cifs.idle_timeout option

Default: 60000

Description: Specifies the number of milliseconds that elapse before the filer disconnects an idle session.

The cifs.oplocks.enable option

Default: On

Description: When this option is on, the filer enables clients to use oplocks (opportunistic locks) on files. Oplocks provide 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, you should disable this option only if there are problems with databases and to isolate problems.

The cifs.netbios_aliases option

Default: None

Description: Specifies a list of alternative names for the filer. Use a comma-separated list of names.

The cifs.scopeid option

Default: blank

Description: 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.

This option is not recommended if you are using DNS for name resolution because NetBIOS Scope IDs and DNS are incompatible.

For more information, refer to Microsoft documentation.

The cifs.show_snapshot option

Default: FALSE

Description: Specifies whether to show the *~snapshot* snapshot directories in folders. To show the snapshot directories, set this option to TRUE.

The cifs.symlinks.cycleguard option

Default: On

Description: 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. It does so 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 more information about this option, refer to ““Preventing symbolic link cycling” on page 125.

The cifs.symlinks.enable option

Default: On

Description: When you set `cifs.symlinks.enable` 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. This ensures 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 more information about this option, refer to “Managing symbolic links for CIFS access” on page 123.

Cluster failover options

What cluster failover options do

The cluster failover options enable and disable features of cluster failover.

The cf.timed.enable option

Default: On

Description: Enables time synchronization on a filer in a cluster. Both filers in a cluster should use the same setting for this option.

DNS options

What the DNS options do

The DNS options control how the filer works with DNS.

For more information about DNS, see Chapter 4, "Network Administration."

The dns.domainname option

Default: None

Description: Sets the DNS domain name to the specified domain name.

The dns.enable option

Default: Off

Description: 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

What the HTTP options do

The HTTP options enable and control HTTP services.

For more information about HTTP on the filer, see Chapter 8, "HTTP Administration."

The httpd.admin.enable option

Default: On

Description: Enables HTTP access to the filer's on-line Help files and other files used by FilerView.

The httpd.enable option

Default: Off

Description: Enables the HTTP server.

The httpd.log.max_file_size option

Default: 2147483647 (2 GB - 1 byte)

Description: Specifies the number of bytes */etc/log/httpd.log*, the HTTP log file, can grow to. The maximum value is 500 GB.

The httpd.rootdir option

Default: None

Description: Specifies the root directory containing files and directories that HTTP transfers to clients.

The httpd.timeout option

Default: 900 seconds (15 minutes)

Description: Specifies the minimum amount of time, in seconds, before an idle HTTP connection times out.

The httpd.timewait.enable option

Default: On

Description: When you set this option to On, the filer drops an HTTP connection one minute after the client closes it. When you set this option to Off, the connection is not dropped and resources are consumed until the connection times out.

NFS and (PC)NFS options

What the NFS and (PC)NFS options do

The NFS and (PC)NFS options enable and control NFS services.

For more information about NFS, see Chapter 6, "NFS Administration."

The nfs.mount_rootonly option

Default: On

Description: Configures the filer to give NFS access only to root by requiring that mount requests come from privileged ports (ports 0 through 1023).

The nfs.per_client_stats.enable option

Default: Off

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

The nfs.tcp.enable option

Default: Off

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

The nfs.v2.df.2gb.lim option

Default: Off

Description: Limits to 2 GB the response the filer gives to requests from NFS v2 clients 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 might appear to be full to the client that initiates the "file system statistics" request.

The nfs.v3.enable option

Default: On

Description: 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.

The nfs.webnfs.enable option

Default: Off

Description: Turns WebNFS On and Off.

The nfs.webnfs.rootdir option

Default: None

Description: 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.

The nfs.webnfs.rootdir.set option

Default: FALSE

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

The pcnfsd.enable option

Default: Off

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

The pcnfsd.umask option

Default: 022

Description: Specifies that users can read and write their own files, but members of their group and others can only read the files. This option uses the standard UNIX permission mask.

NIS options

What the NIS options do

The NIS options control how the filer works with NIS.

For more information about NIS, see Chapter 4, "Network Administration."

The nis.domainname option

Default: None

Description: Sets the NIS domain to the specified domain name.

The nis.enable option

Default: Off

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

RAID options

What the RAID options do

The RAID options control how the filer uses RAID.

For more information about RAID on the filer, see Chapter 3, "Managing Disks and File Systems."

The raid.reconstruct_speed option

Default: 4

Description: 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.

The raid.scrub.enable option

Default: On

Description: Specifies whether the filer performs RAID scrubbing.

The raid.timeout option

Default: 24

Description: Sets the time, in hours (from 1 through 24), that the system runs in degraded mode before an automatic shutdown.

Volume options

What the volume options do

The volume options control volume-level operations. You use these options only with the `vol options` command.

For more information about volume options, see “The vol options command” on page 44.

The minra option

Default: Off

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

The no_atime_update option

Default: Off

Description: 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).

The nosnap option

Default: Off

Description: Temporarily disables automatic snapshots.

The nosnapdir option

Default: Off

Description: Makes invisible the snapshot directory that’s usually 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 might 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.

The raidsize option

Default: None

Description: Sets the maximum size of a RAID group in *volume*. Must be an integer greater than one.

The root option

Default: None

Description: Makes *volume* the root volume.

Miscellaneous options

What the miscellaneous options do

The miscellaneous options control additional aspects of filer operation.

The ip.path_mtu_discovery.enable option

Default: On

Description: Enables or disables path MTU discovery, which is currently used only by TCP. When enabled, it enables 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.

The rps.status option

Default: None

Description: When you set this option to Good, it clears the message that a redundant power supply is bad. Set this option to Good after you replace a failed power supply in the RPS; otherwise, you 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 RPS.

The telnet.hosts option

Default: All hosts

Description: 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 (-).

The wafl.maxdirsize option

Default: 10240

Description: Sets the maximum size, in kilobytes, of a directory file. A directory file with a size of 10,240 kilobytes can hold about 300,000 files or subdirectories.

The wafl.root_only_chown option

Default: On

Description: Enables only the root user to change the owner of a file. When you disable the option, 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 fails.

The vol.copy.throttle option

Default: 10

Description: Specifies the default speed of `vol copy` operations. The speed ranges from 10 (full-speed) to 1 (one-tenth of full-speed).

Sending automatic email to Network Appliance

How automatic email messages are controlled

The filer uses the `autosupport` daemon to control 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.

Technical support response

If necessary, Network Appliance contacts you at the email address that you specify to help resolve a potential system problem.

Mail host requirement for autosupport

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 `setup` is used as a mail host. You can specify more mail hosts.

About configuring autosupport

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 "Use the options command to configure autosupport" on page 65 for more information about specifying the email address and other options.

Events that trigger autosupport email

The mail host sends email to Network Appliance Technical Support about your filer after any of the events listed in the following table.

Event	Subject line of the email message
Low NVRAM lithium battery	`BATTERY_LOW!!!`
Disk failure	`DISK_FAIL!!!`
Disk scrub occurred	`DISK_SCRUB!!!`
Fan failure	`FAN_FAIL!!!`
Shutdown because of overheating	`OVER_TEMPERATURE_SHUTDOWN!!!`
Partial RPS failure[a]	`POWER_SUPPLY_DEGRADED!!!`
System reboot	`REBOOT`
Disk shelf error	`SHELF_FAULT!!!`
Spare disk failure	`SPARE FAIL!!!`
Weekly backup of */etc/messages*	`WEEKLY_LOG`
`option autosupport.doit` command	`THE STRING SPECIFIED IN OPTION autosupport.doit`
Successful cluster takeover of partner	`CLUSTER TAKEOVER COMPLETE`
Unsuccessful cluster takeover of partner	`CLUSTER TAKEOVER FAILED`
Cluster takeover of virtual filer	`REBOOT (CLUSTER TAKEOVER)`
Cluster giveback	`CLUSTER GIVEBACK COMPLETE`

a. 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
  cf monitor
  ```

- the following information, if clustering is licensed:
  - system ID of the cluster partner
  - host name of the cluster partner
  - cluster node status
- contents of */etc/messages*
- contents of */etc/serialnum*

Use 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 as follows:

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

Example: `autosupport.enable off`

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

`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. The default is the administration host.

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 as follows:

`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. The default is autosupport@netapp.com.

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

Be sure to enter the command on a single line.

Specifying the filer administrator's address

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

`options autosupport.from` *`address`*

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

If Network Appliance Technical Support needs to determine whether further action is necessary for solving or preventing filer problems, it sends email to jdoe@abc.com.

Sending an immediate message

Immediate messages contain the same filer data as automatic messages.

How to send immediate messages: The command to send an automatic email message immediately is as follows:

`options autosupport.doit string`

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

Example: `options autosupport.doit TESTING`

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 example shows a 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 the screen on an alphanumeric pager.

Synchronizing filer system time

Commands the filer uses

The filer uses two commands to control the system time.

- The `date` command sets the time locally.
- The `rdate` command synchronizes the filer's time with the time on another host.

Command to set the date and time

The following command sets the date and time to 9:25 a.m. on May 22, 1997:

```
date 199705220925
```

About 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

```
rdate time_node
```

A typical scheme is 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 on the root volume.

For example, on the UNIX computer named *adminhost*, the user named *adminuser* sets up a `cron` job to run every day at 3 a.m. It directs the filer named 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 is as follows:

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

Time synchronization within clusters

Clustered filers communicate with each other at regular intervals to provide time synchronization within the cluster. See “Controlling time synchronization” on page 246 for more information.

Using options command options to maintain filer security

What the options to the options command do

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

- The `telnet.hosts` option restricts `telnet` access to a limited number of hosts. Use this option to 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 on the root volume.

- The `mount_rootonly` option restricts the mount privilege to root using privileged ports (ports 1 through 1,024). 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 non-privileged ports to send requests. If you have these clients at your site, disable the `mount_rootonly` option or upgrade the client software.

- The `wafl.root_only_chown` option enables 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 causes the file's recipient to exceed his or her quota, the attempt fails. This option is applicable only if your filer runs NFS.

Software licenses

About software licenses

The filer requires software licenses to enable NFS, CIFS, HTTP, clustering, and volume copy 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 service 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 services

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

`license` *`protocol=code`*

Example: If the license code for NFS is ABCDEFG, enter

`license nfs=ABCDEFG`

The events that take place after a license command depend on the protocol specified. The following table discusses the events for each protocol.

Protocol	Messages

NFS	`nfs license enabled.`
	`nfs enabled.`
	
	The filer also automatically runs the `nfs on` command to start NFS service. However, the filer 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, add `nfs on` to */etc/rc*.

Protocol	Messages

CIFS	`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 already set up the CIFS configuration as described in the *Getting Started Guide*.

HTTP	`http license enabled.`
	`Use "options httpd.enable" to enable http.`
	
	To start HTTP service, enter the following command:
	`options httpd.enable on`
	
	Results: The `options` command takes effect immediately. If you want the filer to automatically start HTTP service after each reboot, add `options httpd.enable` to */etc/rc*.

VOLCOPY	`volcopy site license enabled.` `vol copy is enabled.`

CLUSTER	`cluster site license enabled.`
	`Clustered Failover will be enabled upon reboot.`
	`Make sure that each individual service is licensed`
	`on both nodes or on neither node. Remember to configure`
	`the network interfaces for the other node.`

Example: The following example shows how to activate several protocols with one command:

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

Enabling volcopy for copying volumes

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

`license volcopy=`*`code`*

Example: If the license code for volcopy is ABCDEFG, enter

`license volcopy=ABCDEFG`

Displaying current license codes

To display licensing information, enter the `license` command without parameters, as follows:

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

Disabling a license

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

```
license nfs=DISABLE
```

After you disable a license, your filer stops service for the corresponding protocol. You can restart the service by reentering the license code.

Replacing license codes

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

Managing Disks and File Systems 3

Chapter contents

This chapter provides information about, and how to manage, the filer's disks and file systems. It is divided into the following sections.

Read...	To learn about...
"Disk concepts" on page 76	Principles of RAID groups and the different functions of disks in them, checking for media errors, using disks of various sizes, and handling disk failures.
"Volume concepts" on page 83	Volume concepts.
"Procedures for managing disks and volumes" on page 87	Managing disks and volumes on a filer.

Disk concepts

Section contents

This section covers the following topics:

- Understanding RAID groups
- About disk addresses
- Use disk scrubbing to protect data from media errors
- Understanding hot spare disks
- Understanding hot swap
- Using disks of various sizes
- Understanding usable space on each disk
- Handling disk failures
- Effects of disk failure on filer operation

Understanding RAID groups

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 enables you to incrementally expand the filer's disk storage capacity.

In a RAID group, different disks have different functions. Most of the disks in the RAID group are *data disks*. One disk is the *parity disk*, which enables the filer to recover the data on a data disk if one fails.

Multiple RAID groups: Beginning with Data ONTAP 5.0, the filer supports multiple RAID groups. The factory default filer configuration contains one RAID group. The filer supports up to 32 RAID groups. Each RAID group belongs to only one volume; you cannot assign more than one volume to the same RAID group.

Spare disks are used by RAID groups as needed. They do not have to be in the same disk shelf to be available to a RAID group. The filer automatically assigns disks to RAID groups and creates new RAID groups as each RAID group is filled with its maximum number of disks.

RAID group size: The following characteristics apply to RAID group size:

- The default number of disks in a RAID group (including the parity disk) is 14.
- A RAID group must contain at least two disks.

- The largest RAID group size you can create *manually* is 28 disks.
- The maximum RAID group size is 52 disks.

Occasionally, you might encounter situations in which you want to specify a RAID group size other than the default. For example, you might want to configure a filer with smaller RAID groups for the following reasons:

- Using smaller RAID groups reduces disk reconstruction time if a disk fails.
- Using smaller (therefore, more) RAID groups provides higher reliability by reducing the risk of data loss due to multiple-disk failure.

  Conversely, configuring with larger (therefore, fewer) RAID groups in a filer uses fewer disks for parity, leaving more disks available for data storage.

Parity disks: In each RAID group, the filer assigns the role of parity disk to the largest disk in the RAID group. After a data disk failure, the filer uses the parity disk in conjunction with the other data disks to reconstruct the failed disk's data and optionally write it to a hot spare disk. The parity disk must be at least as large as the largest data disk.

For more information about how the filer recovers from disk failures by using the parity disk or hot spare disk, refer to "Handling disk failures" on page 81.

About disk addresses

You identify a disk by its address, which is listed in the `HA.Disk_ID` column of the output of the `sysconfig -r` command. In this output listing, `HA` refers to the host adapter and `Disk_ID` refers to the disk ID number.

You use the disk address to

- interpret screen messages (for example, command output or error messages) that you see on your display
- quickly locate the disk that the message is referring to

SCSI disk addresses: With SCSI disks, the disk address 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.

Fibre channel disk addresses: With fibre channel disks, the disk address is a combination of the disk's adapter number and the disk's fibre channel loop ID. To create the fibre channel loop ID, multiply the shelf ID switch value by eight and add it to the bay number. For example, `ha 8`, `shelf 1, disk 2` has disk ID `8.10`.

For more information: For more information about locating specific drives on disk shelves, refer to the hardware guide for the disk shelf.

Use disk scrubbing to protect data from media errors

The filer uses the RAID disk scrubbing procedure to increase data availability. The filer scans each disk in the RAID group for media errors. If the filer finds media errors, it fixes them by reconstructing the data from parity and rewriting the data.

Disk scrubbing reduces the chance of a multiple disk failure, caused by a disk media error encountered while the system was running in degraded mode.

The filer only scans a RAID group when all the group's disks are operational. Although disk scrubbing slows the filer somewhat, network clients might not notice the change in the filer's performance because disk scrubbing starts automatically at 1:00 A.M. on Sunday, when most systems are lightly loaded.

By default, disk scrubbing is enabled. You might want to disable scrubbing if you have a recurring problem that scrubbing encounters.

Example: For example, there might be an unrecoverable error on a disk that you cannot fix before the next disk scrub. The following commands disable and enable disk scrubbing:

```
options raid.scrub.enable off
options raid.scrub.enable on
```

Commands to start and stop disk scrubbing: You can also manually start and stop disk scrubbing regardless of the current value (On or Off) of the `raid.scrub.enable` option. Following are the commands for starting and stopping scrubbing manually:

```
disk scrub start
disk scrub stop
```

Sample messages logged from scrubbing: Messages from the disk scrubbing process are sent to the system error logging daemon.

Following are sample messages that can appear:

- If the filer finds an inconsistent parity block during scrubbing, it prints the following messages:

  ```
  Inconsistent parity on volume volume_name, RAID group n,
  stripe #n.
  Rewriting bad parity block on volume volume_name, RAID group n,
  stripe #n.
  ```

Note

An `Inconsistent parity` error message might indicate file system corruption. If you get such an error, contact Network Appliance Technical Support for assistance.

- If the filer finds a media error on the parity disk, it prints the following message:

  `Rewriting bad parity block on volume volume_name, RAID group n, 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 volume volume_name, RAID group n, stripe #n`

The following sample messages appear after disk scrubbing is complete:

```
Scrub found n parity inconsistencies
Scrub found n media errors
Disk scrubbing finished...
```

Understanding hot spare disks

In addition to data disks and parity disks, the filer supports zero or more *hot spare* disks. A hot spare disk is not part of any RAID group and does not contain file system data. After a disk failure, the filer automatically rebuilds data (or parity) onto a hot spare disk, which then replaces the failed disk in the RAID group. This procedure avoids a system shutdown and returns the system to full performance.

A hot spare disk cannot replace a failed disk that is larger than itself. If you use only one hot spare disk in a filer, the hot spare disk must be as large as the largest file system disk. If you have multiple hot spare disks installed, the system uses the smallest hot spare disk needed to replace the failed disk.

Understanding hot swap

The filer enables you to *hot swap* disk drives. Hot swapping a disk drive means installing or removing it from the disk shelf while the filer is running, with minimal interruption to a file system. For example, you might want to hot swap a disk into a filer to replace a disk or to add a hot spare disk.

Using disks of various sizes

You can use disks of various sizes in the same filer. The following table shows a sample configuration.

Data	Data	Data	Data	Parity	Hot Spare
Disk 0 (4 GB)	Disk 1 (4 GB)	Disk 2 (4 GB)	Disk 3 (9 GB)	Disk 4 (9 GB)	Disk 5 (9 GB)

The following sections discuss the guidelines for choosing the size of a disk to add to the filer or to use as a replacement.

Replacing a failed disk: When you replace a failed disk, the physical space of the new disk must be equal to or larger 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 physical space of the new disk is larger than the usable space of the old disk, the system can use no more space than was available on the old disk. For example, if you replace a 4-GB disk with a 9-GB disk, the extra 5 GB on the new disk is not available to the file system. Therefore, it is most cost-effective to replace a disk with a disk of the same size.

Adding a disk to a file system: When you add to a file system a disk that is larger than the current parity disk, the new disk automatically becomes the parity disk. The disk used as the parity disk before the disk addition becomes a data disk.

Understanding usable space on each disk

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 might differ slightly in size even though they belong to the same size category.
- To ensure that you can replace one disk with another in the same size category, a filer shipped with Data ONTAP 2.1 or later automatically "right-sizes" its disks. That is, it rounds off the disk capacity and divides disks into 2-GB, 4-GB, and 9-GB classes.
- If you 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.
- If a disk has never been right-sized, you can only replace it with another disk that has the same number of blocks or more.

The following table shows the usable disk space for each class of disk supported by the filer.

1-GB disks	2-GB disks	4-GB disks	9-GB disks
1,000 MB (2,048,000 blocks)	2,000 MB (4,096,000 blocks)	4,000 MB (8,192,000 blocks)	9,000 MB (17,612,800 blocks)

Although automatic disk right-sizing is not applied to existing disks in an upgraded system, it is applied to disks being added to the filer.

You can use `sysconfig -r` to compare the physical space and the usable space, and to determine whether disks are right-sized.

Handling disk failures

If one block on a data disk fails, the filer uses the parity disk in its RAID group to reconstruct the data on that block. The block is mapped to a new location on disk. If an entire data disk fails, the parity disk for that RAID group prevents any data loss and enables the filer to continue running.

Although the filer can continue to function with a failed disk, if it cannot reconstruct that failed disk on a hot spare, 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 `raid.timeout` option to the `options` command.

Effects of disk failure on filer operation

The effects of a disk failure on filer operation depend on whether the filer has a hot spare disk.

Without a hot spare disk: If the filer is not equipped with a hot spare disk, after a disk fails the filer enters a state called “degraded mode.” In this state, the RAID feature enable 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 in the same RAID group causes the entire file system to be lost.

When a disk fails, the filer logs a warning message in the */etc/messages* file and to the system console every hour, notifying you of the number of hours before the system shuts down.

The shutdown ensures that you notice the disk failure. You can restart the filer without fixing the disk, but the filer continues to shut itself off at the specified intervals until you repair the problem.

By default, the filer shuts down after 24 hours. You can change this time interval using the `option raid.timeout` 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 either of the following conditions exist:

- One disk in any RAID group has failed.

- The batteries on the NVRAM card are low (if the filer is a PCI-based. system).

With a hot spare disk: If you reserve one or more disks as hot spare disks when you configure your filer, the filer also enters degraded mode after a disk failure. However, the filer immediately begins rebuilding the missing data in the background on the hot spare disk, with minimal interruption to file service.

The filer logs this activity in the */etc/messages* file and does not automatically shut down. If you turn off the filer while it is in degraded mode, it stops data reconstruction. After you turn the filer back on, the filer restarts the data reconstruction process from the beginning.

Except for a loss in performance while data is rebuilt on the hot spare disk, the failure of a single disk is transparent to the user. The filer exits degraded mode and returns to normal operation after it finishes reconstructing the data.

Network Appliance recommends that you replace the failed disk with a new hot spare disk after the filer finishes reconstructing data. This way, the filer continues to have a hot spare disk that it can use in case another disk fails.

The `sysconfig -r` command displays which disk is reserved as the hot spare disk. In addition to disk failure and hot spare disk replacement activity, the */etc/messages* file logs any failure in a periodic check of the hot spare disk.

Command to control RAID data 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 data reconstruction consumes CPU time, sometimes increasing the speed of data reconstruction slows the filer's network operations. The default speed is 4, which means approximately 40% of the CPU time is used for RAID data reconstruction.

When RAID data 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 data reconstruction to maximize CPU utilization. For more information about the `sysstat` command, refer to "Displaying filer statistics" on page 374.

Volume concepts

Section contents

This section covers the following topics:

- Understanding volumes
- Determining the number of volumes to use
- Planning a multiple volume configuration
- Installing a foreign volume

Understanding volumes

Data on the filer is organized in *volumes*. A volume is an independent file system with its own RAID groups.

The initial configuration for new filers running Data ONTAP 5.0 or later includes one 2- to 14-disk volume (a root file system). All remaining disks are spares.

Volume naming conventions: You choose the volume names. The names must follow these naming conventions:

- begin with either a letter or an underscore (_)
- contain only letters, digits, and underscores
- contain no more than 255 characters

The root volume: Each filer must have a *root volume* to boot. The root volume of a filer (configured with either a single volume or multiple volumes) is the volume whose */etc* directory is used by the filer for configuration information.

The filer uses two naming conventions to indicate the root volume:

- */vol/vol0*
- /

In the */vol/vol0* convention

/vol	Indicates that the next part of the path, such as *vol0* in this example, is a volume name.
/vol0	Indicates the default name of the root volume for a filer initially installed with Data ONTAP 5.0 or later. You can change this name using the `vol rename` command.

For backward compatibility with filers running releases of Data ONTAP earlier than 5.0, you can use the / naming convention to indicate the root volume on a filer that you upgraded to Data ONTAP 5.1 that contains a previously existing file system.

Mounting volumes: On filers configured with multiple volumes, mounting / is equivalent to mounting */vol/vol0/*, (where *vol0* is the root directory of the root volume). Paths that begin with / (for example, */etc*) refer to directories on the root volume.

Note

/vol is not a directory—it is a special virtual root path under which the filer mounts other directories. You cannot mount */vol* to view all the volumes on the filer; you must mount each filer volume separately.

In mount requests and server commands, prefix the path names of the volumes and directories you want to mount using the convention */vol/volume_name/directory.*

Example: For example, */vol/users/home/cheryl* is a directory called */home/cheryl* in a volume named *users*.

Determining the number of volumes to use

Whether you should use the default single volume configuration or create additional volumes depends mainly on the storage capacity of the filer.

Use of a single volume: If you want the filer to be configured with a single volume, you do not need to do any further volume configuration after you complete the initial setup.

If your filer doesn't have a large number of disks, a single-volume configuration is probably all you need. You can create additional volumes in the future.

Use of multiple volumes: There are several factors to consider before deciding to create and use multiple volumes:

Configuring with multiple volumes aids in the administration of filers that have large storage capacities, enabling you to

- Perform administrative and maintenance tasks, for example, backup and restore, on individual volumes rather than on a single, large file system.
- Set `option` command values, for example, `snap sched`, `raidsize`, `minra`, `no_atime_update`, and so on, differently for individual volumes.

- Take individual volumes off-line, for example, to perform administrative tasks on their file systems or associated RAID groups, while the other volumes remain on-line, without interrupting the availability of the data on them.

Limitations of configuring with multiple volumes include

- The filer's storage space is partitioned.
- Additional administrative overhead is introduced, for example, defining export points.
- You can expand but not concatenate, shrink, or split volumes.
- You cannot perform a local copy of a volume's contents; you must use `dump` and `restore` or `ndmp copy`.

Planning a multiple volume configuration

Before you configure a filer with multiple volumes, you must decide on the number and sizes of volumes you want to configure.

When deciding the number and sizes of volumes you want to configure, keep the following considerations in mind:

- Configuring with more volumes
  - provides more flexible quota and snapshot configuration
  - requires more export points
- Configuring with larger volumes increases the time needed to restore a volume from tape.
- The maximum number of volumes per filer is 23.
- The maximum *recommended* volume size is 250 GB.

For instructions about specific volume configuration procedures, see "Volume management tasks" on page 91.

Installing a foreign volume

You can remove an entire volume from one filer and install it in another, which makes the moved volume a *foreign* volume to the filer.

Example: For example, you might want to move a volume to a different filer to

- replace the volume's disk shelf with one that has a greater storage capacity
- gain access to the files on a dead filer

When a filer detects a foreign volume at boot time, it places the foreign volume off-line. You can then bring the foreign volume on-line. For more information about installing and bringing up a foreign volume on-line, refer to “Adding a foreign volume” on page 92.

Procedures for managing disks and volumes

Section contents

This section provides step-by-step procedures for performing many filer management tasks from the command line on a filer administration host or client. It is organized into two main sections:

- Disk management tasks
- Volume management tasks

Alternatively, you can perform these procedures using the FilerView program, which has a graphical interface.

Disk management tasks

About this section

This section contains procedures for managing the filer's disks and RAID groups.

Setting the size of a volume's RAID groups

To set the RAID group size of a volume when you create it, enter the command

`vol create` ***`volume`*** **`-r`** ***`n`***

where ***`volume`*** is the name of the volume and ***`n`*** is the number of disks you want in each RAID group. Every RAID group must contain at least two disks.

Changing the size of a RAID group after creating it

To change the size of a RAID group, enter the command

`vol volume options raidsize` ***`size`***

where ***`size`*** is the number of disks you want in the RAID group.

Note

You can only change the size of the last RAID group in a volume. You cannot change the size of RAID groups after they have been filled.

Installing new disks

New disks are ones that have never been used. Complete the following steps to install new disks.

Step	Action
1	Install one or more disks according to the hardware guide for your disk shelf. The system displays a message confirming that one or more disks were installed, then waits 15 seconds as the disk(s) is turned on. The system recognizes the disks as a hot spare disks. **Note** If you added multiple disks, they might require 25–40 seconds to come up to speed as the system checks the device addresses on each adapter and returns to normal operation.

Step	Action
2	Type the following command: **`sysconfig -r`**
3	Check the `sysconfig -r` output to verify that the new disk has been added.

Adding disks to volumes

To add new disks to a volume, follow these steps:

1. Enter the command

   `sysconfig -r`

   to verify that there are spare disks available for you to add.

2. Add the disks to a volume by entering the command

   `vol add` ***`volume ndisks`***

   where ***`volume`*** is the name of the volume and ***`ndisks`*** is the number of disks you want to add to the volume.

Refer to the *na_vol*(1) man page for details about adding disks to volumes.

Removing a failed disk

When a raid disk has failed, you need not enter any commands—just remove the failed disk from the disk shelf.

Removing a hot spare disk

Complete the following steps if you want to swap disks because you want to use a hot spare disk in another filer:

Step	Action
1	Type the following command and use the output to determine the disk number: **`sysconfig -r`**

Step	Action
2	Type the following command to spin down the disk, replacing *disk_name* with the name of the disk from the output in Step 1: **`disk remove disk_name`** After the disk stops spinning, the disk is ready to be removed.
3	Remove the disk from the filer following the instructions in the hardware guide for your filer model. File service resumes 15 seconds after you remove the disk.

Removing an active file system disk

Complete these steps if you want to remove a disk because it is logging excessive errors.

Step	Action
1	Type the following command and use the output to determine the disk number: **`sysconfig -r`**
2	Type the following command to fail the disk, using the disk name from the output in Step 1 in place of *disk_name*: **`disk fail disk_name`** **Note** The `disk fail` command permanently marks a disk as failed. You cannot reuse the disk; you must replace it. After the `disk fail` command, the system operates in degraded mode, which means that a disk is missing from the RAID group.
3	Remove the disk from the filer following the instructions in the hardware guide for your filer model. File service resumes 15 seconds after you remove the disk.

Volume management tasks

This section contains procedures for configuring and managing volumes.

Note

Although you can expand volume sizes and the RAID groups assigned to them after you create an initial multiple volume configuration, you cannot split or shrink volumes after you create them.

Creating volumes

You can create up to 23 volumes on a filer. Each volume must contain at least two disks.

To create a new volume, at the system prompt enter

`vol create` *`newvol n`*

where ***`newvol`*** is the name for the new volume and ***`n`*** is the number of disks to use. You must have at least ***`n`*** spare disks available.

After creating a new volume

After you create a new volume on a CIFS filer, you must create shares that refer to the new volume to enable clients to access it.

After you create a new volume on an NFS filer, you must

1. Update the system */etc/exports* file.
2. Run `exportfs`.
3. Add the appropriate mount point information to the */etc/fstab* or */etc/vfstab* file on clients that mount volumes from the filer.

Adding disks to a volume

To add more disks to an existing volume, enter

`vol add` *`vol n`*

where ***`vol`*** is the name of the volume and ***`n`*** is the number of disks to be added.

Monitoring volume status

To determine volume status, such as size, options, disk assignments, and so on, enter

vol status *volume*

To view the RAID group and individual disk information for a particular volume, enter

vol status -r *volume*

To view the RAID group and individual disk information for all volumes, enter

vol status -r

Setting volume options

To set various volume options, enter

vol options *volume option value*

Making a volume inactive

To remove a volume from active use upon next reboot, enter

vol offline *volume*

Reactivating an off-line volume

To reactivate an off-line volume, enter

vol online *volume*

Adding a foreign volume

To add a foreign volume, that is, a volume that was previously installed on another filer, you move the disks that contain the volume from the old filer to the destination filer.

Complete the following steps to add a foreign volume.

Step	Action
1	Follow the instructions in the hardware guide to remove the disks from the old filer.
2	Turn off the destination filer and install the disks in the destination filer's disk shelf.

Step	Action
3	Turn on and boot the destination filer. **Results:** When the destination filer boots, it places the foreign volume off-line. If the foreign volume has the same name as an existing volume on the filer, the filer renames it *volume_name(1)*, where *volume_name* is the original name of the volume. **Caution** If the foreign volume is incomplete, repeat Steps 1 and 2 to add the missing disks. Do not try to add missing disks while online—doing so will cause them to become hot spare disks.

4	If the filer renamed the foreign volume because of a name conflict, type the following command to rename the volume:
	
	`vol rename oldname newname`
	
	Example: The following command renames the volume *vol0(1)* to *vol1*:
	
	`vol rename vol0(1) vol1`

5	Type the following command to bring the volume on-line in the new filer, replacing *volume_name* with the name of the volume: `vol online volume_name`
6	Enter the following command to confirm that the added volume came on-line: `vol status`

Destroying a volume

To destroy a volume, turning its disks back into spare disks, follow these steps.

Step	Action
1	To deactivate the volume, enter `vol offline volume`
2	Enter the **`reboot`** command to reboot the filer.

Step	Action
3	To destroy the volume, enter **`vol destroy volume`**

Renaming a volume

To rename a volume, follow these steps.

Step	Action
1	Enter **`vol rename oldvolume newvol`**
2	Update the */etc/exports* file and run `exportfs`.
3	Update any CIFS shares that refer to the volume.

Handling volume failures

A volume might fail because of an *inconsistent* directory or a double-disk failure. If the system does not reboot after a volume failure, take the volume off-line, as follows:

1. Boot the filer from a system boot diskette into maintenance mode.
2. Use the `vol` command to take the failed volume off-line.

If the failed volume was the filer's root volume, you must designate another volume as the new root volume.

Network Administration 4

Chapter contents

This chapter discusses the topics listed in the following table.

Read...	To learn about...
"Working with large files" on page 96	How the filer supports large files.
"Using SNMP" on page 97	Using SNMP on the filer from a UNIX host, including information about the Network Appliance custom Management Information Base (MIB).
"Host name resolution" on page 100	How the filer resolves names.
"Routing" on page 107	How the filer routes packets using its routing table.
"Using ifconfig to configure an interface" on page 111	How to configure a network interface.
"Using vif to configure a virtual interface" on page 116	How to configure a virtual interface.
"ATM PVCs" on page 118	How to work with ATM (Asynchronous Transfer Mode) PVCs (permanent virtual circuits).

Working with large files

About large files

You can have large files on your filer. The maximum size of a large file is determined by the smaller of the following two values:

- maximum file size supported by your server
- maximum file size supported by your clients

However, if the maximum file size on your client is larger than the maximum file size on your server and you enter a command to display the volume size on your server, the file size is displayed incorrectly. In this case, the maximum file size appears as 2 GB, even though the file on the client is larger than 2 GB.

Caution

If you create a file that exceeds 64 GB, you cannot revert the filer to Data ONTAP 5.0.1 or earlier and access that file.

Software requirements

To use large files, ensure that your system meets the following requirements.

For...	Make sure you have...
Filer operating system	Data ONTAP 5.1 or later
NFS version 3	enabled
UNIX host and clients	Solaris 2.6 or later
Windows NT host and clients	4.0 or later

How to enable NFS

To enable NFS version 3, use the `options` command to set the `nfs.v3.enable` option to On.

Using SNMP

Data SNMP provides

For diagnostic and other network management services, the filer supports the SNMP MIB-II specification. Based on SNMP version 1, this specification provides data about the following MIB-II groups:

- system
- interfaces
- address translation
- IP
- ICMP
- TCP
- UDP
- SNMP MIB-II

SNMP commands enable users to specify up to eight communities and trap notifications for up to eight management stations.

Command to configure 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 in the root volume is 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 supported by Network Appliance

SNMP commands that Network Appliance supports are described in detail in the *na_snmp*(1) man page. The following paragraphs provide brief explanations of these commands:

- `snmp contact 'jdoe@abc.com 415-555-1212'`

  Sets the email address and telephone number of the person responsible for the filer. You can include the person's full name, but an email address and a telephone number enable Network Appliance Technical Support to contact the right person after receiving an automatic email message through autosupport from your site concerning your filer status.

- `snmp location 'ABC corporation, engineering lab'`

Sets the physical location of the filer. This value is returned by the SNMP agent.

- `snmp community add ro private`

  Creates 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`

  Makes the system `snmp-mgr1` the recipient of all SNMP traps from the filer.

- `snmp init 1`

  Initializes the SNMP daemon with the values specified with `snmp` commands. This command also sends the SNMP cold start and links up or down traps, as appropriate, to any trap hosts that were previously registered using the `snmp traphost` command. This command should be the last SNMP command in the filer's */etc/rc* file.

About the Network Appliance custom MIB

The Network Appliance custom MIB provides detailed information about many aspects of filer operation. The current version of the Network Appliance custom MIB is 1.1.0. It contains objects that help you manage multivolume features.

Where to get the MIB: You can obtain the custom MIB from the Data ONTAP CD, or from the NOW Web page at *http://www.now.netapp.com*. To locate the MIB on the CD, read the *intro.txt* file.

Installing the MIB: Install the MIB file on your network management workstation according to the installation procedure for your workstation, so that your workstation can obtain information from the filer about the objects that are part of the MIB.

Using deprecated single-volume objects: *Single-volume objects* are *deprecated*, but you can use them for single-volume systems as before. If you are using a single-volume filer, you do not need to make changes to use the new MIB.

Finding multivolume objects: The descriptions of the deprecated single-volume objects contain the names of the corresponding new *multivolume objects*. For multivolume objects, use the new objects rather than the deprecated ones.

About MIB group contents

The top-level groups in the custom MIB and the information they contain are described in the following table.

Group name	Contents
`product`	Product-level information, such as the software version string and the system ID.
`sysstat`	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 using disk quotas” on page 296.
`filesys`	Information related to the file system, including the equivalent of the `maxfiles` and `df` commands, and some of the information from the `snap list` command. For more information about `maxfiles`, see “Increasing the maximum number of files” on page 307. For more information about the `df` command, see “The df command” on page 308. For more information about snapshots, see Chapter 10, “Snapshots.”
`raid`	Information about RAID equivalents to the `sysconfig -r` output. For more information about `sysconfig`, see Chapter 15, “System Information and Performance.”

Host name resolution

How the filer resolves host names

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 in the root volume.

Name resolution search

By default, first the filer tries to resolve host names locally by searching the */etc/hosts* file in the root volume and in the */etc/nsswitch.conf* file in the root volume. If it cannot resolve the host name, the filer tries NIS, if NIS is enabled. If the filer still cannot resolve the host name, the filer requests services from a DNS server, if DNS is enabled. You can specify any or all of the resolution methods.

Default search order

The following table shows the default search order for each map.

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

Specifying a search order

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

```
map: service ...
```

Parameter	Description
`map`	One of the following maps or databases: `hosts`, `passwd`, `netgroup`, `group`, or `shadow`.

Parameter	Description
`service`	One or more of the following: ◆ files for local files in the */etc* directory in the root volume ◆ dns for DNS ◆ nis for NIS

Example search order

You can list services in the order in which you want the filer to contact the services. For example, the following file instructs the filer to contact first NIS for hosts, then DNS, and finally local files in */etc* in the root volume. For passwords, the contact order is NIS, then local files in */etc* in the root volume.

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

When the filer resolves a host, the search stops.

Note

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

Using the /etc/hosts file for host name resolution

The filer can use the */etc/hosts* file in the root volume to resolve host names used in the */etc/rc*, */etc/syslog.conf*, */etc/dgateways*, */etc/exports*, */etc/netgroup*, and */etc/hosts.equiv* root volume files. If you do not use any host names, you do not need */etc/hosts*.

The sample setup session shown in the *Getting Started Guide* generates the following root volume */etc/hosts* file:

```
#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
```

By default, the filer reads the */etc/hosts* file in the root volume 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 doesn't take effect until you reboot the filer and execute the appropriate `ifconfig` command 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 you put files first in the */etc/nsswitch.conf* file in the root volume: The filer looks up host addresses in */etc/hosts* in the root volume 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* in the root volume, when using DNS, keep only a minimum number of entries in */etc/hosts*.

If you use NIS: You can do one of the following actions:

- Modify the Makefile of the NIS master to copy the NIS master's */etc/hosts* file from the root volume to the filer when it is changed.
- Have the filer use NIS directly.

Example

You can put the following line 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
```

Substitute the name of each filer in the "for filer in..." list in place of toaster1, toaster2, and so on.

Using DNS

The filer includes DNS client capabilities to 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 in the root volume every time you add a new host 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.)

Note

To prevent naming inconsistencies, Network Appliance recommends that when you enable DNS, you use only the default */etc/hosts* file in the root volume.

Enabling DNS during setup: At `setup`, if you enter **y** in response to the following prompt, `setup` prompts you for a DNS domain name, as follows:

```
Do you want to run DNS resolver [n]: y
Please enter DNS domain name []:
```

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

```
nameserver ip_address
```

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

Enabling DNS without using setup: If you didn't start DNS during `setup`, you can start DNS using the following procedure.

Step	Action

1	Create a */etc/resolv.conf* file in the root volume. 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`

Step	Action

2	Edit the */etc/rc* file in the root volume 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	Reboot the filer or enter the commands at the filer prompt. **Result:** DNS is now enabled. You no longer have to update the filer's */etc/hosts* file in the root volume every time you add a new host to the network, unless you specify files first in the */etc/nsswitch.conf* file in the root volume.

Disabling DNS

To disable DNS, enter the following command or put the command in the */etc/rc* file in the root volume to make the change permanent:

```
options dns.enable off
```

Changing your DNS domain name: To change your DNS domain name, enter the following command or put the command in the */etc/rc* file in the root volume to make the change permanent:

```
options dns.domainname domainname
```

Using NIS

The filer includes NIS client capabilities to query NIS servers for host-name-to-IP-address and IP-address-to-host-name translation services.

Note

Because the `nsswitch.conf` file already enables 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 in the root volume. Any existing + or - entries are ignored.

NIS maps the filer uses

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 setup: During `setup`, the following prompt appears:

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

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

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

Enter an NIS domain name.

Enabling NIS without using setup: If you didn't start NIS during `setup`, you can start NIS using the following procedure.

Step	Action

1	Edit the */etc/rc* file in the root volume to make sure that the option specifying the NIS domain name is set and the option to enable NIS is On. Insert lines similar to the following:
	`options nis.domainname netapp.com`
	`options nis.enable on`

2	Enter the `options` commands from the command line or reboot the filer. **Result:** Options entered only on the command line are not saved if you reboot. NIS is now enabled. You no longer have to update the filer's */etc/hosts* file in the root volume every time you add a new host to the network, unless you specify files first in the */etc/nsswitch.conf* file in the root volume.

Disabling NIS: To disable NIS on a running system, enter the following command, or put the command in the */etc/rc* file in the root volume to make the change permanent:

```
options nis.enable off
```

Changing your NIS domain name: To change the domain name on a running system, enter the following command or put the command in the */etc/rc* file in the root volume to make the change permanent:

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

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

```
ypwhich
```

The `ypwhich` command has no options.

Routing

About filer 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. It can, however, route its own packets.

Routing table on the filer

For routing its own packets, the filer relies on the default route and explicit routes. Typically, 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:

```
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 you must enter explicit information into the filer's routing table, use the `route` command. See the *na_route*(1) man page about 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 in the root volume.

Specifying default routers

One default router is specified during `setup`. You can, however, add potential default routers at any time to the */etc/dgateways* file in the root volume. For each added router, you should also specify the metric, which is a number indicating the route preference for the router. The highest preference is 1, which is also the default preference for the router specified during `setup`. The lowest preference is 15. For information about the format of the */etc/dgateways* file, refer to "The /etc/dgateways file" on page 109.

Using the routed daemon to manage multiple routers

To help manage multiple routers and to enable you to create redundant routing schemes, the filer runs the `routed` daemon, a simple routing daemon, 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 `routed` daemon selects the one with the highest preference to use as the filer's default router.

However, the filer doesn't rely on the `routed` daemon to construct the routing table. The function of the filer's `routed` daemon is 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 the `routed` daemon Off and On with the `routed` command.

To turn the `routed` daemon Off, enter

```
routed off
```

To turn the `routed` daemon back On, enter

```
routed on
```

Note

If you turn the `routed` daemon Off by editing */etc/rc* in the root volume, 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. An example of the `routed status` display is as follows:

```
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
```

The `routed status` display shows the following information:

- whether RIP snooping is active (On or Off)

- the current list of default gateways
- the metrics of the default gateways (1 through 15)
- the state of the gateways (ALIVE or DEAD)
- the last time each gateway was heard from

The /etc/dgateways file

The */etc/dgateways* file in the root volume is the configuration database for the `routed` daemon. From 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).

#Gateway	Metric
192.9.200.10	1
eng_gateway	2

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 *toaster-e1* interface to send packets in response to NFS requests received on the *toaster-e1* interface.

This way of handling NFS-over-UDP requests enables 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 might be returned through a different path than you might expect from an examination of the IP routing table using `netstat -r`. This scheme generally works well, although it can result in different routes than expected if your environment contains one-way routes. For example, the IP packets might 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 routes 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, -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 uses the first interface it finds with that number to send the responses.

Using ifconfig to configure an interface

About the ifconfig command

The */etc/rc* file in the root volume contains `ifconfig` commands to configure network interfaces at system boot. You can also manually use the `ifconfig` command when the system is operating.

The ifconfig command syntax

The `ifconfig` command syntax is as follows:

```
ifconfig <interface>
    [ [ alias | -alias ] <address> ] [ up | down ]
    [ netmask <mask> ] [ broadcast <address> ]
    [ mtusize <size> ] [ mediatype <type> ]
    [ trusted | untrusted ] [ wins | -wins ]
    [ partner | -partner <address> ]
```

Reasons to use the ifconfig command

You use the `ifconfig` command for the following purposes:

- changing the interface's IP address, network mask, or broadcast address
- setting the media type on an Ethernet interface
- setting the maximum transmission unit (MTU)
- configuring the interface up or down
- specifying the partner IP address in a cluster

Changing the interface's IP address, network mask, or broadcast address

The following examples show 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 might be different, depending on your filer model. For information about interface naming conventions, refer to "Naming conventions for network interfaces" on page 34.

Setting the media type on an Ethernet interface

The following example show configuring an Ethernet interface on a filer:

```
ifconfig e1 mediatype tp
```

The media types you can use depend on the type of Ethernet card. The possible types you can enter in the `ifconfig` command are the same as those you can select when running `setup`. They are described in the following table.

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 Auto-Negotiation
1000fx	Gigabit Ethernet

Setting the maximum transmission unit (MTU)

The following example shows setting the MTU for an FDDI interface:

```
ifconfig f0 mtusize 1500
```

The following table lists the default MTU sizes.

Interface	Default MTU size
FDDI	4352
Ethernet	1500
Gigabit Ethernet	1500
ATM	9188

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 following command:

```
netstat -i
```

Configuring the interface up or down

- The following example illustrates how to configure interfaces up and down:

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

Specifying the partner IP address in a cluster

Within a cluster, each filer must have one or more interfaces configured with its partner's IP address. The filer uses the interface configured with its partner IP address to serve data while in takeover mode. The following terms relate to IP addresses in clusters:

Term	Definition
partner IP address	One or more IP addresses assigned to a clustered filer's partner.
dedicated standby interface	An interface in a clustered filer that is dedicated to a partner IP address. This interface is idle when the cluster is operating in normal mode. It becomes active when the filer in which it is installed takes over for its partner.
shared interface	An interface in a clustered filer that is shared by the IP addresses of the filer in which it is installed and a partner IP address. The partner IP address is idle when the cluster is operating in normal mode. It becomes active when the filer in which it is installed takes over for its partner.

Partner IP addresses can be configured to use dedicated standby interfaces, or they can be configured to shared interfaces.

Partner IP addresses must be configured to use interfaces on the same subnetwork as the partner.

Dedicated interface example: The following example shows how to configure two filers to assume each other's identities during takeovers using dedicated interfaces.

Filer 1	Filer 2
Name: Toaster1	Name Toaster2
Interface e1 address: 198.9.200.28	Interface e1 address: 198.9.200.38
Interface e2 partner address: 198.9.200.38	Interface e2 partner address: 198.9.200.28

The following commands entered on the toaster1 console:

- configure interface e1 on toaster1 to use 198.9.200.28 as its own IP address
- configure interface e2 on toaster 1 to use 198.9.200.38 as the partner IP address for toaster2's e1 interface during takeover:

```
ifconfig e1 198.9.200.28
ifconfig e2 partner 198.9.200.38
```

The following commands entered on the toaster2 console:

- configure interface e1 on toaster2 to use 198.9.200.38 as its own IP address
- configure interface e2 on toaster2 to use 198.9.200.28 as the partner IP address for toaster1's e1 interface during takeover:

```
ifconfig e1 198.9.200.38
ifconfig e2 partner 198.9.200.28
```

Shared interface example: The following example shows how to configure two filers to assume each other's identities during takeovers using shared interfaces.

Filer 1	**Filer 2**
Name: Toaster1	Name Toaster2
Interface e1 address: 198.9.200.28	Interface e1 address: 198.9.200.38
Interface e1 partner address: 198.9.200.38	Interface e1 partner address: 198.9.200.28

The following commands entered on the toaster1 console:

- configure interface e1 on toaster1 to use 198.9.200.28 as its own IP address
- configure interface e1 on toaster 1 to use 198.9.200.38 as the partner IP address for toaster2's e1 interface during takeover:

```
ifconfig e1 198.9.200.28
ifconfig e1 partner 198.9.200.38
```

The following commands entered on the toaster2 console:

- configure interface e1 on toaster2 to use 198.9.200.38 as its own IP address
- configure interface e1 on toaster2 to use 198.9.200.28 as the partner IP address for toaster1's e1 interface during takeover:

```
ifconfig e1 198.9.200.38
ifconfig e1 partner 198.9.200.28
```

Edit /etc/rc file to make changes persistent after reboot

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

Viewing interface configuration information

The following table illustrates examples of how to use the `ifconfig` command to view interface configuration information.

Description	Syntax
Show the current configurations of all network interfaces	`ifconfig -a`
Show the current configuration of a specific network interface	`ifconfig` *`interface`* **Example:** Enter the following command to show the current configuration of interface e0: `ifconfig e0`

na_ifconfig man page has more information

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

Using vif to configure a virtual interface

About virtual interfaces

Using your existing network infrastructure, you can create an EtherChannel virtual interface by grouping together multiple 10/100Base-T Ethernet links into a single channel, or "trunk." This results in improved Ethernet performance and recovery from link failures.

Note

Virtual interface trunking is supported only on Ethernet cards. Quad-port 10Base-T cards are not supported.

Use the vif command to manage trunks

You use the `vif` command for the following purposes:

- to create a virtual interface
- to destroy a virtual interface
- to display statistics about a virtual interface

Creating a virtual interface

The `vif create` command creates a virtual interface. You can give the new virtual interface any name that is not already in use by using the `ifconfig` command to see what virtual interface names have been used. See "Using ifconfig to configure an interface" on page 111 for more information about how to list interfaces.

Example: The following example shows creating a virtual interface named vif1 on a filer:

```
vif create vif1 e0 e7a e6b e8
```

The interfaces grouped into a trunk do not have to be on the same network card. Be sure that all the cables from a channel, for example, channel A, on the Ethernet switch connect to the corresponding channel ports on the filer.

Note

All Ethernet interfaces forming a trunk should be the same media type if you configure them manually. If you use Auto-Negotiation, all interfaces forming the trunk must support Auto-Negotiation. Network Appliance recommends using the `100tx-fd` media type in filer trunks.

Check the documentation that comes with your Ethernet switch or router to see whether you need to configure the filer Ethernet interfaces to be half-duplex or full-duplex.

When you create a virtual interface to group together multiple interfaces, only one of the interfaces, called the "primary interface," is listed in the `route` table.

After you create a trunk, use `ifconfig` to perform any further operations that affect the virtual interface.

Eliminating a virtual interface

The `vif destroy` command eliminates an existing virtual interface. You must configure the virtual interface down using the `ifconfig` command before entering the `vif destroy` command.

Example: The following example shows eliminating a virtual interface named vif1 on a filer:

```
vif destroy vif1
```

Displaying statistics about a virtual interface

The `vif stat` command displays statistics about a virtual interface. You can specify the time interval, in seconds, at which the statistics are displayed. By default, the statistics are displayed at one-second intervals. The statistics are the number of packets received and transmitted on each Ethernet interface that makes up the virtual interface.

Example: The following example shows displaying statistics about vif1 on a filer:

```
vif stat vif1
Virtual interface (trunk) vif1
    e5d                      e5c             e5b             e5a
In      Out         In      Out     In      Out     In      Out
8637076 47801540        158     159     7023083 38300325 8477195 47223431
1617    9588        0       0       634     3708    919     5400
1009    5928        0       0       925     5407    1246    7380
1269    7506        0       0       862     5040    1302    7710
1293    7632        0       0       761     4416    964     5676
920     5388        0       0       721     4188    981     5784
1098    6462        0       0       988     5772    1003    5898
2212    13176       0       0       769     4500    1216    7185
1315    7776        0       0       743     4320    530     3108
```

ATM PVCs

About 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. PVCs are static; for each destination, you must 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 on the filer, use the following filer command:

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

Variable	Description
`hostname`	Name or IP address of the remote host.
`iface`	Name of the ATM interface.
`vpi`	VPI (virtual path identifier); this must be 0.
`vci`	VCI (virtual channel identifier); this number must have the following properties: ◆ It must not be in use on your filer. ◆ It must be less than 1,024. ◆ It must obey the limits of the destination host and interconnecting devices.
`aal`	ATM adaptation layer type. It must be 4 or 5, and should be the AAL type supported by the destination host, which is typically 5. The default is 5.

Variable	Description
encap	Encapsulation type. Specify one of the following encapsulation types: ◆ 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) It should also be the same as the encapsulation type used by the destination host. If the encapsulation type is llc_bridged_8023, you must include *addr.*
addr	Six-byte colon-separated destination MAC address.

Establishing an incoming PVC on the filer

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

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

Use the variables as described in the previous section.

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, use the following command:

`arp` *`hostname`*

where *hostname* is the name of a host.

To display information about all the current PVCs, use the following command:

```
arp -a
```

A display similar to the following is displayed:

```
arp -a

aye (272.16.1.1) at 0:0:3b:80:36:88
bdellium (272.16.1.2) at 0:0:3b:80:36:d4
ctenophore (272.16.1.3) at 0:0:3b:80:36:d7
mnemonic (272.16.1.4) at 0:0:3b:80:2e:5b
oedipus (272.16.1.5) at 8:0:20:80:8d:47
psych (272.16.1.6) at 8:0:20:80:8d:d3
xenon (272.16.1.7) at 8:0:20:81:d9:47
yttrium(272.16.1.9) 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
```

Deleting a PVC

To delete an outgoing PVC entry, use the following command:

```
arp -d hostname
```

where

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

To delete an incoming PVC for a remote host, use the following command:

```
arp -x iface vpi vci
```

where

iface is the name of the interface and *vpi* and *vci* have the values of the VPI and VCI of the PVC to be deleted for the specified interface.

Sharing Files between NFS and CIFS Users 5

About file sharing

This chapter describes how the filer works with NFS and CIFS clients simultaneously. Because these clients interact with a file server differently, you need to understand how the read and write operations performed by one client affect the operations performed by the other client.

What this chapter discusses

This chapter discusses the following topics.

Read...	To learn about...
"File-locking interactions" on page 122	CIFS and NFS file locks.
"Managing symbolic links for CIFS access" on page 123	How the filer supports CIFS access to symbolic links.
"NFS and CIFS use of the read-only bit" on page 126	How NFS and CIFS use the per-file read-only bit.
"Naming files used by both NFS and CIFS" on page 128	How the filer manages names of files used by both NFS and CIFS users.
"Timing of directory conversion for CIFS access to NFS-created files" on page 130	How much time the filer takes to convert directories created by NFS clients to be used by CIFS clients.

What this chapter does not discuss

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 were exported, and CIFS users *share* directories to which they were given access in the Access Control List (ACL).

  For more information about these procedures, refer to the documentation for the clients' operating systems.

- Filer administration procedures for individual file-sharing protocols.

  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

About this section

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 client 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.

Types of clients: 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.

Types of locks: 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 a 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

About symbolic links

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.

Controlling access to symbolic links

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, which is the default setting, if the object being accessed by a CIFS client is an absolute or relative symbolic link, 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.

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. Therefore, 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, you might want to disable symbolic links for CIFS.

Syntax

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

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

To enable symbolic links for CIFS, use

```
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 */u/users/charlie* directory to the */home/charlie* directory on the filer. You do so by specifying symbolic link redirection mappings in a text file named */etc/symlink.translations*.

The format of the */etc/symlink.translations* file is

```
Map link target
```

where both *link* and *target* are absolute symbolic link path names.

For example, the entry

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

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

Preventing symbolic link cycling

You can 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. Therefore, a simple recursive descent of the tree goes deeper and deeper until the maximum path length is reached. At that point an error is returned. For example, if you used Windows Explorer to search for files in such a cyclic directory, the same files 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, which is the 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

About read-only bits

The filer, along with MS-DOS and 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 Windows systems, requires that the read-only bit reflects whether the file is writable.

How NFS treats the read-only bit

The following list 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 preceding paragraph, a file with the read-only bit turned On 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 superuser.

How the filer tracks the NFS or CIFS client read-only bit

Whenever the read-only bit is turned On by a client, even if it was already On before the client did so, the filer tracks 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, renaming the file is not allowed. This is because file systems on MS-DOS and Windows systems do not allow renaming a file whose read-only bit has been set. NFS deletes follow NFS conventions. That is, deletes are allowed if the user has write permission in the parent directory.

- 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. This is because file systems on UNIX systems allow removing or renaming a file that has no write permission bits set.
- If the filer is using UNIX-style security, CIFS clients are also allowed to delete a file with the read-only bit set. This is required for compatibility with standard UNIX source control programs, such as RCS.
- If the filer is configured with PC-style security, the read-only bit is enforced.

Naming files used by both NFS and CIFS

About file naming conventions

This section discusses how different kinds of clients can obtain access to the same files despite different file naming conventions.

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 Windows operating systems.

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 no longer has 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 sees 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 is 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 see the file names as follows:

- On NFS clients, one file is displayed as *Spec.txt* and the other is displayed 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

About CIFS access to NFS files

If you license both NFS and CIFS protocols, your filer supports access through 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. Additionally, CIFS maintains certain attributes for each file 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.

One-time access conversion

If you add CIFS access to files that were created by NFS clients, the first time a directory is accessed through CIFS, the filer automatically converts 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 by accessing any files or directories contained in that directory.

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

After a directory is converted, it does not have to be converted again.

Conversion time

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 immediately triggers 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.

NFS Administration 6

NFS administration covers four areas

This chapter has four sections

This chapter is divided into four sections that contain the topics described in the following table.

Read...	To learn about...
"Section A: Managing NFS Exports" on page 134	Exporting filer volumes and directories to clients so that users can access them.
"Section B: Configuring a filer for (PC)NFS" on page 149	Setting the filer up to support (PC)NFS clients so that users on those clients can access filer volumes and directories.
"Section C: Configuring a filer for WebNFS" on page 156	Setting up the filer to serve Web pages to browsers that support the NFS transport.
"Section D: Displaying NFS statistics" on page 160	Displaying and interpreting NFS statistics.

Section A: Managing NFS Exports

Section contents

This section describes how to manage NFS exports. It includes the topics listed in the following table.

Topic	Description
"Introducing the /etc/exports file" on page 135	Describes the format for entries in the */etc/exports* file.
"Rules for exporting volumes and directories" on page 137	Describes rules you should follow when creating entries in the */etc/exports* file.
"Default /etc/exports entries" on page 140	Describes the default entries that `setup` places in the */etc/exports* file.
"Restricting access to volumes and directories" on page 141	Describes how to restrict access to exported volumes and directories, including ways to use the `-access`, `-root`, `-ro`, and `-rw` options with */etc/exports* file entries.
"The exportfs command" on page 143	Describes the syntax for the `exportfs` command and how to use the command to activate and deactivate exports.
"The /etc/netgroup file" on page 145	Describes how to define groups of hosts, subnets, and other groups to make entries in the */etc/exports* file easier to manage.
"Exporting to subnets" on page 147	Describes how to make entries in the */etc/exports* file more manageable by exporting volumes and directories to subnets rather than to individual clients.

Introducing the /etc/exports file

/etc/exports controls client access to directories

The */etc/exports* file controls how NFS clients access filer directories. You add entries to the */etc/exports* file for all the directories you want to export.

Format for /etc/exports entries

The format of the entries in the */etc/exports* file is as follows:

```
filer_directory_path export_specification
```

Filer directory path format

The filer directory path specifies which directory is made available to clients. The format is as follows:

`/vol/`*`volume_name/directory`*

Examples: The following lines show examples of filer directory paths:

```
/vol/vol0/home
/vol/users/local/jarnold
```

Export specification determines access privileges

The export specification specifies the privileges that clients have to mount and access the filer directory path. The format is as follows:

`-root=`*`list`*`,-access=`*`list`*`,-ro=`*`list`*`,-rw=`*`list`*

One keyword is required: The keywords `-root`, `-access`, `-ro`, and `-rw` are all optional; however, you must include at least one keyword in an export entry.

What the list variable represents: The *list* variable represents a list that includes one or more

- host names
- netgroup names
- subnets

You can combine elements in an entry: You can combine host names, netgroup names, and subnets in an entry, as shown in the following example:

```
-root=adminhost:administrators,-rw=blender:pcusers:123.45.67.0/24
```

Example 1: exporting default filer volume to administration host:

The following line exports the root directory of the default filer volume to the administration host with root privileges. The administration host can mount the directory, change permissions and ownerships, and create and delete directories and files.

```
/vol/vol0 -root=adminhost
```

Example 2: exporting home directory to administration host and clients: The following line exports the home directory from the default filer volume to the administration host and two clients. The administration host has root privileges and can mount the directory, change permissions and ownerships, and create and delete directories and files. The clients can mount the directory and create and delete directories.

```
/vol/vol0/home -root=adminhost,-rw=blender:mixer
```

Related information

The following table describes related information that you might want to read.

Read...	To learn about...
“Rules for exporting volumes and directories” on page 137	Rules for exporting volumes and directories.
“Default /etc/exports entries” on page 140	The default entries that `setup` places in the */etc/exports* file.
“Restricting access to volumes and directories” on page 141	Ways that you can restrict access to volumes and directories.
“The exportfs command” on page 143	How to activate the exports listed in the */etc/exports* file.

Rules for exporting volumes and directories

Export each volume separately

If the filer has multiple volumes, you must export each volume separately; you cannot export all volumes by specifying */vol* as the exported directory.

Example: The following lines show how to export the volumes on a filer that has three volumes:

```
/vol/vol0 -root=toaster, -access=mixer:blender
/vol/users -root=toaster, -access=mixer:blender
/vol/builds -root=toaster, -access=mixer:blender
```

Nonexample: The following line shows an invalid entry for the */etc/exports* directory for a filer that has three volumes; */vol* cannot be used by itself as a path:

```
/vol -root=toaster, -access=blender:mixer
```

Filer must resolve host names

To export directories to hosts, the filer must be able to resolve host names into IP addresses. Host name resolution can take place

- through DNS name resolution
- by using the */etc/hosts* file on the filer root volume

Refer to "Host name resolution" on page 100 for more information about host-name-to-IP-address resolution.

Cannot restrict access by host

Neither the filer nor NFS provide a way to specify hosts that *cannot* mount the filer. To restrict access to exports, you must export volumes and directories in a manner that specifically includes those hosts that should be allowed to mount them. Exclusion occurs when a host is not specifically included in the list of hosts authorized to mount exported volumes and directories.

You can export ancestors and descendants

The filer permits directories that have exported ancestors to be exported. In many implementations of the UNIX operating system, you cannot export a directory that has an exported ancestor in the same file system.

Example: The following lines show exports that the filer allows:

```
/vol/vol0 -access=adminhost,-root=adminhost
/vol/vol0/home -access=blender:mixer
```

Nonexample: The following lines show exports that are not allowed on some UNIX systems:

```
/home -root=adminhost,-access=blender:mixer
/home/local -root=adminhost,-access=blender:mixer
```

Filer determines permissions by matching longest prefix

The filer uses the longest matching prefix in determining permissions.

In the preceding example

- A client mounting */vol/vol0/home/user1* gets permissions for */vol/vol0/home* because */vol/vol0/home* is the longest matching prefix.
- A client mounting */vol/vol0* gets `-access=adminhost, -root=adminhost` permissions.

Note

Because of the way the filer determines permissions, it makes little sense to give a client greater permissions at a higher level in the file system.

Example: The following lines show an */etc/exports* file that creates a security breech by enabling any host to mount the */vol/vol0* directory while restricting specific hosts to mounting the */vol/vol0/home* directory. In this example, any host can gain access to the */vol/vol0/home* directory by mounting the */vol/vol0* directory:

```
/vol/vol0
/vol/vol0/home -access=bashful:dopey:sleepy
```

Edit /etc/exports after changing volume names

If you rename a volume using the `vol rename` command

- Entries in the */etc/exports* file that refer to the volume become incorrect.
- The in-memory information about active exports gets updated automatically, and clients continue to access the exports without problems.
- The filer displays an error message when the `exportfs -a` command is entered, or when the filer is rebooted.
- Clients display the error message, “Stale NFS file handle,” after the filer is rebooted.

Caution

To ensure that the entries in the */etc/exports* file remain valid, always edit the entries in the file to reflect volume name changes immediately after renaming volumes.

Default /etc/exports entries

/vol/vol0 and /vol/vol0 home are exported by default

By default, the root volume (*/vol/vol0*) and the */vol/vol0/home* directory are exported to the administration host when you run `setup`.

Example of default exportfs file

The default */etc/exports* file contains the following lines; in this example, the name of the administration host is toaster:

```
#Auto-generated by setup Mon Oct 27 14:15:36 PST 1997
/vol/vol0 -access=toaster,root=toaster
/vol/vol0/home -root=toaster
```

The following table explains the contents of the default */etc/exports* file:

/etc/exports entry	Explanation
`/vol/vol0 -access=adminhost, -root=adminhost`	Only the administration host (adminhost) can mount the root directory and modify files in the directory. **Note** On filers with a single volume, you can refer to the root directory without the */vol* prefix.
`/vol/vol0/home -root=adminhost`	The administration host (adminhost) can mount the home directory as root. All other clients can access the home directory to read and write files.

Restricting access to volumes and directories

Use export options to restrict directory access

You can use the export options to restrict access to directories in various ways.

Restricting access to /home

You can restrict access to the */home* directory to particular groups by

- using the `chmod` command to change access modes for the directory
- using the `-rw` or `-access` options in the */etc/exports* file to limit write privilege to specific hosts

The -access option

The `-access` option lists the hosts that can mount exported directories. When you use the `-access` option, only the hosts listed can mount the associated directory.

Syntax: The syntax for the `-access` option is as follows:

`-access=`*`hostname`*`[:...:`*`hostname`*`]`

Limits: There is no limit to the number of host names you can specify with the `-access` option. However, the length of a line in the */etc/exports* file cannot exceed 1,024 characters.

Note

If you cannot fit all the host names in a 1,024-character line, you can use netgroups in place of host names.

The -root option

The `-root` option lists the hosts that can mount exported directories as root. Hosts that mount exported directories as root have full control over the directories and can perform the following operations:

- Create and delete directories and files.
- Change ownership and group associations of directories and files.
- Set access permissions for directories and files.

Syntax: The syntax for the `-root` option is as follows:

`-root=`*`hostname`*`[:...:`*`hostname`*`]`

Limits: You can specify 1 to 256 host names with the `-root` option.

Restrictions: You cannot use netgroup names with the `-root` option.

The -rw option

The `-rw` option lists the hosts that can modify the exported directories; hosts not listed by the `-rw` option have read privilege only.

Syntax: The syntax for the `-rw` option is as follows:

`-rw=`*`hostname`*`[:...:`*`hostname`*`]`

Limits: You can specify 1 to 256 host names with the `-rw` option.

Restrictions: You cannot use netgroup names with the `-rw` option.

The -ro option

The `-ro` option lists the clients that cannot modify the exported directories.

Syntax: The syntax for the `-ro` option is as follows:

`-ro=hostname[:...:hostname]`

Related information

The following table lists additional information that you might find helpful.

Read...	To learn about...
The *na_exports(1)* man page.	More about the */etc/exports* file.
"The /etc/netgroup file" on page 145	How to create and maintain netgroups.

The exportfs command

Using the exportfs command

You use the `exportfs` command to export and unexport volumes and directories. Depending on which options you use, the command exports the volumes and directories listed in the */etc/exports* file or a specific volume or directory.

Syntax

The syntax for the `exportfs` command is as follows:

`exportfs [ -aiuv ] [ -o options ][ filer_directory_path ]`

The following table describes the options.

Option	Description
`-a`	Exports all the entries listed in the */etc/exports* file.
`-u`	Unexports all the entries listed in the */etc/exports* file. **Note** When you use the `-u` option with the `-a` option, all exports are unexported regardless of whether they were created from the */etc/exports* file or with the `-o` option.
`-i`	Exports all the export entries listed in the */etc/exports* file, but ignores the options specified for the entries.
`-v`	Prints each path name as it is exported or unexported.

Option	Description
`-o`	Specifies the options for a volume or directory that you include in the command line. **Example:** To export the */vol/vol/home/terry* directory to the host "mixer" with read/write access, you enter the following command: **`exportfs -o -rw=mixer /vol/vol0/home/terry`** **Note** Volumes and directories that are exported directly rather than through the */etc/exports* file remain exported until canceled using the `-au` option or until the filer is rebooted.

Canceling all exports

To cancel all exports, enter the following command:

`exportfs -au`

Updating exports through /etc/exports

When you make changes to the */etc/exports* file, take one of the following actions to make the changes take effect:

- Run the `exportfs -a` command.

Or

- Reboot the filer.

If you delete entries from the */etc/exports* file when you make changes, take one of the following actions to activate the changes and ensure that deleted exports are deactivated:

- Run the following commands:

```
exportfs -au
exportfs -a
```

Or

- Reboot the filer.

The /etc/netgroup file

The /etc/netgroup file defines groups of clients

The filer */etc/netgroup* file defines groups of clients that the filer uses for checking access permission while processing a mount request.

Syntax

The following line shows the syntax for each line in the */etc/netgroup* file:

```
groupname member-list
```

Limits: Each line in the */etc/netgroup* file is limited to 4,096 characters.

Member-list syntax: Each element in `member-list` is

- another group name
- an entry in the following form:

  (*hostname*, *username*, *domainname*)

  - An element in an entry can be blank, but the commas must be present.
  - When group names are used in the */etc/exports* file, the `username` and `domainname` fields are ignored.
  - When domain names are used, they must be DNS names; NIS names cannot be used.

Restrictions

You cannot use netgroup names with the `-rw` and `-root` options.

Changes take effect immediately

Changes made to the netgroup file take effect immediately.

Example of /etc/netgroups

The following lines show an example of a */etc/netgroups* file:

```
trusted-hosts (adminhost,,)
untrusted-hosts (bashful,,) (dopey,,) (sleepy,,)
all-hosts trusted-hosts untrusted-hosts
```

Example of /etc/exports using netgroups

The following lines show an example of an */etc/exports* file that uses netgroup group names:

```
/vol/vol0 -access=trusted-hosts,root=adminhost
/vol/vol0/home -access=all-hosts,root=adminhost
```

Copy /etc/netgroup when filer doesn't use NIS

If your filer is not configured as an NIS client, the network groups on the filer are not linked with NIS.

Must copy NIS netgroup file: You must copy an existing NIS network group over to */vol/vol0/etc/netgroup* on the filer before the `exportfs` command can use it.

Automating copying with a Makefile: You can modify the Makefile of the NIS master to copy the NIS master's */etc/netgroup* file to the filer when it is changed.

Example Makefile: The following lines of code in the NIS Makefile section for `netgroup.time` copy the */etc/netgroup* file to filers named toaster1, toaster2, and toaster3; substitute the name of your filers in the "for" list, in place of *toaster1*, *toaster2*, and *toaster3*, and add any other filer names to which you want the file copied:

```
@mntdir=/tmp/nac_etc_mnt_$$$$;\
if [ ! -d $$mntdir ]; then rm -f $$mntdir; mkdir $$mntdir; fi;\
for filer in toaster1 toaster2 toaster3 ; do \
mount $$filer:/vol/vol0/etc $$mntdir;\
mv $$mntdir/netgroup $$mntdir/netgroup.bak;\
cp /etc/netgroup $$mntdir/netgroup;\
umount $$mntdir;\
done;\
rmdir $$mntdir
```

Exporting to subnets

About exporting to subnets

You can export a directory to clients on a subnet rather than to individual clients.

Valid export options for subnets

The valid export options in the */etc/exports* file for exporting to subnets are as follows:

`-ro=`*`subnet_address`*`[:`*`subnet_address`*`]...`

`-rw=`*`subnet_address`*`[:`*`subnet_address`*`]...`

`-root=`*`subnet_address`*`[:`*`subnet_address`*`]...`

Format for IP subnet addresses

The subnet address is a dotted IP subnet address and a mask written in the following format:

`dotted_ip/num_bits`

dotted_ip can be

- an IP address (“a.b.c.d”)

or

- an IP subnet
  - a for a class A network
  - a.b for a class B network
  - 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*.

Export to a subnet as you do to a client

You export a directory to a subnet as you do to an individual client, except that you specify a subnet address rather than a full IP address in an export option.

Example 1: root access: To export */vol/vol0/home* on the filer for root access to a client named tooting and all addresses of the form 123.45.67.x with a netmask 255.255.255.0 (24 leading bits), place the following entry in the */etc/exports* file:

```
/vol/vol0/home -root=tooting:123.45.67.0/24
```

Example 2: read/write access: To export */vol/vol0/home* for read and write access to all addresses of the form 123.45.x.y with a 16-bit netmask (255.255.0.0), place the following entry in the */etc/exports* file:

```
/vol/vol0/home -rw=123.45.0.0/16
```

Example 3: equivalent methods for exporting: The following entries in the */etc/exports* file are equivalent. They export */vol/vol0/home* to a client named host1, the specified subnet, and a client named host2.

```
/vol/vol0/home -rw=host1:123.45.67.8/24:host2
```

```
/vol/vol0/home -rw=host1:123.45.67/24:host2
```

Section B: Configuring a filer for (PC)NFS

About configuring a filer for (PC)NFS

Section contents

This section describes how to use the (PC)NFS daemon that is supplied with the NFS protocol. The daemon enables DOS, Windows, Windows NT, OS/2, and Macintosh clients that run (PC)NFS client software to access filer volumes and directories. It includes the topics listed in the following table.

Topic	Describes...
"Introducing the (PC)NFS daemon" on page 150	◆ What the daemon is. ◆ What the daemon does.
"Enabling the daemon" on page 151	◆ How to start the *pcnfsd* daemon. ◆ How to have the filer start the daemon automatically whenever the filer is rebooted.
"Setting up (PC)NFS user entries" on page 152	◆ The requirement that users have entries in the */etc/password* file and, optionally, the */etc/shadow* file on a UNIX host. ◆ How to copy the */etc/passwd* and */etc/shadow* files from a UNIX host to the filer.
"Setting the default umask" on page 154	◆ What the umask is. ◆ What the umask does. ◆ What values you can use in the umask. ◆ The syntax for the `options.pcnfsd umask` command.

Introducing the (PC)NFS daemon

What the daemon does

The filer `pcnfsd` authentication daemon enables clients running a (PC)NFS client to access a volumes and directories on the filer.

(PC)NFS refers to NFS client implementations for DOS, Windows, Windows NT, OS/2, and Macintosh operating systems.

NFS protocol must be licensed

The `pcnfsd` daemon can run on the filer only if the NFS protocol is licensed. If the NFS protocol is not licensed, the following warning messages appear:

```
pcnfsd is NOT running
Cannot use pcnfsd without nfs.
(Use the “nfs on” command to enable nfs.)
```

What (PC)NFS clients can do

Using (PC)NFS, clients can mount volumes and directories from the filer after the user provides the UNIX user name and password for authentication.

Clients must be configured to mount the filer.

Alternate access method for (PC)NFS clients

If you want (PC)NFS clients to be able to access filer volumes and directories but you do not want to run the `pcnfsd` daemon on the filer, you must set up another host to provide the (PC)NFS authentication.

(PC)NFS version 2 supported

The filer 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.

Enabling the daemon

Default daemon configuration

The `pcnfsd` daemon is disabled by default. You need to enable the daemon before (PC)NFS clients can access volumes and directories on the filer.

Procedure for enabling the daemon

Complete the steps in the following table to enable the `pcnfsd` daemon.

Step	Action
1	Enter the following command to enable the `pcnfsd` daemon: **`options pcnfsd.enable on`**
2	If you want the `pcnfsd` daemon to start automatically whenever the filer reboots, place the following command in the */etc/rc* file in the filer default volume: `options pcnfsd.enable on`

Setting up (PC)NFS user entries

Procedure for setting up user entries

Use the steps in the following table to set up user entries for (PC)NFS users.

Step	Action	
1	Verify that each user of (PC)NFS has an entry in your UNIX */etc/passwd* file and */etc/shadow* file, if applicable.	
2	**If...**	**Then...**
	The filer has not been configured to use NIS for the password database	Copy */etc/passwd* and */etc/shadow*, if applicable, to the filer. **Results:** 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.
	The filer has been configured to use NIS for the password database	No action is necessary.

Example of how to copy files manually

This example shows how to copy the */etc/passwd* and */etc/shadow* files from a UNIX host to the filer. In this example, the filer root volume is mounted to the UNIX host at the following mount point:

/nfs/toaster

The following commands copy the */etc/passwd* and */etc/shadow* files from the UNIX host to the filer:

```
cp /etc/passwd /nfs/toaster/etc/passwd
cp /etc/shadow /nfs/toaster/etc/shadow
```

Copying files to the filer automatically

You can set up a UNIX host to copy the */etc/passwd* and */etc/shadow* files automatically. The following table shows you which information to read to learn about automatic copying.

If the filer...	Then read...
Uses NIS to authenticate users	"Copying files to the filer automatically from NIS Makefile"
Uses */etc/passwd* and */etc/shadow* files to authenticate users	"Copying files to the filer automatically with a cron job"

Copying files to the filer automatically from NIS Makefile

If you use NIS, you can modify the Makefile for the NIS master to copy the */etc/passwd* and */etc/shadow* files to the filer automatically.

Example: Put the following code at the end of the NIS Makefile section for `passwd.time`, substituting 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.

```
@mntdir=/tmp/nac_etc_mnt_$$$$;\
if [ ! -d $$mntdir ]; then rm -f $$mntdir; mkdir $$mntdir; fi;\
for filer in toaster1 toaster2 toaster3 ; do \
mount $$filer:/vol/vol0/etc $$mntdir;\
mv $$mntdir/hosts $$mntdir/hosts.bak;\
cp /etc/passwd $$mntdir/passwd;\
cp /etc/shadow $$mntdir/shadow;\
umount $$mntdir;\
done;\
rmdir $$mntdir
```

Copying files to the filer automatically with a cron job

If you do not use NIS, you can establish a `cron` job to update the filer.

Setting the default umask

What is umask?

The `umask` option for the `pcnfsd` daemon controls the permissions that the filer assigns to directories and files created by (PC)NFS clients.

How umask works

The maximum permission for a user, group, or everyone else is 6, which enables the appropriate user to read files, write files, or have no access to files. The value specified in the `umask` option is subtracted from 0666 to determine the permissions assigned to a file.

Example: The following example enables a file's owner to read and write a file, but denies access to the file by anyone else:

```
066
```

The umask format

The format of `umask` consists of three integers (*nnn*) that have values of 0, 2, 4, or 6. The integers specify permissions in the following manner:

- The first (left) integer specifies the permissions that the owner of a file has for that file.
- The second (middle) integer specifies the permissions that members of the same group as the file's owner have for the file.
- The third (right) integer specifies the permissions that everyone who is not the owner or a member of the owner's group has for the file.

Values of umask

The following table shows the `umask` integers that you can specify in any of the `umask` integer positions (*nnn*).

umask integer	Meaning
0	Creates files with permissions that enable users, groups, or everyone else to read from and write to files.
2	Creates files with permissions that enable users, groups, or everyone else to write to files.

umask integer	Meaning
4	Creates files with permissions that enable users, groups, or everyone else to read from files.
6	Creates files with permissions that prevent users, groups, or everyone else to read from or write to files.

The default umask

The filer uses a default umask of 022, which

- enables users to read and write their own files
- enables group members and others to read any users' files

Syntax

The syntax for the `umask` option is as follows:

`options pcnfsd.umask` *nnn*

nnn is the umask.

Section C: Configuring a filer for WebNFS

About configuring a filer for WebNFS

The filer can respond to NFS requests from browsers

The filer can use NFS rather than HTTP to respond to file transfer requests made through Web browsers that support the WebNFS protocol.

The filer does not need a license for the HTTP protocol to respond to WebNFS requests; however, the filer must be licensed for the NFS protocol.

Web browser requirements

To access files through the WebNFS protocol, users type URLs that start with "nfs://". Web browsers must be capable of sending requests using the WebNFS protocol.

Advantages of WebNFS

With WebNFS, the filer can transfer files much faster than with HTTP because the WebNFS protocol can transfer several files, including graphics files, with only one TCP connection. The HTTP protocol, in version 1.1, requires a separate connection for each file that is transferred.

How WebNFS restricts file access

WebNFS access does not use the `mount` command to enable access to a subtree, and does not consider UID/GID mappings.

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.

Related information

For information about the `exportfs` command, consult the *na_exportfs*(1) man page.

Setting up WebNFS

Procedure for setting up WebNFS

Use the steps in the following table to set up WebNFS.

Step	Action	
1	Enter the following command to turn on WebNFS: `options nfs.webnfs.enable on`	
2	**If you...**	**Then...**

	Want to specify a public directory, known as the root directory, for WebNFS access	Enter the following commands, replacing *directory* with the path to the root directory:
		`options nfs.webnfs.rootdir directory`
		`options nfs.webnfs.rootdir.set TRUE`
		Results: All NFS lookups are done relative to the root directory. All WebNFS clients can access files and directories under the root directory.

	Do not want to specify a public directory for WebNFS access	Do nothing.

Example of specifying WebNFS root directory

To use the */vol/vol0/webfiles* directory as the WebNFS root directory, you enter the following commands:

```
options nfs.webnfs.rootdir /vol/vol0/webfiles
options nfs.webnfs.rootdir.set TRUE
```

Managing WebNFS

Tasks you can perform

You can perform the following tasks to manage WebNFS service:

- Change the root directory.
- Disable the root directory.
- Turn off WebNFS service.

Changing the root directory

Complete the steps in the following table to change the WebNFS root directory.

Note

If you use the `vol rename` command to change the name of the volume in which the WebNFS root directory resides, remember to use this procedure to specify the new name of the root directory.

Step	Action
1	Enter the following command, replacing *newdir* with the path of the new root directory: **`options nfs.webnfs.rootdir newdir`** **Example:** To change the root directory to */vol/vol0/corpwebfiles*, you the following command: **`options web.webnfs.rootdir /vol/vol0/corpwebfiles`**
2	Enter the following command to enable the root directory: **`optins nfs.webnfs.rootdir.set TRUE`**

Disabling the root directory

Complete the step in the following table to disable the root directory.

Step	Action
1	Enter the following command: **`options nfs.webnfs.rootdir.set FALSE`**

Turning off WebNFS service

Complete the step in the following table to turn off WebNFS service.

Step	Action
1	Enter the following command: `options nfs.webnfs.enable off`

Section D: Displaying NFS statistics

About displaying NFS statistics

The nfsstat command displays NFS and RPC statistics

The `nfsstat` command displays statistics about NFS and Remote Procedure Calls (RPCs) 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 about this topic is *Managing NFS and NIS* by Hal Stern, O'Reilly & Associates, Inc.

Syntax

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

```
nfsstat [ interval ] | [ -h [ ip_address | host_name ] ] | [ -l ] |
[ -z ]
nfsstat -h [ ip_address | host_name ]
nfsstat -l
nfsstat -z
```

Options

The following table describes the options for the nfsstat command.

Option	Description
none	When no options are specified, the command displays statistical information since the last time the filer was rebooted.
interval	When an interval is specified, the command displays statistics continually. The interval specifies the number of seconds the command waits between updates.

Option	Description
`-h`	Displays statistics for a single client. You must provide the client's host name or IP address as an argument to the `-h` option. ◆ To use the `-h` option, you must enable the `nfs.per_client_stats.enable` option by entering the following command: **`options nfs.per_client_stats.enable on`** ◆ Enable the per-client statistics collection mode as soon as possible after you start the filer or 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.
`-l`	Displays statistics that have been collected for all NFS clients.
`-z`	Resets the statistics counters.

Example: no options

The following lines show the output of the `nfsstat` command when you specify no options:

```
nfsstat
Server rpc:
TCP:
calls      badcalls   nullrecv   badlen     xdrcall
0          0          0          0          0

UDP:
calls      badcalls   nullrecv   badlen     xdrcall
24         0          0          0          0

Server nfs:
calls      badcalls
24         0

Server nfs V2: (24 calls)
null     getattr   setattr  root      lookup  readlink   read
0 0%     5 21%     0 0%     0 0%      16 67%  0 0%       0 0%
wrcache  write    create   remove    rename  link       symlink
```

```
0 0%     0 0%     0 0%     0 0%      0 0%    0 0%       0 0%
mkdir    rmdir    readdir  statfs
0 0%     0 0%     3 13%    0 0%

Server nfs V3: (0 calls)
null     getattr   setattr  root      lookup  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%
```

Example: using the -l option

The following lines show the output of the `nfsstat` command when you specify the `-l` option:

```
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%)
```

Example: using the -h option

The following lines show the output of the `nfsstat` command when you specify the `-h` option with a host name:

```
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%
```

Example: resetting counters with the -z option

The following command resets the counters:

```
nfsstat -z
```

Related information

The following table lists additional information that you might find helpful.

To learn about...	Read...
How the Network Appliance custom Management Information Base (MIB) provides the same information as the `nfsstat` command through SNMP	“About the Network Appliance custom MIB” on page 98.
About the `nfsstat` command	The *na_nfsstat(1)* man page.

CIFS Administration 7

What is CIFS?

CIFS (Common Internet File System) is a file-sharing protocol based on the Server Message Block (SMB) protocol widely in use by personal computers and workstations running a wide variety of operating systems. CIFS provides an open, cross-platform mechanism for client systems, including Windows systems, to request file and print services from server systems over a network.

This chapter describes how to manage the CIFS file protocol and users.

What you can do only from the filer command line or FilerView

You can do some operations that affect CIFS administration only from the filer command line or FilerView. These are

- Viewing volumes and examining their status. For additional information about volume operations, see Chapter 3, "Managing Disks and File Systems."
- Setting or changing volume and qtree security style and oplocks status. For information about administering qtrees from the filer command line, see Chapter 11, "Administering Qtrees." For information about how to administer qtrees in FilerView, see FilerView on-line help.

Chapter contents

This chapter discusses the topics listed in the following table.

Read...	To learn about...
"CIFS limitations" on page 168	Describes CIFS limitations when using the ACL editor, User Manager, and Server Manager.
"Limits on CIFS open files, sessions, and shares" on page 169	CIFS limits on a filer.
"Changing or viewing the filer's description" on page 170	How to change or examine descriptive information about a filer.
"Adding users to the filer" on page 172	How to add users to a filer from Windows NT and the filer command line.
"Adding local groups to the filer" on page 174	How to use User Manager to add local groups to the filer.

Read...	To learn about...
"Using CIFS commands with a remote shell program" on page 177	How to use a remote shell to automate CIFS management.
"Enabling guest access" on page 178	How to give access to users who do not have an account on the filer.
"Displaying a filer's shares" on page 180	How to display information about the shared directories on a filer.
"Creating and changing a share" on page 183	How to create a share from Windows NT and the filer and change the value of its parameters.
"Displaying information about shares" on page 189	How to get information about shares from Windows NT and the filer.
"Deleting a share" on page 192	How to delete shares from Windows NT and the filer.
"Creating a home share for each user" on page 194	How to create home shares from Windows NT and the filer.
"Assigning and changing access rights" on page 197	How to define user or group access rights to a share.
"Displaying access rights to an NTFS file" on page 202	How to show who has rights to an NTFS file on the filer.
"Viewing and changing UNIX permissions from Windows" on page 203	How to use SecureShare™ Access to view and change UNIX permissions.
"Sending a message to all users on a filer" on page 206	How to send a message in an alert box to all users on a filer.
"Using oplocks" on page 207	How to set and remove opportunistic locking from the filer as a whole.
"Displaying CIFS statistics" on page 209	How to display CIFS information.

Read...	To learn about...
"Displaying CIFS session information" on page 210	How to display information from the filer command line about who is logged in to CIFS.
"Stopping and restarting CIFS sessions" on page 212	How to stop and start CIFS service from the filer command line.
"Reconfiguring the filer for CIFS" on page 216	How to run the `cifs setup` program to reconfigure CIFS on the filer

Effects of renaming a volume on shares

If you change the name of a volume that contains at least one share, whether through Windows NT or the `vol rename` command, the filer automatically offers the share to users in the renamed volume at the next reboot. The new volume name is reflected in the `cifs shares` command.

Scope of this chapter

This chapter does not discuss procedures that take place on the clients and it does not describe how a machine joins a workgroup or domain. For information about these topics, refer to the manuals for your client operating systems or books about PC networking.

CIFS limitations

Introduction

This section describes CIFS limitations when operating on files on the filer.

ACL editing limitations

When you view the Security tab in the Properties dialog box on a Windows computer for a file that resides on a filer, the capability to edit the file's ACL might be limited. If the filer is using */etc/passwd* for authentication, or if the file you are examining does not have an ACL set on it, the following conditions are in effect:

- The Add and Remove buttons are disabled.
- A warning appears to the effect that an ACL, if one is set, can only be viewed and not edited.
- If an ACL has been set, it displays a single entry, Everyone, that gives the current user's rights to the file.

User Manager limitations

The Policy menu items and the New Users menu item are permanently disabled.

Server Manager limitations

The following Server Manager features are not supported:

- stopping and starting services
- specifying the recipients of alerts

Limits on CIFS open files, sessions, and shares

CIFS file access limits

The filer is subject to limits on file access through CIFS, as summarized in the following tables.

Limits for F200 and F300 series filers

The following table shows access limits for F200 and F300 series filers.

CIFS limits by filer memory	128 MB	192 MB	256 MB
Maximum number of users	1,600	3,000	4,000
Maximum number of open files	32,000	60,000	80,000
Maximum number of locked files	36,096	66,144	88,192
Maximum number of shares	3,200	6,000	8,000

Limits for F500 and F600 series filers

The following table shows access limits for F500 and F600 series filers.

CIFS limits by filer memory	128 MB	192 MB	256 MB	512 MB
Maximum number of users	1,200	2,200	3,200	7,200
Maximum number of open files	24,000	44,000	64,000	144,000
Maximum number of locked files	32,000	50,144	72,192	160,384
Maximum number of shares	2,400	4,400	6,400	14,400

Changing or viewing the filer's description

When to change or view a filer's description

The description of a filer appears next to its name wherever a machine's description comments appear. Initially, the filer has no description. You might want to change the description to something more informative so that you can distinguish filers from each other. You might want to view the description of a filer to find out, for example, what a particular filer does or who is in charge of it.

Changing a filer's description from Server Manager

To change the description of a filer from Server Manager, follow these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.
4	Double-click a filer name. **Result:** The Filer Properties window, shown below, appears. 
5	Type a new description of the filer in the Description field.
6	Click OK. **Result:** The new description goes into effect.

Viewing a filer's description from the filer command line

To view the description of a filer, enter the following command:

```
cifs comment
```

Changing a filer's description from the filer command line

To change the description of a filer, enter the following command:

```
cifs comment "description"
```

where *description* is its description, which must be enclosed in quotation marks. The description must be no longer than 48 characters.

Adding users to the filer

When you add CIFS users

By default, Windows NT users who have an account map to a UNIX account of the same name. If they do not have a Windows NT account, the user becomes a generic user and uses the generic account, which is described in "Generic accounts" on page 178. In the rare case that you must add a user explicitly, you can do so with the methods described in this section.

You can add CIFS users to the filer at any time. The method you use for adding users to the filer depends on whether you are authenticating with a domain controller or the UNIX password database.

When authenticating with a domain controller

To add a user, create an account for the CIFS user within your Windows NT domain environment. If you want the user to also use UNIX files, either create an entry in the */etc/passw*d file on the filer or include the user in the */etc/usermap.cfg* file.

What is the /etc/usermap.cfg file? The */etc/usermap.cfg* file explicitly maps Windows NT users to the correct UNIX account. The file can be coded as follows:

- As an ASCII file with non-ASCII characters encoded as ISO Latin-1SGML entity names. For information about SGML entity names, consult SGML documentation.
- As a UNICODE file created using the Windows NT tools Notepad or Microsoft Word.

The filer automatically detects which of these forms is in use.

Format of the /etc/usermap.cfg file: The format of the */etc/usermap.cfg* file is a list of text records in any of the following three forms:

- *Domain\NTUser UnixUser*
- *NTUser UnixUser*
- * *UnixUser*

Lines are processed sequentially.

Symbol conventions: The following symbol conventions are in effect:

- An asterisk (*) matches any name.

For example, to map all unmapped users to the UNIX “nobody” account, add the following line:

```
*       nobody
```

- The null string ("") matches no name and rejects any user.

  For example, to prevent access completely, add the following line:

  ```
  *       ""
  ```

  This line prevents the default mapping of Windows NT users who have an account map to a UNIX account of the same name. Any lines after this line are disregarded by the filer.

- You can use either spaces or tabs as separators.

Name requirements: Windows NT and UNIX names have different requirements, as follows:

- Windows NT names are case-insensitive and can contain non-ASCII characters within the character set in the current code page. For information about code pages, see *Getting Started* or the *Upgrade Guide*. Windows NT user names can contain spaces, in which case you must enclose the name in quotation marks.
- UNIX user names are case-sensitive and must in be in ASCII.

Default file contents: If the filer is domain authenticated, by default the */etc/usermap.cfg* file contains the following line:

```
domain\administrator    root
```

When authenticating with the UNIX password database

To add a user, enter the user’s information into the NIS password and group maps.

Note

If you do not use NIS, create entries for the user in the filer’s */etc/passwd* and */etc/group* files.

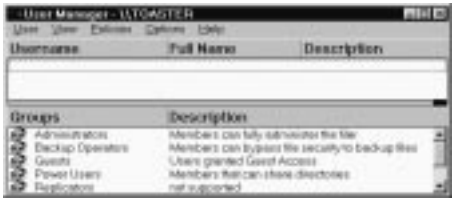
Adding local groups to the filer

How to add a local group

You add a local group to the filer with the New Local Group window in the User Manager for Domains.

Adding a group with the New Local Group window

Use the New Local Group window to create a new local group as follows

Step	Action
1	Open User Manager for Domains.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK. **Result:** The User Manager window shows information for the specified filer, as shown below. 
4	From the New User menu, choose New Local Group. **Result:** The New Local Group window, shown below, appears. 

Step	Action
5	Type the name of the new group in the Group Name text box.
6	Type a description for the new group in the Description text box.
7	To add a member, type a user or group name in the Members list box or use the Add Users and Groups window, as described in Step 8.
Adding a name with the Add Users and Groups window	
8	Click Add. **Result:** The Add Users and Groups window, shown below, appears. 
9	Click the arrow next to the List Names From text box to choose a domain that contains names that you want to add. **Result:** A list of names in the selected domain appears in the Names list box.
10	To add a name, type one or more user names in the Add Names list box or select one or more names and click Add.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>11</td><td>Specify one or more name in one of three ways:
Click one or more names in the Names list box.
Type valid user or group names in the Add Names list box.
Search for a name by following these steps:
1. Click the Search button.
Result: A search window appears where you can find a user or group.
2. Click a name in the search window, and follow these steps.<table><tr><th>Step</th><th>Action</th></tr><tr><td>a.</td><td>Select a valid group name.</td></tr><tr><td>b.</td><td>Click the Members button.
Result: A Global Group Membership window appears showing the names in the selected group.</td></tr><tr><td>c.</td><td>Select one or more names in the Global Group Membership window.</td></tr></table></td></tr>
<tr><td>12</td><td>Click Add in the window where you selected the names.
Result: The names are added to the Add Names field of the Add Users and Groups window.</td></tr>
<tr><td>13</td><td>If you want, display the full name of a user associated with an account name by clicking Show Full Name.</td></tr>
<tr><td>14</td><td>Click OK.
Result: The names appear in the Members field of the New Local Group window.</td></tr>
<tr><td colspan="2">Final steps in the New Local Group window</td></tr>
<tr><td>15</td><td>To remove one or more names from the list, select a name or names in the Names list, then click Remove.</td></tr>
<tr><td>16</td><td>Click OK to put the additions into effect.</td></tr>
</table>

Using CIFS commands with a remote shell program

What you can use a remote shell program for

You can use a remote shell program, such as `rsh`, to

- execute CIFS commands
- create scripts containing CIFS commands to automate similar access rights tasks

UNIX example

For example, using `rsh` on the administration host, you can set access rights for a specific user as follows:

```
rsh -l root -n toaster cifs access home jsmith r-x
```

In this example, the filer name is toaster, the user is jsmith, and the share is home. The user has read, execute, and browsing rights to the directory on the filer that has been defined as the home share.

Automating access rights

Because you can use CIFS commands through a remote shell program, 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 similar user information for each filer at your site.

Required information in hosts.equiv file

Make sure that the following entries are added to the *hosts.equiv* file in the *etc* directory:

host *user18*
host root

where *host* is the host you are using and *user* is your name.

Note

Make sure that you include both lines, or not all remote shell functionality works properly.

Enabling guest access

Two ways to give access to users who have no /etc/passwd entry

You can enable users who do not have an entry in the */etc/passwd* file on the filer to have access to the filer through CIFS in one of two ways.

For users who are...	Use...
not in a trusted domain	a guest account
authenticated	a generic user account

Guest accounts

If you using Windows NT domain authentication, guests are users who are not in trusted domains.

If you use a UNIX password database for authentication, guests are users who do not have an entry in the database.

Setting up a guest account: To set up a guest account, use the following `options` command:

```
options cifs.guest_account account_name
```

where *account_name* is the name of the guest account, usually *guest* or *nobody*, which is a preconfigured account in the */etc/passwd* file.

If you are using UNIX-style authentication, set the guest account to the name of an account in the UNIX password database, typically *guest*, which is mapped to the UNIX account *nobody* with the same access rights as the user *everyone*.

Disabling guest access: To disable guest access, use the following `option` command:

```
options cifs.guest_account
```

with a “” as *account_name*.

Generic accounts

- If you are using Windows NT domain authentication, a generic account is mapped by default with the name “pcuser.” The generic account enables users who meet the criteria described later in this section to connect to NTFS or mixed qtrees on the filer. For information about qtrees, see Chapter 11, “Administering Qtrees,” or FilerView on-line help.

Note

For generic accounts to be active, pcuser must be an account in the */etc/passwd* file.

Who can use the generic account: To use the generic account, a user must

- be authenticated
- be in a trusted domain
- not have an entry in the UNIX password database

All users of the generic account have the same UNIX rights and the Windows NT rights granted by their Windows NT group membership because they appear as one account to the system.

Changing the name of the generic account: To change the name of the generic account, use the following `options` command:

```
options cifs.generic_account newname
```

where *newname* is the new account name.

Disabling the generic account: To disable the generic account, use the following `options` command:

```
options cifs.generic_account “”.
```

Displaying a filer's shares

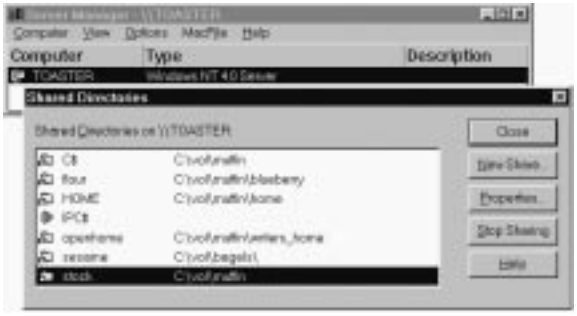
Ways to share folders

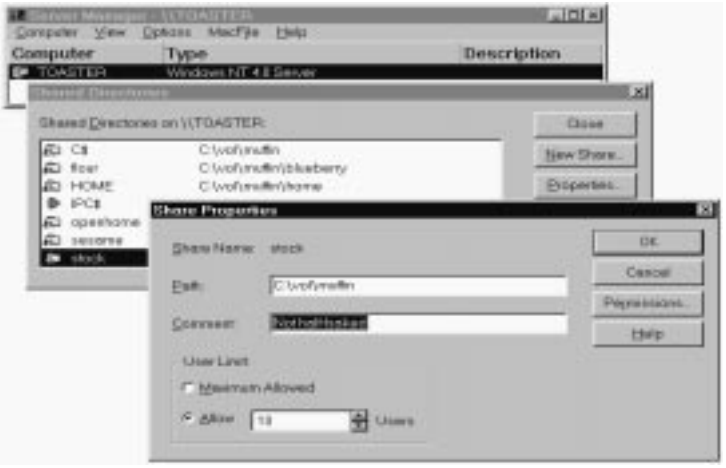
If you want to create a folder to be shared by CIFS clients, do one of the following:

- From Windows NT, use Server Manager as described in the next section, "Using Server Manager to display a filer's shares."
- From the filer command line, use the `cifs shares` command to display share information, as described in "Using the cifs shares command to display a filer's shares" on page 181.

Using Server Manager to display a filer's shares

To display a filers's shares with Server Manager and get detailed information, follow these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.
4	Choose Computer > Shared Directories. **Result:** The Shared Directories window, shown below, appears. 

Step	Action
5	To get additional details about a share, double-click a share name. **Result:** The Share Properties window, shown below, appears. 

Using the cifs shares command to display a filer's shares

To display the filer's list of shares from the filer's command line, use the `cifs shares` command.

Command syntax: The syntax is as follows:

```
cifs shares
```

Example of displaying a filer's shares: The following example shows a filer's shares:

```
cifs shares
Name         Mount Point                      Description
----         -----------                      -----------
HOME         /vol/muffin/home                 Default Share
                       everyone / Full Control
                       techpubs{g} / Full Control
C$           /vol/muffin                     Remote Administration
                       everyone / Full Control
                       BUILTIN\Administrators / Full Control
openhome     /vol/muffin/writers_home         readable home dirs
             ... user limit=1200
             ... forcegroup=techpubs
                       everyone / --x
```

```
                    techpubs{g} / r-x
stock       /vol/muffin                     Not half-baked
            ... user limit=10
                    everyone / Full Control
flour       /vol/muffin/blueberry           Flour power
                    everyone / Full Control
sesame      /vol/bagels/
                    everyone / Full Control
```

Creating and changing a share

Ways to share folders

If you want to create a folder to be shared by CIFS clients, do one of the following:

- From Windows NT, use Server Manager as described in the next section, "Creating a share from Server Manager."
- From the filer command line, use the `cifs shares` command to define a new share, as described in "Creating a share with the cifs shares command" on page 186.

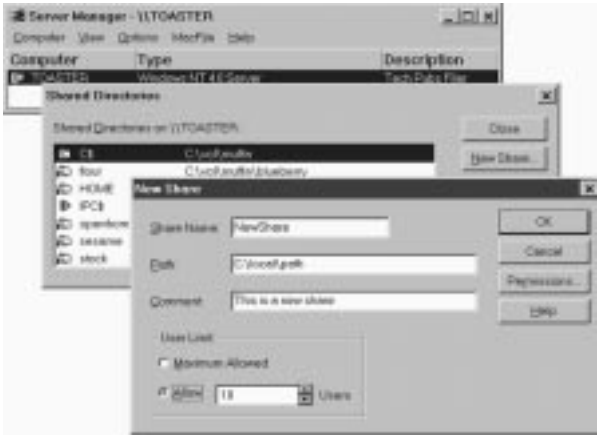
Note

By default, three shares are created during CIFS setup: *C$*, *IPC$*, and *HOME*. In filer console displays, the *C$* share corresponds in UNIX to */vol/vol0* and the HOME share corresponds to */vol/vol0/home*.

Creating a share from Server Manager

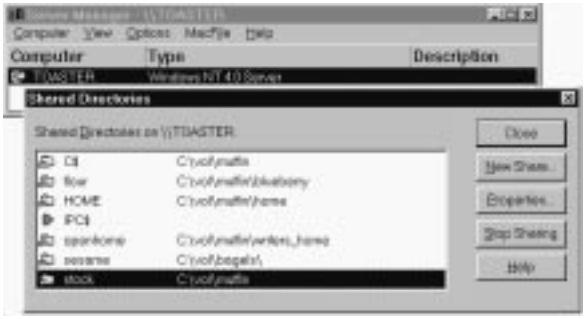
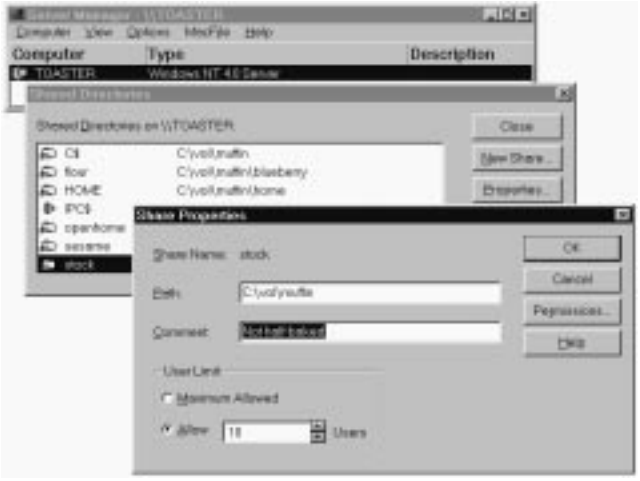
To create a share from the Windows NT desktop using Server Manager, create a folder, then share it by following these steps.

Step	Action
1	Create a folder on the filer.
2	Open Server Manager.
3	Choose Select Domain.
4	Select the filer you want by typing its UNC name (for example, \\FILERNAME) in the Domain field, then clicking OK.

Step	Action
5	Chose Computer > Shared Directories, then click the New Share button. **Result:** The New Share window, shown below, appears. 
6	Type a name for the new share in the Share Name field. The share name is case-sensitive.
7	Type the local path here of the folder you created in Step 1. This is usually C:*VOL\VOLNAME\FOLDER*.
8	Type a description of the share in the Comments field, if you want.
9	If you do not want to limit the number of users that can connect to the share at the same time, select Maximum Allowed under User Limit.
10	To limit the number of users that can connect to the share at the same time, click the arrows next to Users until the desired number appears.
11	If the share has Windows NT security, click Permissions to set permissions.
12	Click OK. **Result:** The New Share window disappears and the share is created.

Changing the share description and user limit with Server Manager

To change the description and user limit with Server Manager, follow these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.
4	Choose Computer > Shared Directories. **Result:** The Shared Directories window, shown below, appears. 
5	Double-click a share name. **Result:** The Share Properties window, shown below, appears. 

Step	Action
6	Type a new description of the share in the Comments field, if you want.
7	If you do not want to limit the number of users that can connect to the share at the same time, select Maximum Allowed under User Limit.
8	To limit the number of users that can connect to the share at the same time, click the arrows next to Users until the desired number appears.
9	Click Permissions to change permissions.
10	Click OK. **Result:** The window disappears and the new values go into effect.

Creating a share with the cifs shares command

Following is the syntax for the `cifs shares` command for creating a share:

```
cifs shares -add sharename path [ -comment description ]
[ -forcegroup groupname ] [ -maxusers n ]
```

The following table describes the parameters.

Variable	Description
description	A string describing the purpose of the share. It must contain only characters in the current code page. 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, you must enclose it in single quotation marks.
groupname	The name of the group you want all files in the share to get the group membership of.
n	The maximum number of users that you specify can connect to the share at the same time. The limit on this number is dependent on filer memory, and is shown in “Limits on CIFS open files, sessions, and shares” on page 169.

Variable	Description
path	The path name, relative to the root of the filer's file system, of the root directory of the share. **Note** Because the `cifs shares` command is case-sensitive, be sure that you use the appropriate case when entering the path name. Separators used in the path name must be forward slashes (/).
sharename	The name of the share, which is used by CIFS users to obtain access to the directory on the filer. If *sharename* already exists, the `cifs shares -add` command fails. **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.

Example: The following example creates a new share called *library*:

```
cifs shares -add library /vol/vol0/home/lib 'New file library'
```

This example creates the *library* share and defines it as *HOME\LIB* (*/home/lib* in UNIX notation) in the filer's root volume. With the appropriate access rights, CIFS users can gain access to the *HOME\LIB* directory in the root volume, which is displayed as the *library* share on their computers. For more information about setting access rights, refer to "Assigning and changing access rights" on page 197.

Using the cifs shares command to change the share

To change the description, forced file ownership, and user limits of a share, use the following command:

```
cifs shares -change sharename [ -comment description | -nocomment ]
[ -forcegroup groupname | -noforcegroup ]
[ -maxusers n | -nomaxusers ]
```

The following table describes the parameters.

Parameter or variable	Description
description	A string describing the purpose of the share. It must contain only characters in the current code page. It is displayed in the share list in Network Neighborhood on the client. If the description contains spaces, you must enclose it in single quotation marks.
groupname	The name of the group you want all files in the share to get the group membership of.
n	The maximum number of users that you specify can connect to the share at the same time. The limit on this number is dependent on filer memory, and is shown in “Limits on CIFS open files, sessions, and shares” on page 169.
-nocomment	Specifies no description.
-noforcegroup	Specifies no particular group to own the files that are created in the share.
-nomaxusers	Specifies no maximum number of users that can connect to the share at the same time.
sharename	The name of the share, which is used by CIFS users to obtain access to the directory on the filer. It must contain only characters in the current code page.

Displaying information about shares

Methods of displaying information about shares

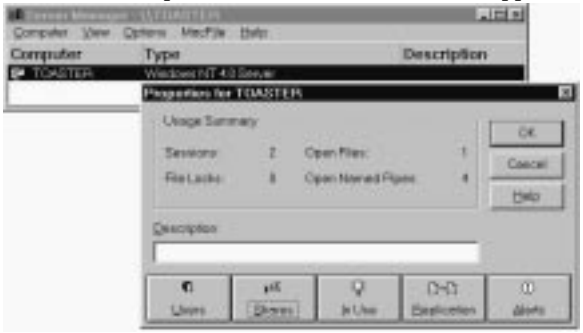
You can display information about a specific share or all shares on the filer.

To display information about shares, do either of the following:

- From the Windows NT desktop, use Server Manager as described in the next section, "Using Server Manager to view information about shares."
- From the filer, use the `cifs shares` command as described in "Using the cifs shares command to view information about shares" on page 190.

Using Server Manager to view information about shares

To display information about shares from Server Manager, follow these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.
4	Double-click the name of a filer. **Result:** The Filer Properties window, shown below, appears. 

Step	Action
5	Click the Shares button. **Result:** Information about each share on the filer appears, as shown below. 

Using the cifs shares command to view information about shares

To display information about a share or shares that exist, enter the following command:

```
cifs shares sharename
```

where *sharename* is the specific name of the share about which you want information. If you want information about all shares, leave *sharename* blank.

Examples of displaying share information: The following command example displays the share information only for the share *library* created by the `cifs shares -add` command described in “Creating a share with the cifs shares command” on page 186.

```
cifs shares library
```

The following command displays information about all shares:

```
cifs shares
```

The sample output is from `cifs shares`:

```
Name         Mount Point                      Description
----         -----------                      -----------
HOME         /vol/muffin/home                 Default Share
                       everyone / Full Control
                       techpubs{g} / Full Control
C$           /vol/muffin                     Remote Administration
                       everyone / Full Control
                       BUILTIN\Administrators / Full Control
openhome     /vol/muffin/writers_home         readable home dirs
             ... user limit=1200
             ... forcegroup=techpubs
                       everyone / --x
                       techpubs{g} / r-x
stock        /vol/muffin                      Not half-baked
             ... user limit=10
                       everyone / Full Control
flour        /vol/muffin/blueberry            Flour power
                       everyone / Full Control
sesame       /vol/bagels/
                       everyone / Full Control
```

Deleting a share

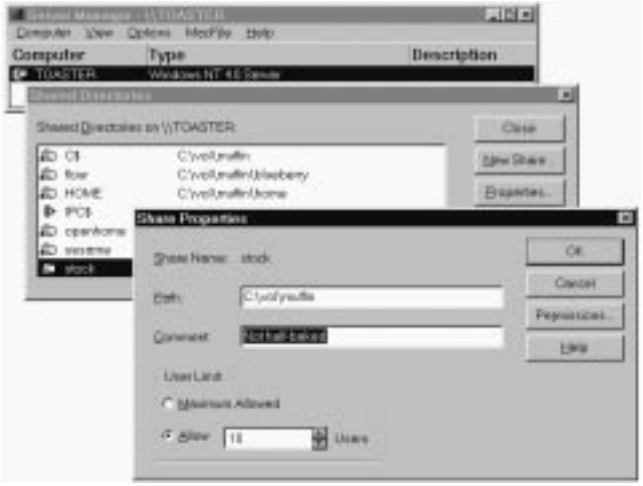
How to delete a share

To delete shares, do either of the following:

- From the Windows NT desktop, use Server Manager as described in the next section, "Using Server Manager to delete a share."
- From the filer, use the `cifs shares` command, as described in "Using the cifs shares command to delete shares" on page 193.

Using Server Manager to delete a share

To delete a share from Server Manager, follow these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.
4	Choose Computer > Shared Directories, then click the Properties button. **Result:** The Share Properties window appears, as shown below. 

Step	Action
5	Click Stop Sharing, then click OK. **Result:** The folder is no longer shared.

Using the cifs shares command to delete shares

To delete a share, use the `cifs shares -delete` command.

Command syntax: The `cifs shares -delete` command has the following syntax:

`cifs shares -delete` *`sharename`*

where *sharename* is the specific name of the share that you want to delete.

Example: The following command deletes the share created by the `cifs shares -add` command described in “Creating a share with the cifs shares command” on page 186.

```
cifs shares -delete library
```

Creating a home share for each user

When to create a home directory

You can create a share that contains home directories of registered CIFS users.

For example, if there are users called *user1* and *user2*, the share contains directories for user1 and user2. When user1 connects to the filer and asks for its list of shares, the display shows a share called user1, but not user2 or any other individual user.

Note

If Domain1\user and Domain2\user are the same, they do not have different home directories. To prevent access by the wrong user, set Windows permissions and UNIX permissions at the root of the user's home directory.

Accessing a home directory

Users access their home directories in the same way as any other share. That is, users can open the share with Network Neighborhood, by mapping a drive, or by using a UNC name. The UNC name is

`\\filer\`*`username`*

Share name length limitations

Because share names are truncated to 12 characters, the home directory name might show a truncated version of the user's account name.

For example, consider the 13-letter name *administrator*. From the filer 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*, they would see an offer for their own home directory, but both would see it as an offer to share the name *administrato*. Therefore, each user gets the correct directory, even though the share names appear the same.

Creating a share containing user home directories

The following procedure creates a share that automatically contains home directories for CIFS users:

1. Enable the home directory option by entering the following command.

   `options cifs.home_dir` *`homedirpath`*

   Where *homedirpath* is the UNIX path name that will be mapped to the share CIFS.HOMEDIR. This share automatically contains CIFS user home directories.

   If the directory containing the home directories is */vol/vol1/homes*, enter the following command:

   `options cifs.home_dir /vol/vol1/homes`

2. Within the directory specified by the `cifs.home_dir` option, create a directory for each user. Make sure that the following conditions are met:
   - Each directory name matches the user's login name exactly.
   - Each user is the owner of the directory.

Creating share home directories

Using the `cifs.home_dir` option to create home directories is useful only if users do not need to read or write other users' home directories. If they must access other users' home directories, follow this procedure:

1. Create an additional share, from either Windows or the filer, that maps to the same path name as the CIFS.HOMEDIR share.
2. From either Windows or the filer, assign each user the appropriate access permissions to other users' home directories.

Example from the filer: The following example shows how to create an additional share, assign user access rights to the share, and display the share information from the filer command line. If authentication is through the */etc/passwd* file, UNIX permissions are shown; otherwise, Windows NT permissions appear.

```
cifs shares -add enghomes /vol/vol1/homes \
   -comment "Readable home directories"

“Readable home directories”

cifs access enghomes -g engineering r-x
```

```
cifs shares
Name         Mount Point                       Description
----         -----------                       -----------
enghome      /vol/vol1/homes                   Readable home
directories
                        engineering{g} / r-x
C$           /vol/vol0                        Remote Administration
                        BUILTIN\Administrators / Full Control
HOME         /vol/vol0/home                    Default Share
                        everyone / Full Control
CIFS.HOMEDIR /vol/vol1/homes                   Home Directories
                        everyone / Full Control
```

Result: Users in the *engineering* group can read all home directories in HOMES on the root volume, which corresponds to the share defined by the `options cifs.home_dir` command. However, they can only write to their own home directories, which reside in the CIFS.HOMEDIR share.

Assigning and changing access rights

When to assign or change access rights

After you create a share, you define the user or group access rights to the share. If a group or a user no longer exists, you can remove the corresponding entry from an ACL.

Methods of assigning or changing access rights to a share

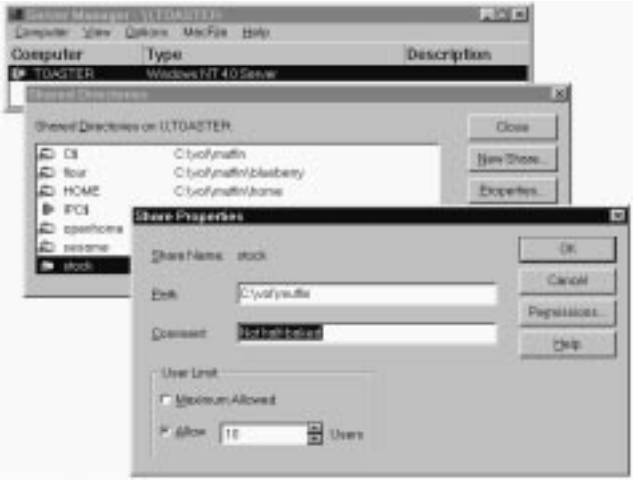
To assign access rights, use either of the following methods:

- From Windows, follow the instructions in the next section, "Assigning or changing access rights with Server Manager."
- From the filer command line, do one of the following:
  - Add access with the `cifs access` command, as described in "Giving access with the cifs access command" on page 200.
  - Remove a user or group with the`cifs access -delete` command, as described in "Removing a user or group with the cifs access -delete command" on page 201.

Assigning or changing access rights with Server Manager

To assign or change access rights with Server Manager, use the Access Through Share Permissions window by following these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.

Step	Action
4	Click Computer > Shared Directories > Properties button. **Result:** The Share Properties window appears, as shown below. 
5	Click Permissions. **Result:** The Access Through Share Permissions window, shown below, appears. 
6	To add a new user or group, click Add. **Result:** The Add Users and Groups window, shown below, appears.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td colspan="2">Adding a name with the Add Users and Groups window</td></tr>
<tr><td>7</td><td>Click the arrow next to the List Names From field to choose a domain that contains names that you want to add.

Result: A list of names appears in the Names list box.</td></tr>
<tr><td>8</td><td>Specify one or more names in any of five ways:
◆ Click one or more names in the Names list box.
◆ Type valid user or group names in the Add Names text box.
◆ Choose from all the names in a domain by following these steps:
1. Click Show Users to see at the bottom of the Names list box all the names in the domain you selected in Step 7.
2. Select one or more of them.
◆ Search for a name by following these steps:
1. Click the Search button.
Result: A search window appears where you can find a user or group.
2. Click a name in the search window.
3. Select from a list of group members by following these steps.
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>a.</td><td>Select a valid group name.</td></tr>
<tr><td>b.</td><td>Click Membership.

Result: A Global Group Membership window appears, showing the names in the selected group.</td></tr>
<tr><td>c.</td><td>Select one or more names in the Global Group Membership window.</td></tr>
</table>
</td></tr>
<tr><td>9</td><td>Click Add in the window where you selected the names.

Result: The names are added to the Add Names field of the Add Users and Groups window.</td></tr>
</table>

Step	Action
10	If you want, add or modify the access type by selecting one or more names and choosing an access type from the Type of Access list.
Steps in the Access Through Share Permissions window	
11	To assign or change an access type, select a name or names in the Names list, then click the arrow next to Type of Access and select an access type.
12	To remove one or more names from the list, select a name or names in the Names list, then click Remove.
13	Click OK.
Final step in the Properties window	
14	Click OK to put the changes into effect.

Giving access with the cifs access command

You use the `cifs access` command to assign access to a share from the filer command line. To change the permissions, run the command with the new permissions.

Command syntax: The `cifs access` command has the following syntax:

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

The `-g` flag specifies that the access rights are defined for a group.

rights can be UNIX-style permissions or Windows NT-style rights:

- UNIX-style permissions are defined as `r`, `w`, and `x`, which mean read, write and delete, and execute and browse, respectively. To deny a right to a user, use a hyphen (-).
- Windows NT-style rights are No Access, Read, Change, and Full Control.

Examples: Here are some examples of assigning access rights from the filer:

- `cifs access library -g engineering rwx`
- `cifs access library domain\joed Change`

Note

The group *everyone* is reserved. When you use it 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, you enter the following command:

```
cifs access library everyone rwx
```

Removing a user or group with the cifs access -delete command

If a user or group no longer exists, you can remove the corresponding entry from an ACL. Use the following command syntax to remove an entry in an ACL:

```
cifs access -delete share user | group
```

Following are some examples of removing entries from the *library* share.

The following command removes the *engineering* group from the *library* share.

```
cifs access -delete library engineering
```

The following command removes the user *joed* from the *library* share.

```
cifs access -delete library domain\joed
```

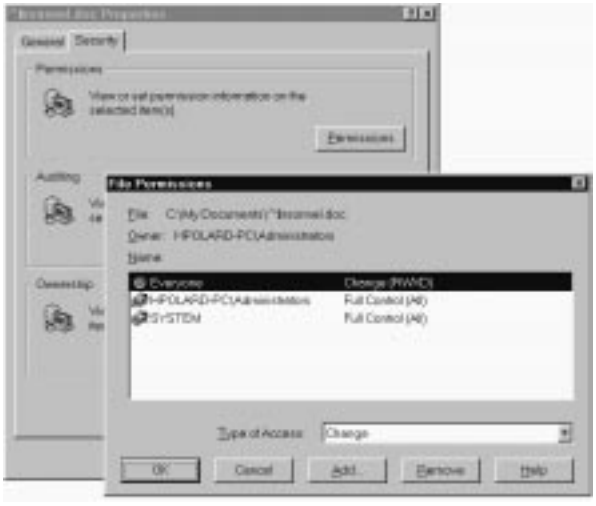
Displaying access rights to an NTFS file

Access rights display methods

You can display access rights to an NTFS file from Windows.

Displaying access rights from the Windows desktop

To display access rights to a file from the Windows desktop, follow these steps.

Step	Action
1	Right-click a file and choose Properties from the pop-up menu. **Result:** The Properties sheet appears.
2	Click the Security tab. **Result:** If the file is a Windows file, the Security sheet appears.
3	Click Permissions. **Result:** Permissions are displayed, as shown below. 

Viewing and changing UNIX permissions from Windows

How to change UNIX permissions

Some files and directories on the filer have both Windows and UNIX-style permissions. You can change Windows-style permissions from Windows by editing Windows permissions. To change UNIX permissions in a UNIX file system from the Windows desktop, you use the SecureShare Access tool.

To use the SecureShare Access tool from a client, you must install the tool on the client. For instructions about downloading the SecureShare Access tool, see the *Upgrade Guide* or *Getting Started*.

Note

To change UNIX permissions, you must understand them. Explaining UNIX permissions is outside the scope of this guide. Consult literature about UNIX for an explanation of UNIX permissions.

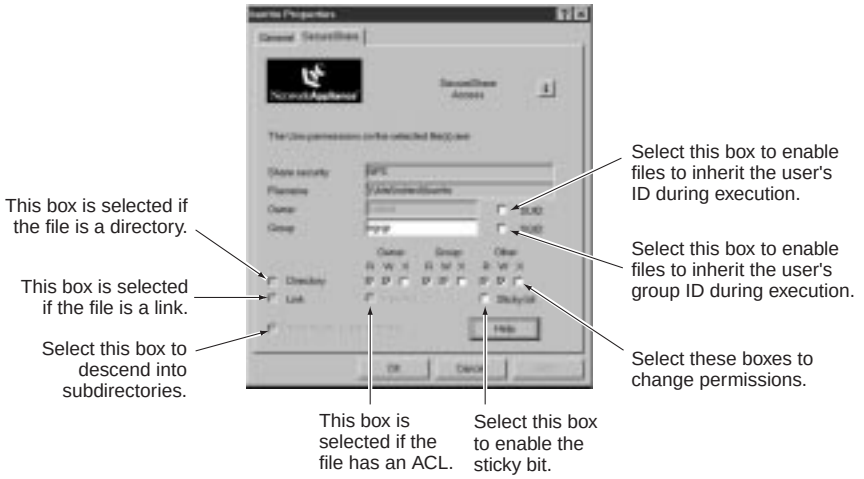
Displaying SecureShare Access

To display SecureShare Access, follow these steps.

Step	Action	Result
1	Select the files and directories whose permissions you want to change.	The items are highlighted.
2	Right-click one of the items you selected.	A pop-up menu appears.
3	Choose Properties from the pop-up menu.	The Properties dialog box appears.
4	Click the SecureShare tab.	SecureShare Access appears.

Changing the permissions of a single item

If you select only one item, SecureShare Access appears as follows:



Caution

SecureShare Access has no undo feature. Use it very carefully.

Qtree security style effects: The security style of the qtree from which you select an item has the following effects:

- In NTFS-style qtrees, all the edit fields and check boxes are disabled. Use standard Windows NT tools to manipulate NTFS files.
- In NFS-style qtrees, the Has ACL check box is meaningless and is disabled.
- In mixed-style qtrees, if a file has an ACL, the ACL is removed when you click the OK button. A dialog box prompts you to confirm that this is what you want.

Recursive application of changes: If one or more of the files is a directory, the Descend Into Subdirectories check box is enabled. Select this check box to apply the changes you make recursively to the contents of any directory you selected. Links are not changed, but their targets are.

Changing the permissions of multiple items

You can change the permission of several items at once by selecting them, then displaying SecureShare Access by following Steps 2 through 4 in “Displaying SecureShare Access” on page 203.

In the multiple-item display, the following conditions are in effect that are different than the single-item display.

- Permissions or attributes, for example, whether an item is a directory, that are common to all the items you selected appear with a white background.
- Permissions or attributes, for example, whether an item is a directory, where one item differs from the rest of the items you selected appear with a light gray background.
- Permissions that you set apply to all items you selected.
- Turning one of the check boxes gray preserves the permissions on any of the selected items.

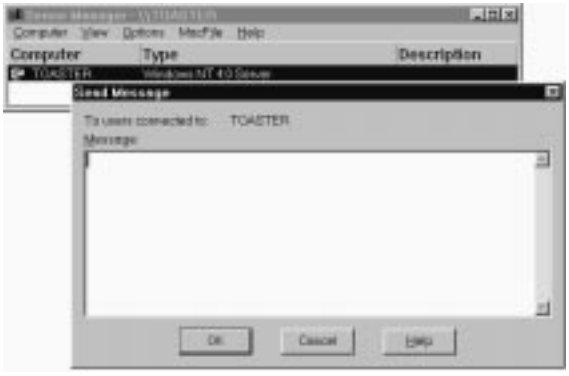
Sending a message to all users on a filer

When to send a message

You might want to send a message to all users on a filer to tell them of important events. The message appears in an alert box. For example, you might need all users to close any open files on the filer, but not want to stop CIFS services and therefore not take advantage of the messaging function of the `cifs terminate` command.

How to send the message

To send a message to all users on a filer, follow these steps:

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK.
4	From the Computer menu, choose Send Message. The Send Message window, shown below, appears. 
5	Enter a message in the text box of the Send Message window.
6	Click OK to send the message.

Using oplocks

What oplocks do

Oplocks (opportunistic locks) enable the redirector on a CIFS client in certain file-sharing scenarios to perform client-side caching of read-ahead, write-behind, and lock information. A client can then work with a file (read or write it) without regularly reminding the server that it needs access to the file in question. This improves performance by reducing network traffic.

When to use oplocks

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.

Data loss possibilities: 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.

Error handling and write completion: The cache itself does not have any error handling—the applications do. When the application makes a write to cache, the write is always completed. If the cache, in turn, makes a write to the target system over a network, it must assume that the write is completed because if it does not, the data is lost.

When to turn oplocks Off

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; that is, you have a good network but you cannot afford even the slightest data loss.

Otherwise, you can leave CIFS oplocks On.

Turning oplocks On and Off globally

You can turn CIFS oplocks On or Off globally for the entire filer or for individual qtrees, which are special directories that are described in detail in Chapter 11, "Administering Qtrees."

Turning oplocks Off: You turn all CIFS oplocks Off with the following `options` command:

```
options cifs.oplocks.enable off
```

Turning oplocks On: You turn CIFS oplocks On with the following `options` command:

```
options cifs.oplocks.enable on
```

Turning oplocks On or Off at individual clients

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.

For additional information: For additional information about oplocks, consult the Microsoft Knowledge Base at *http://www.microsoft.com/kb.*

Displaying CIFS statistics

How and why to display CIFS statistics

You use the `cifs stat` command to display statistics about CIFS operations that take place on your filer. You use the `cifs stat` display for diagnostic purposes.

Statistics displays with the cifs stat command

You can use the `cifs stat` command in two forms:

- If you specify a time interval, the command displays statistics at the specified intervals.
- If you do not specify a time interval, the command displays CIFS statistics that have accumulated since the last reboot.

Example of cifs stat output: The following command displays statistics every second:

```
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
```

The following table describes the fields. The figures are aggregate operation counts.

Field	Description
`GetAttr`	Attribute operations.
`Read`	Read data operations.
`Write`	Write data operations.
`Lock`	Lock operations.
`Open/Cl`	Open and close operations.
`Direct`	Directory operations.
`Other`	Other operations, such as deletes and logoffs.

Displaying CIFS session information

CIFS session information you can display

You can display information about connected users and the number of shares and open files for each user. You can also display information about a specific connected user.

Displaying information with the cifs sessions command

The `cifs sessions` command syntax is as follows:

```
cifs sessions <username>
```

Displaying information about all connected users: To display information about all connected users, use the following command syntax:

```
cifs sessions
```

Sample output is

```
Server Registers as 'TOASTER' in group WNT-DOMAIN
WINS Server: 272.320.0.4
PC style Access Control is being used
Using domain controller WNT-DOMAIN-PDC for authentication
=======================================
PC (user)           #shares       #files
SMITHPC (qsmith)      1             1
PETERSPC (zpeters)    2             3
```

Displaying information about one user: To display the information for a particular user, specify the user name in the command; for example:

```
cifs sessions ghopper

users
    shares/files opened

  HALEY-HOME1   (ghopper)
    ENG-USERS
              Read-denyW   -
\GHOPPER\SRC\PROD\COMMON\HTTPD\httpd_fast.c


  HALEY-PC      (ghopper)
    ENG-USERS
```

Displaying connected user security information: To display security information for each connected user, use `cifs sessions` with the `-s` option. After the first two lines, detailed information for each connected user is displayed. The following example lists only one user.

```
cifs sessions -s

users
  Security Information

HOLARD-PC     (root)
        ***************
        UNIX uid = 0
        user is a member of group daemon (1)
        user is a member of group www (204)
        user is a member of group well (0)
        user is a member of group http (500)

        NT membership
                WNT-DOMAIN\root
                WNT-DOMAIN\Domain Users
                WNT-DOMAIN\Domain Admins
                WNT-DOMAIN\SU Users
                WNT-DOMAIN\Installers
                BUILTIN\Users
                BUILTIN\Administrators
        User is also a member of Everyone, Network Users,
        Authenticated Users
        ***************
```

Stopping and restarting CIFS sessions

Ways to stop CIFS sessions

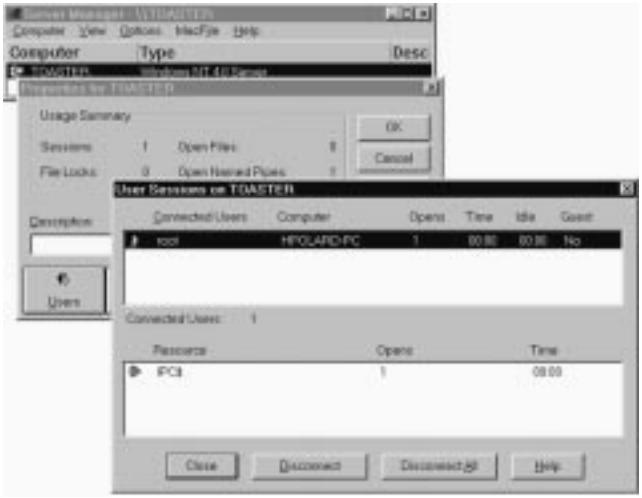
If you want to stop CIFS sessions for all clients or for a single client, do one of the following:

- From Windows NT, use Server Manager as described in the next section, "Disconnecting users with Server Manager."
- From the filer command line, use the `cifs shares` command to display share information, as described in "Using the cifs terminate command" on page 213.

Disconnecting users with Server Manager

To stop CIFS sessions with Server Manager, follow these steps.

Step	Action
1	Open Server Manager.
2	Choose Select Domain.
3	Select the filer you want by typing its UNC name, for example, \\FILERNAME, in the Domain field, then clicking OK. **Result:** A Server Manager window for the filer appears.
4	Double-click the name of a filer. **Result:** The Properties window for the filer appears.

Step	Action
5	Click Users. **Result:** The User Sessions window appears, as shown below. 
6	To disconnect one or more users, do one of the following: ◆ To disconnect a single user or selected users, select them, then click Disconnect. ◆ To disconnect all users, click Disconnect All. **Result:** The selected users are disconnected. **Note** If at least one of the selected users has open resources, an alert box appears for you to confirm or cancel your command.

Using the cifs terminate command

You can stop CIFS service for a specific client or shut down CIFS service from the filer by using the `cifs terminate` command. Always terminate all CIFS sessions before you reboot or turn Off the filer.

You can specify a single client or all clients, and the time delay, in minutes, before the CIFS sessions are terminated, as shown in the following command syntax:

```
cifs terminate [client] [[-t] time]
```

Variable	Description
client	Name of the client for which you are ending a CIFS session.
time	Number of minutes before the termination happens.

The cifs terminate command not persistent

The `cifs terminate` command disables CIFS sessions 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 automatically starts CIFS service. If you want to prevent CIFS from restarting after a reboot, remove the */etc/cifsconfig.cfg* file from the filer.

Time delay

You can delay the termination of CIFS service after you enter the `cifs terminate` command.

Default time delay: The default time delay is five minutes.

Changing the time delay: When you use the `-t` option, the command counts down from the time specified.

Canceling the cifs terminate command

If you want to cancel the `cifs terminate` command, press Ctrl-C before the end of the countdown.

Note

The `halt` command automatically invokes the `cifs terminate` command.

Caution

The `reboot` command also stops CIFS service; however, it does not provide a time delay during which users can save their open files before the disconnect. 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.

Examples of the cifs terminate command

Here are some examples of the `cifs terminate` command.

Terminating CIFS service for all users on the filer: To terminate CIFS service for all users on the filer after 10 minutes, enter the following command:

```
cifs terminate -t 10
```

The `cifs terminate` command displays an alert message on CIFS clients that warns them of the pending shutdown of CIFS service.

Note

Windows 95 and Windows for Workgroup clients must have the WinPopup program configured before they can display the alert message.

Console display: Here is the display for `cifs terminate` when you do not specify a time:

```
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
```

Terminating a CIFS session for a specific client: To terminate a CIFS session for a particular client, specify the name of the computer in the command. For example, the following command terminates a CIFS session for a computer named PETERSPC after 10 minutes:

```
cifs terminate PETERSPC -t 10
```

Using the cifs restart command to restart CIFS service

To restart CIFS service, use the `cifs restart` command, as follows:

```
cifs restart

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

Reconfiguring the filer for CIFS

When to reconfigure a filer for CIFS

You can reconfigure the filer for CIFS service at any time, for example, if you want to change the authentication method from PDC (Primary Domain Controller) to UNIX password database.

How to reconfigure a filer for CIFS

To reconfigure a filer for CIFS, follow these steps:

1. Enter the `cifs terminate` command to stop CIFS service.
2. Enter the `cifs setup` command to reconfigure CIFS service.
   - If you enter only valid information, when you exit the program, the filer automatically restarts CIFS using the new configuration information.
   - If you enter some invalid information, for example, you mistype a domain name, when you exit the program, the filer restarts CIFS with the previous configuration.

For more information about the prompts displayed in `cifs setup`, refer to *Getting Started*.

HTTP Administration 8

Chapter contents

This chapter describes how to perform the following tasks.

Read...	To learn about...
"Starting HTTP service" on page 218	Starting HTTP service.
"Protecting Web pages with passwords" on page 220	Protecting Web pages with passwords.
"Using the HTTP virtual firewall" on page 223	Using the HTTP virtual firewall.
"Using virtual hosting" on page 224	Enabling virtual hosting.
"Specifying MIME Content-Type values" on page 226	Specifying MIME Content-Type values.
"Translating URLs" on page 228	Translating URLs.
"Displaying HTTP connection information" on page 231	Displaying HTTP connection information.
"Displaying HTTP statistics" on page 232	DISPLAYING HTTP statistics.

Note

You can use the filer as an HTTP server *only* if you purchased the license for HTTP. Without the license, you can use an HTTP client (Web browser) only to display the filer's man pages and to use FilerView.

Starting HTTP service

Procedure for starting HTTP service

To start HTTP service on your filer, follow these steps:

Step	Action
1	Enable the `httpd` daemon by entering the following 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 */vol/vol0/home/users/pages*, enter the following command: **`options httpd.rootdir /vol/vol0/home/users/pages`**
3	If you want to limit the size of the */etc/log/httpd.log* log file to other than the default of 2,147,483,647 bytes (2 GB minus 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 starting HTTP service is now complete, and clients can display text files under the root directory 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 226 to configure your filer so that the appropriate MIME Content-Type header is sent with each file transferred.

Procedure for testing HTTP service

To test the filer's HTTP service, follow these steps.

Step	Action
1	Create an HTML file in the root directory for HTTP. For example, create a file named *myfile.html* in the HTTP root directory, which is */vol/vol0/home/users/pages*, assuming that the HTML root directory is */vol/vol0/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 is *toaster* and the root directory for HTTP is */vol/vol0/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 *index.html* file within the directory. If *index.html* does not exist, the filer returns "Error 404. No such file or directory."

Protecting Web pages with passwords

Configuration files for password protection

You can restrict access to a specified directory so that only specified users or groups have access to it.

Password protection involves three configuration files:

- */etc/httpd.access*
- */etc/httpd.passwd*
- */etc/httpd.group*

The /etc/httpd.access file

The */etc/httpd.access* file contains directives that govern authentication for each directory. 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: Specifies a directory tree to be protected and encloses all other directives. The syntax of the `directory` directive is as follows:

```
<Directory directory_name>
directive ...
</Directory>
```

The AuthName directive: Specifies a "realm," that is, an alias for the directory that 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 as follows:

```
AuthName realm name
```

The require user directive: Specifies the users who can access the directory. The syntax of the `require user` directive is as follows:

```
require user user_id [, user_id, ... ]
```

The require group directive: Specifies the groups that can access the directory. The syntax of the `require group` directive is as follows:

```
require group group_id [, group_id, ... ]
```

The /etc/httpd.passwd file

The */etc/httpd.passwd* file contains the *user_id* and encrypted-password pairs. The pairs have the following 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

The */etc/httpd.group* file 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 */vol/vol0/home/htdocs/private/spec* to only the user bob:

```
<Directory /vol/vol0/home/htdocs/private/spec>
AuthName polard Private Stuff
<Limit GET>
require user bob
</Limit GET>
</Directory>
```

The <Limit GET> and </Limit GET> directives, which might have been imported from an Apache or NCA Web server, are not supported. 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 sample procedure restricts user access to a particular directory:

1. Enter the following lines in the */etc/httpd.access* file:

```
<Directory /vol/vol0/home/htdocs/private/specs>
```

```
AuthName Social commentary
require group engineering
</Directory>
```

2. Enter the following line in */etc/httpd.group*:

```
engineering: bob larry nancy rose
```

The */vol/vol0/home/htdocs/private/specs* directory is now accessible only to the group engineering, which consists of the following user IDs:

- bob
- larry
- nancy
- rose

Using the HTTP virtual firewall

About the HTTP virtual firewall

The HTTP virtual firewall feature enables you to maintain security on your filer.

You can 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.

Mark an interface untrusted if it meets all 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.

Syntax

Mark an interface as untrusted or trusted by setting an option to the `ifconfig` command. Following are examples of the command:

- To mark the *f0* interface as untrusted, enter the following command:

  ```
  ifconfig f0 untrusted
  ```

- To mark the *f0* interface as trusted, enter the following command:

  ```
  ifconfig f0 trusted
  ```

Using virtual hosting

About 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 set up and enable virtual hosting

To enable virtual hosting, you

- direct HTTP requests by putting subdirectory and host or address entries in the */etc/httpd.hostprefixes* file
- map virtual host addresses to the virtual host interface with the `ifconfig` command

Directing HTTP requests

To direct HTTP requests, use the following format in the */etc/httpd.hostprefixes* file:

prefix [*host-name-or-address ...*]

Variable	Description
prefix	Specifies a subdirectory in the HTTP root directory, which is defined by the `options httpd.rootdir` command.
host-name-or-address	Specifies 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 that 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 */customer* directory.

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:

- 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.

- 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

About 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, or example, *.gif*, *.html*, or *.mpg*, according to information in the */etc/httpd.mimetypes* file.

Note

On a Windows 95 or Windows NT 4.0 client, the */etc/httpd.mimetypes.sample* file name 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 check box next to "Hide MS-DOS file extensions for file types that are registered."

Modifying MIME Content-Type mappings

To modify MIME Content-Type mappings or to add MIME Content-Types, edit the entries in */etc/httpd.mimetypes*. Entries are in the following format:

```
# An optional comment.
suffix   Content-Type
```

Lines preceded by the # sign are comments. The file name suffix is case-insensitive. Following are sample entries:

```
# My clients' browsers can now use
# PICT graphics files.
pct        image/pict
pict       image/pict
```

In the sample entries, files whose names ended with *.pact* or *.pact* are mapped to the MIME Content-Type value of image/pict. 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

How the filer responds to 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 the configuration file contains up to three fields in the following format:

```
rule template result
```

Field	Description
`rule`	Defines the response of the filer to a request.
`template`	Specifies a component of a URL.
`result`	Depends on the rule, as described in the following section.

Translation rules supported by the filer

This section explains the meanings of the 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, the following */etc/httpd.translations* entry causes any requests to a URL containing the */image-bin* directory to be mapped to the */usr/local/http/images* directory:

```
map /image-bin/* /usr/local/http/images/*
```

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. The `result` field for the redirect rule must be specified as a complete URL beginning with *http://* and the host name.

For example, if */etc/httpd.translations* contains the following entry

```
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.

The pass rule: The pass rule specifies that if a component of a URL matches the template, the filer accepts the request, processes the request as is, and disregards other rules.

For example, if */etc/httpd.translations* contains the following entry

```
pass /image-bin/*
```

the filer processes 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 accepts the request, processes the request by using the URL defined in the `result` field, and disregards other rules.

The fail rule: The fail rule specifies that if a component of a URL matches the template, the filer denies access to that component and disregards other rules.

For example, if */etc/httpd.translations* contains the following entry

```
fail /usr/forbidden/*
```

the filer does not provide access to the */usr/forbidden* directory.

How the filer processes rules

The filer processes the rules defined in */etc/httpd.translations* in the order they are listed, and applies the rule if the URL matches the template. However, the filer stops processing other rules after it applies a pass or fail rule.

In the `template` or `result` field of an */etc/httpd.translations* entry, you can use asterisks (*) as wildcard characters, as follows:

- In the `template` field, the wildcard character matches zero or more characters, including the slash (/) character.
- In the `result` field, the wildcard character represents the text expanded from the match in the `template` field. Include the wildcard character in the `result` field only if you used a wildcard character in the `template` field.

- If you use multiple wildcard characters, 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.

Following is an example showing how a wildcard 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 are redirected to another host named cgi-host.

For example, if the filer receives the following requests

```
http://toaster/cgi-bin/displayfares

http://toaster/cgi-bin/displaydates
```

the filer expands the wildcard character to displayfares and displaydates and redirects the requests to the host named cgi-host. To the client, the results of these requests are the same as the results of the following requests:

```
http://cgi-host/cgi-bin/displayfares

http://cgi-host/cgi-bin/displaydates
```

Displaying HTTP connection information

Information in the /etc/log/httpd.log file

You can 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.
- 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 the MIME header.

Following is 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
```

Displaying HTTP statistics

httpstat statistic types

The `httpstat` command displays four types of statistics about HTTP operations on the filer:

Column	Description
`gets`	Successful requests for files.
`badcalls`	Requests for nonexistent files.
`open conn.`	Number of HTTP connections currently open.
`peak conn.`	Largest number of simultaneous HTTP connections since the filer was booted or since the `-z` option was used.

Syntax

The syntax for the `httpstat` command is as follows:

`httpstat [ -t|-z ][ interval ]`

If you use no 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. Following is an example of `httpstat`:

```
httpstat
HTTPD statistics:
gets         badcalls         open conn.   peak conn.
451          11                5            17
```

Cluster Administration 9

Introducing clusters

About clusters

A cluster consists of a pair of NetApp filers that are connected and configured in a special way to provide fault tolerance. When one filer in a cluster undergoes a system failure and can't reboot, the partner filer in the cluster can take over the failed filer's functions and serve the data on the failed filer's disks. This is known as a takeover. In addition to an automatic takeover, an operator may issue a `cf takeover` command at any time to force a takeover.

The filers in a cluster have the following characteristics:

- They are connected to each other through a Cluster Interconnect adapter through which they:
  - constantly check whether the other filer is functioning
  - mirror log data each other's NVRAM
  - synchronize each other's time
- They use two Fiber Channel Arbitrated Loops (FC-AL)
  - Each filer uses the A loop to manage its own disks
  - Each filer uses the B loop in takeover mode to manage its partner's disks
- They own their spare disks and do not share them with the other filer.

If one filer in a cluster stops functioning physically, its partner filer:

- creates a virtual filer using the failed filer's identity
- accesses the failed filer's disks and serves its data to clients
- maintains its own identity and continues to serve its own data to clients

When the failed filer is functioning again:

- An operator initiates a giveback that terminates the virtual filer on the partner
- The failed filer resumes normal operation, serving its own data
- The cluster resumes normal operation, with each filer ready to take over for its partner if the partner fails

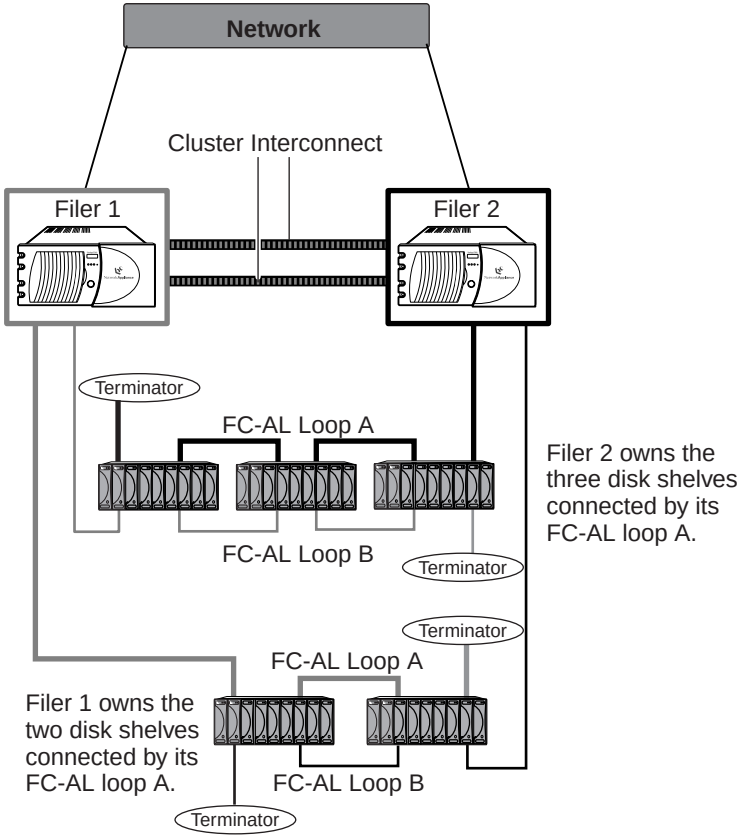
Chapter contents

This chapter discusses the topics described in the following table.

Read...	To learn about...
"Understanding cluster concepts" on page 235	What clusters are and how they work.
"Understanding takeover" on page 239	How the takeover process works and its result.
"Understanding giveback" on page 242	How the giveback process works.
"Managing a cluster" on page 244	How to manage a cluster in general.
"Managing the cluster in normal mode" on page 245	How to manage a cluster when no takeover has taken place.
"Managing takeover" on page 247	How to manage a cluster after a takeover has taken place.
"Managing filers in takeover mode" on page 249	How to manage the filer that has taken over its partner.
"Managing a virtual filer" on page 251	How to manage the virtual filer that represents the filer that has been taken over.
"Performing a giveback" on page 256	How to restore a cluster to normal operation.

Cluster diagram

The following figure diagrams how the cluster hardware components are connected.



Cluster components and terminology

The following table describes components and explains terms that apply to clusters.

Term	Explanation
cluster	A pair of F630 filers connected so that one filer can detect when the other is not working and, if so, can serve the failed filer's data.
cluster interconnect	Cables and adapters with which the two filers in a cluster are connected and over which heartbeat and WAFL™ log information are transmitted when both filers are running.
cluster monitor	Software that administers the relationship of filers in the cluster. You access the cluster monitor with the `cf` command.
failed filer	The physical filer that has ceased operating. It remains the failed filer until giveback succeeds.
FC-AL disk shelves	In normal operation, each filer in a cluster owns the set of disk shelves that are connected to it through the FC-AL A loop.
giveback	The return of identity from the virtual filer to the failed filer, resulting in a return to normal operation; the reverse of takeover.
heartbeat	A repeating signal transmitted from one filer to the other that indicates that the filer is in operation. Heartbeat information is also stored on disk.
takeover filer	A filer that remains in operation after the other filer stops working and that hosts a virtual filer that manages access to the failed filer's disk shelves and network connections. The takeover filer maintains its own identity and the virtual filer maintains the failed filer's identity.
local filer	The filer you are logged in to.

Term	Explanation
mailbox disk	One of a set of disks owned by each filer that is used to store the cluster state information of the filer. If that filer stops operating, the takeover filer uses the information in the mailbox disks in constructing a virtual filer. Mailbox disks are also used as filesystem disks.
normal mode	The state of a filer when there is no takeover in the cluster.
NVRAM mirror	A synchronously updated copy of the contents of a filer's NVRAM (Non-Volatile Random Access Memory) contents kept on the partner filer.
partner	From the point of view of the local filer, the other filer in the cluster.
partner mode	The method you use to communicate through the command line interface with the virtual filer during a takeover. Partner mode is accessed using the `partner` command.
takeover	The emulation of the failed filer's identity by the takeover filer in a cluster.
takeover mode	The method you use to interact with a filer while it has taken over its partner. The console prompt indicates when the filer is in a takeover state.
virtual filer	A software copy of the failed filer that is hosted by the takeover filer. The virtual filer appears to users and administrators like a functional version of the failed filer. For example, it has the same name as the failed filer.

Normal cluster operation

During normal operation, in addition to serving data, the cluster performs the activities described in the following table.

Action	Description
Heartbeat transmission	The heartbeat signals transmitted over the cluster interconnect cable tell each filer the state of the other filer. This information is written to disks on each filer that the cluster monitor designates as mailbox disks. The disks provide redundancy.
Monitoring	The cluster monitor software looks at the heartbeat signal, commands from filers, and disk connections, and initiates a takeover under the following conditions: ◆ The operator initiates a takeover using the cf takeover command ◆ The partner is determined to have stopped functioning due to a lack of heartbeat activity ◆ The partner's power-on-self-test (POST) fails **Note** If the filer that is not transmitting a heartbeat is rebooting, no takeover occurs. Takeover may be disabled if NVRAM mirroring is not functioning.
NVRAM mirroring	Each filer dedicates half of its NVRAM to a synchronously updated copy of its partner's NVRAM. If a takeover occurs, the takeover filer uses the cluster data in the part of the NVRAM dedicated to the failed filer to ensure that no information is lost.
Time synchronization	Filers' times can deviate from each other. A daemon updates the times on filers periodically to keep them synchronized.

Understanding takeover

Takeover process

The takeover process is shown in the following table.

Stage	Description
1	One of these three things happen: ◆ One filer in a cluster undergoes a hardware or power failure and can't reboot. That filer is the failed filer, and the surviving filer is the takeover filer. ◆ An operator issues the `cf takeover` command from a filer. The filer from which the command was issued becomes the takeover filer; the other filer becomes the failed filer. ◆ An operator issues the `halt` command without the `-f` option.
2	The takeover filer reserves the disk drives on its FC-AL B loop (which is the failed filer's FC-AL A loop) so that they do not receive commands from the failed filer.
3	The takeover filer creates a virtual filer that takes over the resources and identity of the failed filer and executes the equivalent of a reboot of the virtual filer. For example, it uses network interface information from the failed filer's configuration files to duplicate the failed filer's network services.
4	The virtual filer goes through a process similar to the reboot process. ◆ All filer information preserved across filer reboot is preserved in a takeover. ◆ Filer information that is lost in a reboot, such as CIFS connection state, is lost in a takeover.
5	Users of the failed filer experience the effects of a reboot of the failed filer, and then client service continues.

Messages indicating a successful takeover

During a typical takeover, the cluster monitor issues messages like the following on the console of the takeover filer. In this case, the partner of filer1 was halted.

```
filer1> Tue Jul 21 23:02:02 GMT [filer1: cf_main]: Cluster monitor:
partner halted
Tue Jul 21 23:02:11 GMT [filer1: cf_main]: Cluster monitor: partner
not responding
```

Timeout messages such as the following appear as filer1 waits for signs of life from its partner:

```
Tue Jul 21 23:02:34 GMT GMT [filer1: cf_main]: Cluster monitor:
takeover scheduled in 30 seconds
```

Then the following messages appear:

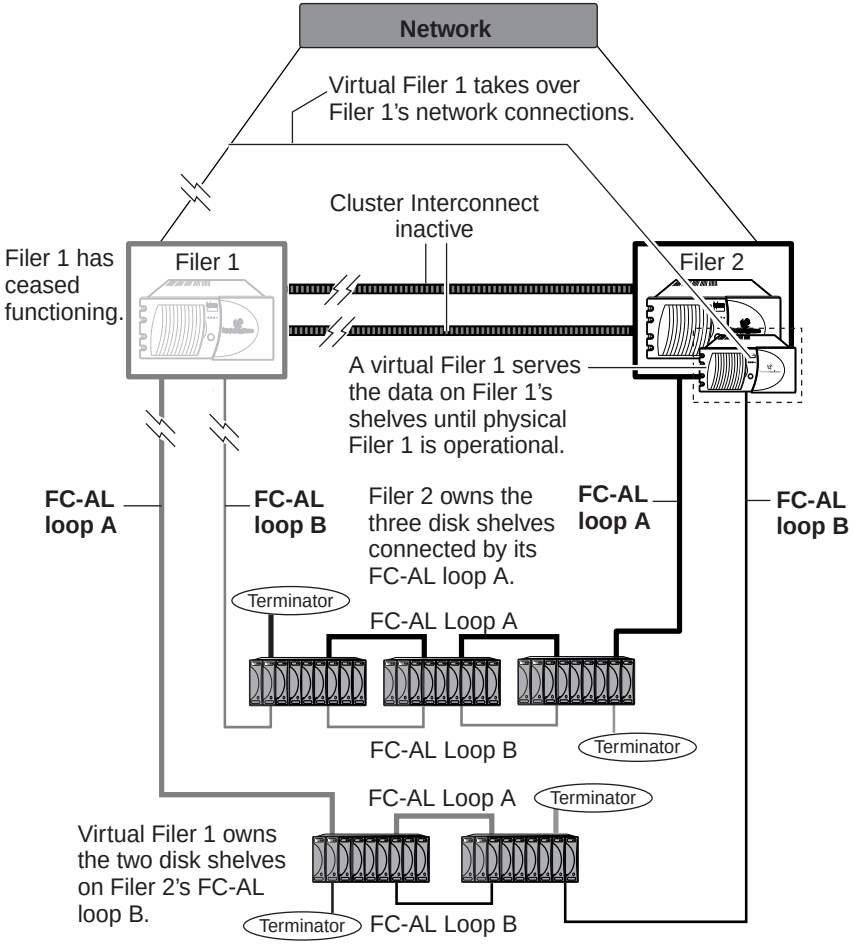
```
Tue Jul 21 23:03:06 GMT [filer1: cf_takeover]: Cluster monitor:
takeover started
```

Reboot messages follow as the virtual filer boots up, then the following message appears and indicates a successful takeover:

```
Tue Jul 21 23:03:22 GMT [filer1: cf_takeover]: Cluster monitor:
takeover completed
filer1(takeover)>
```

Takeover result diagram

The following figure shows the state of a cluster after a takeover of Filer 1 by Filer 2.



Understanding giveback

Giveback process

The giveback process is described in the following table.

Stage	Description
1	An operator issues a `cf giveback` command.
2	The takeover filer updates the failed filer's disks.
3	The takeover filer destroys the virtual filer so that no more writes can occur to the failed filer's disks.
4	The takeover filer releases the disk reservations that it held on behalf of the failed filer. This returns control to the failed filer of its FC-AL A loop disks.

When the failed filer comes up, it returns to normal cluster operation, and the users of that filer experience the effects of a reboot.

Note

Giveback occurs on the takeover filer and does not depend on the state of the failed filer.

Messages indicating a successful giveback

During a typical giveback, the cluster monitor issues messages like the following on the console of the takeover filer:

```
filer1(takeover)> Tue Jul 21 22:37:34 GMT [filer1: rc]: Cluster
monitor: giveback initiated by operator
Tue Jul 21 22:37:34 GMT [filer1: cf_giveback]: Cluster monitor:
giveback started
Tue Jul 21 22:37:35 GMT [filer1: cf_giveback]: Cluster monitor:
giveback completed
```

The preceding message indicates that the giveback process was successful. When the failed filer reboots, the following messages appear:

```
Tue Jul 21 22:38:35 GMT [filer1: cf_main]: Cluster monitor: partner
ok
```

```
Tue Jul 21 22:38:35 GMT [filer1: cf_main]: Cluster monitor:
takeover of partner enabled
```

The last message indicates that the filers are operating normally and therefore either filer can take over for its partner if necessary.

Note

After a giveback, the takeover filer's ability to automatically take over its partner is not reenabled until the partner reboots successfully. If the partner fails to reboot, you can enter the `cf takeover` command to manually initiate a takeover of the partner.

Forcing a giveback

There are two variations of the cf giveback command that force a giveback to occur:

- `cf giveback -f`
- `cf forcegiveback`

cf giveback -f: The `cf giveback` command will not perform giveback if the following long running or stateful operations are underway on the virtual filer:

- parity check
- disk scrub
- active cifs sessions
- dump or restore
- kernel download
- snapshot creation or deletion
- quota initialization

If the `cf giveback` command fails because any of the above operations are running, you can force a giveback by entering the `cf giveback -f` command. The cf giveback -f command forces the long running and stateful operations to terminate.

cf forcegiveback: In the event that the filer panics during the giveback process, you may use the `cf forcegiveback` command to release the partner's identity and resources.

Caution

Using the cf forcegiveback command may result in the loss of recently written data. The command should be used only under extreme circumstances.

Managing a cluster

What managing a cluster entails

Managing a cluster entails tasks described in the following topics:

- "Managing the cluster in normal mode" on page 245. You can enable or disable takeover, and monitor the cluster in normal mode to verify that it is functioning properly.
- "Managing takeover" on page 247. You can initiate a takeover.
- "Managing filers in takeover mode" on page 249. After a takeover, you can administer the takeover filer, perform maintenance on the failed filer, and initiate a giveback.
- "Managing a virtual filer" on page 251. You manage the virtual filer with the `partner` command.
- "Performing a giveback" on page 256. You can initiate or force a giveback.

Managing the cluster in normal mode

Cluster management overview

You manage a cluster in normal mode by performing the following actions:

- Monitoring the status of the cluster as a whole with the `cf status` and `cf partner` commands, as described in "Monitoring cluster status" and "Displaying the partner's name" on this page.
- Viewing information about the cluster, as described in "Displaying cluster configuration information" on this page.
- Controlling whether time in a cluster is synchronized, as described in "Controlling time synchronization" on page 246.
- Performing license operations on the cluster feature, as described in "Licensing and unlicensing the cluster feature" on page 246.

Monitoring cluster status

To find out whether the cluster feature is enabled and whether the other filer in the cluster is up, enter **`cf status`**.

A display like the following appears:

```
filer1> cf status
Cluster enabled, filer2 is up.
```

This tells you that both filer1 and filer2 are functioning and available for takeover.

If you see a `Cluster disabled` message, it might mean that

- you disabled the cluster feature. To reenable the cluster feature, use the `cf enable` command.
- the system disabled clustering because the interconnect failed.

Displaying the partner's name

To find out the name of the other filer in the cluster, enter **`cf partner`**.

Displaying cluster configuration information

To find out about cluster configuration, use the `sysconfig` or `vol status` command. In a cluster, the `sysconfig` and `vol status` displays are special in that they display information about both filers.

For each filer, the `sysconfig` command display includes disks on both FC-AL loop A and FC-AL loop B.

- The information about disks on FC-AL loop A is the same as for filers not in a cluster.
- The information about disks on FC-AL loop B is for hardware only; the `sysconfig` command displays information about the adapters supporting the disks. The command does not show whether a disk on FC-AL loop B is a file system disk, spare disk, or parity disk.

Controlling time synchronization

You can disable time synchronization on a filer or in a cluster. Normally, both filers should have the same value for this option.

- To disable time synchronization on a filer, use the following command:
  ```
  options cf.timed.enable off
  ```
- To enable time synchronization on a filer, use the following command:
  ```
  options cf.timed.enable on
  ```

Licensing and unlicensing the cluster feature

You can license or unlicense the cluster feature only in normal mode with takeover disabled. Complete these steps to unlicense the cluster feature.

Step	Action
1	If the filer has taken over its partner, enter the following command to perform a giveback: **`cf giveback`**
2	Enter the following command to disable clustering: **`cf disable`**
3	Enter the following command to unlicense clustering: **`license cluster=disable`**
4	Enter the following command to reboot the filer and make the change take effect: **`reboot`**

Managing takeover

Takeover management tasks

You manage takeover by performing either of the following tasks:

- Enabling and disabling takeover for the cluster, as described in the next section, "Enabling and disabling takeover."
- Initiating a takeover of the partner by the local filer, as described in "Initiating a takeover" on page 247.

Enabling and disabling takeover

You might want to disable takeover if you are doing maintenance work that normally causes a takeover, then reenable takeover when you have finished.

- You disable takeover on a cluster with the `cf disable` command.
- You enable takeover on a cluster with the `cf enable` command.

You can issue this command from either filer in the cluster.

Halting a filer without a takeover

To halt a filer and prevent its partner from taking over, enter the following command:

```
halt -f
```

This is useful if you need to perform maintenance on both the filer and its disks.

Initiating a takeover

You might want to initiate a takeover to do maintenance work on a filer in the cluster while still serving the data on the disks of that filer.

You can initiate a takeover in the two ways described in the following table.

Method	Description
`cf takeover`	Initiates a takeover of the partner of the local filer. Takeover is aborted if time-consuming operations, such as a dump, are in progress.
`cf takeover -f`	Initiates a takeover of the partner of the local filer regardless of whether time-consuming operations, such as a dump, are in progress.

Method	Description
`cf forcetakeover`	Tells the cluster monitor to ignore some configuration problems that otherwise prevent a takeover, such as unsynchronized NVRAM due to a faulty interconnect connection. It then initiates a takeover of the partner of the local filer. **Caution** Use `cf forcetakeover` only in certain controlled circumstances, when you cannot get `cf takeover` to succeed. You can lose uncommitted data with this command.

Managing filers in takeover mode

Takeover mode tasks

You can do the following tasks in takeover mode:

- Administer the local filer as you would any normal filer. Take into account the minor differences between a filer in normal mode and a filer in takeover mode described in the next section,"Takeover mode differences from normal mode."
- Enter partner mode and administer the virtual filer, as described in "Managing a virtual filer" on page 251.
- Initiate or force a giveback, as described in "Performing a giveback" on page 256.

Takeover mode differences from normal mode

The following subsections describe differences between managing a filer in normal mode and doing so in takeover mode.

Takeover mode prompt: In takeover mode, the prompt looks like this:
`filer1(takeover)>`

This indicates that filer1 is in takeover mode.

Statistics in takeover mode: Statistics are treated differently in the following commands:

`httpstat`

`ifstat`

`netstat`

`nfsstat`

`sysstat`

In takeover mode, statistics for the preceding commands differ from the statistics in normal mode in the following ways:

- Each display reflects the sum of operations that take place on the takeover filer plus the operations on the virtual filer. Therefore, the display does not differentiate between the operations on the takeover filer and the operations on the virtual filer.
- The statistics displayed by each of these commands are cumulative.

- After giving back the failed partner's resources, the takeover filer does not subtract the statistics it performed on behalf of the failed filer in takeover mode.
- The giveback does not reset (zero out) the statistics.

  If you want accurate statistics from a command after a giveback, reset (zero out) the statistics as described in the man page for the command you are using.

SNMP in takeover mode: You can have different settings on each filer for SNMP options, but any statistics gathered while a filer has been taken over does not distinguish between filers.

Managing a virtual filer

Virtual filer management overview

You manage a virtual filer as you do any filer, including managing disks, with the following exceptions, which are described in greater detail later in this section:

- A virtual filer can access only failed filer disks.
- Some commands are unavailable.
- Some displays differ from normal displays.

Accessing the virtual filer from the takeover filer

You access the virtual filer from the takeover filer in takeover mode with the `partner` command. You can issue the `partner` command in two forms:

- The `partner` command without an argument toggles between partner mode, in which you manage the virtual filer, and takeover mode, in which you manage the takeover filer. The prompt changes to the partner mode prompt, which has the form:

  virtual_filer/*takeover_filer*>,

  The following example shows the change from takeover mode to partner mode and back:

  ```
  filer1(takeover)> partner
  Login from console: filer2
  filer2/filer1> Thu Aug 20 16:44:39 GMT [filer1:
  rc]: Login from console: filer2
  partner
  Logoff from console: filer2
  filer1(takeover)> Thu Aug 20 16:44:54 GMT [filer1:
  rc]: Logoff from console: filer2
  filer1(takeover)>
  ```

  The `partner` command with a filer command as an argument executes the filer command on the virtual filer in partner mode, then returns to takeover mode, as shown in the following example:

  ```
  filer1(takeover)> partner cf status
  filer2 has been taken over by filer1.
  filer1(takeover)>
  Accessing the virtual filer remotely
  ```

You can also access the virtual filer using a remote shell program such as `rsh` without using the `partner` command.

Using Telnet: With Telnet, you can access directly only the takeover filer. To access the virtual filer, log in to the takeover filer, then use the `partner` command.

Using a remote shell: To execute a command on the virtual filer using a remote shell, issue the command to that filer, which has the same name as the failed filer.

Virtual filer tape drives inaccessible

You cannot access tape devices on the virtual filer. You must use the tape drives on the takeover filer instead.

Virtual filer commands

Almost all the commands that are available to a normal filer are available on the virtual filer in partner mode. The commands in the following subsections are exceptions.

Unavailable commands: The following commands are not available on a virtual filer:

`cf disable`

`cf enable`

`cf forcegiveback`

`cf forcetakeover`

`cf giveback`

`cf takeover`

`date` *`date`*

`halt`

`ifconfig partner`

`ifconfig -partner`

`ifconfig mtusize`

`license cluster`

`rdate`

`reboot`

`timezone` *`name`*

Commands with different behaviors: The commands in the following table behave differently in partner mode than they do on a takeover filer.

Command	Difference
`ifconfig [`*`interface`*`]`	Displays the following: ◆ Virtual filer interface mappings based on the failed filer's */etc/rc* file. ◆ Virtual filer's interface names rather than the takeover filer's interface names that have performed the takeover. ◆ Only interfaces that have been configured.
`mt`	Uses the tape devices on the takeover filer. This is because the failed filer has no access to its tape devices.
`netstat -i`	Appends a plus sign (+) to shared interfaces. A shared interface is one that has two IP addresses assigned to it: an IP address for the filer in which it physically resides, and an IP address for it's partner filer in the cluster.
`shelfchk`	Does not distinguish between the takeover filer's disk shelves and the virtual filer's disk shelves. The `shelfchk` command has the same result in partner mode as it does in takeover mode.
`sysconfig`	When it displays hardware information, the `sysconfig` command displays information only about hardware that is attached to the takeover filer. It does not display information about any hardware that is attached only to the failed filer. For example, the disk adapter information that the `partner sysconfig -r` command displays is about the disk adapters on the takeover filer.
`uptime`	Displays how long the failed filer has been down and the host name of the takeover filer.

Command	Difference
`vol status`	When it displays hardware information, the `vol status` command displays information only about hardware that is attached to the takeover filer. It does not display information about any hardware that is attached only to the failed filer. For example, the disk adapter information that the `partner vol status -r` command displays is about the disk adapters on the takeover filer.

Backing up and restoring from the virtual filer

You can perform dumps and restores on the virtual filer. However, neither the takeover filer nor the virtual filer can access the tape drives or other peripheral devices on the failed filer.

Performing dumps on a virtual filer: Any dump commands directed to the failed filer's tape drives are executed on the takeover filer's tape drives. Therefore, any `dump` commands that you execute through the `cron` command succeed only under the following conditions:

- The device names are the same on both filers in the cluster.
- The dump commands for the filer and the virtual filer are not scheduled to occur during the same time period—the filer and the virtual filer cannot access the tape drives simultaneously.

Caution

Make sure there that is a tape on which you want the backup to be written in the tape device for each dump. Any tape in the tape device can get overwritten by a dump.

Performing restores on a virtual filer: To perform restores to the virtual filer, follow these steps:

Step	Action
1	Put the restore tape in a drive accessible by the takeover filer.
2	Use the `restore` command in partner mode or as an argument to the `partner` command.

Verifying the link between clustered filers

You can verify the status of the link between clustered filers by entering the following command:

```
cf status
```

The following table shows the messages that may be displayed by the `cf status` command.

Message	Meaning
`Cluster enabled, partner_name is up.`	The cluster is operating normally.
`partner_name_1 has taken over partner_name_2.`	One filer in the cluster has taken over the other filer in the cluster.
`Interconnect not present.`	The system does not recognize the existence of an interconnect device.
`Interconnect is down.`	The interconnect is unable to access the partner. This may be due to cabling problems or the partner may be down.
`Interconnect is up.`	The interconnect is active and is able to transmit data to the partner.
`X link is down; Y link is up.`	The X link (upper cable) is currently down. This condition may occur briefly when a cluster node is booting. If this condition persists, check the cable seating.
`X link is up; Y link is down.`	The Y link (lower cable) is currently down. This condition may occur briefly when a cluster node is booting. If this condition persists, check the cable seating.

Note

Even if one of the interconnect links (X or Y) is down, cluster communication continues over the link that is up.

Performing a giveback

Initiating a giveback

To restore a cluster to normal operation, from the takeover filer, enter one of the commands in the following table.

Command	Description
`cf giveback`	Initiates a giveback to the failed filer. **Note** The cf giveback command will not succeed if any of the following events are taking place: ◆ Dump processes ◆ Outstanding CIFS sessions ◆ Parity checks ◆ RAID reconstructs ◆ RAID scrubs ◆ Volume creation
`cf giveback -f`	Terminates the following events on the virtual filer, then initiates a giveback to the failed filer: ◆ Dump processes ◆ Outstanding CIFS sessions ◆ Parity checks ◆ RAID reconstructs ◆ RAID scrubs ◆ Volume creation

Command	Description
`cf forcegiveback`	Forces the takeover filer to give back the resources of the virtual filer even if the takeover filer detects an error that would normally prevent a complete giveback, such as data not being flushed from NVRAM to the failed filer's disks. When the failed filer reboots as a result of a forced giveback, it displays the following message if data from the virtual filer did not fully synchronize with the virtual filer's disks: `partner giveback incomplete, some data may be lost` **Caution** Use `cf forcegiveback` only in desperate circumstances, when you cannot get `cf giveback` to succeed. You can lose uncommitted data with this command.

What happens when giveback is interrupted

If a failure or power outage happens to the takeover filer during the giveback process, when the failure is repaired or power is restored, the giveback process stops and the takeover filer returns to takeover mode. Some operations, such as quota initialization, will always cause a giveback to fail.

Snapshots 10

Chapter contents

This chapter explains the NetApp snapshot system. The following table lists the topics available.

Read...	To learn about...
“Understanding snapshots” on page 260	Snapshots in general.
“Snapshot commands and options” on page 264	Snapshots commands.
“Understanding snapshot disk consumption” on page 269	How snapshots work with disk consumption.
“Managing snapshot disk consumption” on page 273	How to manage snapshot disk usage.
“Accessing snapshots from clients” on page 278	How to access snapshots from NFS and CIFS clients.

Understanding snapshots

What is a snapshot?

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.

Accessing snapshots

Any client of a filer can access snapshots to recover old versions of files; for example, files that were accidentally changed or deleted. The snapshot feature enables users to restore their own files without help, because files in snapshots can be viewed and copied by those who have permission to do so with the original files.

Simplifying tape backup

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. However, 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 you to take the filer off-line.

Snapshots use little disk space

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 about the copy-on-write technique used by snapshots, refer to “How snapshots work” on page 261.

Creating snapshots for your needs

The filer creates and deletes snapshots automatically at preset intervals. You can also create and delete snapshots manually. Each volume on the filer can have up to 20 different snapshots at one time.

Snapshots maintain original file permissions

Snapshot files carry the same permissions and inode numbers as the original files, keeping the integrity of the security system intact. Inodes are data structures that hold information about files on the filer. 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

The inode number for a file in a snapshot is the same as the inode number for the corresponding file in the active file system. As a result, some programs on UNIX clients consider the two files to be the same. For example, if you use the GNU `diff` program to compare the two files, it does not find any differences between them. To make sure that the two files have different inode numbers before the comparison, copy one of the files to another name.

How snapshots work

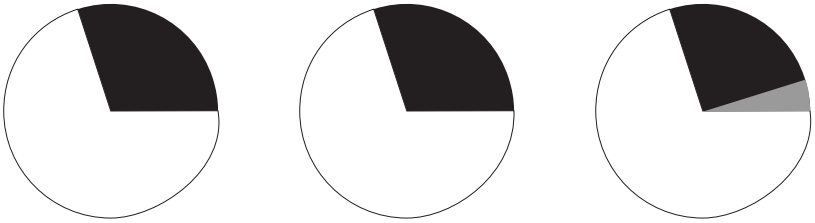
When the filer creates a snapshot, it doesn't copy disk blocks; instead, it identifies all blocks in the file system as belonging to the snapshot as well as to the active file system.

Example: Consider a particular file named *foo* in a newly created snapshot. Because the snapshot was just created, the snapshot version of *foo* has 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. Therefore, deleting *foo* from the active file system does not free any disk space.

Diagram of a snapshot

The following figure illustrates how disk space is used before and after *foo* is removed.



Before any snapshot is taken, disk space is consumed by the active file system only.

After a snapshot is taken, the active file system and snapshot point to the same disk blocks. The snapshot doesn´t use extra disk space.

After the *foo* file is deleted, the space previously used by *foo* is referenced by the snapshot. That´s why deleting active file system data doesn´t free disk space.

Space used by the active file system

Space used by the snapshot only

Unused 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 some directories, most data remains unchanged from day to day. For example, 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 directories, data changes quickly. If a large percentage of data changes every day, there might not be room to keep snapshots for even a few hours. To accommodate the needs of different users, create multiple volumes on the filer. In this way, you can apply different snapshot schedules to different volumes.

In summary, when the filer creates a snapshot, it doesn’t use any disk space, but as files in the active file system are changed or deleted, the snapshot uses more and more disk 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.

Snapshot commands and options

Snapshot commands

The commands related to snapshots are listed in the following table. If the volume name is omitted in any of these commands, the command applies to the root volume.

Command	Meaning
`snap list`[a] *volume_name*	Lists all available snapshots.
`snap create` *volume_name snapshot_name*	Creates a snapshot with a specified name.
`snap delete` *volume_name snapshot_name*	Deletes a specified snapshot.
`snap rename` *volume_name from to*	Renames a snapshot.
`snap reserve` *volume_name*	Reserves a percentage of the disk space for snapshots.
`snap sched` *volume_name*	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 2, "Routine Filer Administration."

Snapshot options

The following options for the `vol options` command affect snapshots in the specified volume. The options remain in effect after the filer reboots.

Options	Descriptions
`nosnap`	Disables automatic snapshots. By default, this option is disabled.
`nosnapdir`	Makes the *.snapshot* directory that is 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 if the `nosnapdir` or `nosnap` option is on.

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.

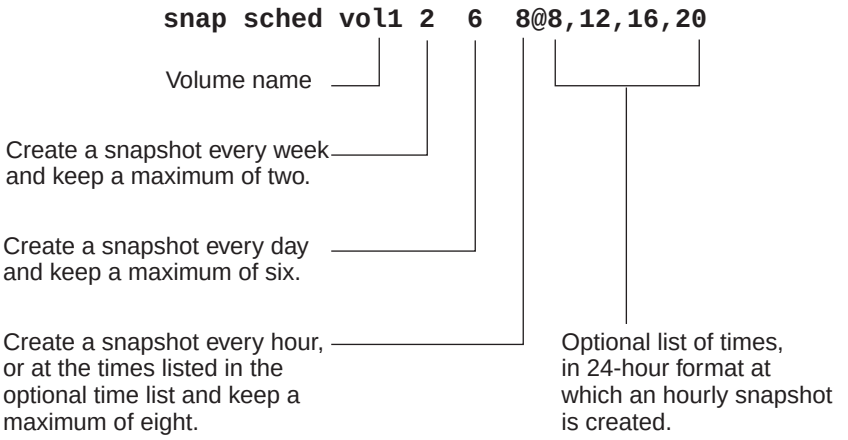
Types of automatic snapshots: The following table describes the three types of automatic snapshots.

Type	Description
Weekly	The filer creates these every Sunday 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 the filer creates a weekly snapshot, 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 the filer creates a nightly snapshot, the value of *n* is adjusted for all nightly snapshots. The higher the value of *n*, the older the snapshot.

Type	Description
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 the filer creates an hourly snapshot, the value of *n* is adjusted for all hourly snapshots. The higher the value of *n*, the older the snapshot.

Example 1 of snap sched command

The following example shows a sample `snap sched` command:



In the `snap sched` command, the first argument after the volume name in the example indicates how many weekly snapshots to keep (2), the second argument indicates how many nightly snapshots to keep (6), and the third argument indicates how many hourly snapshots to keep (8). 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 (8, 12, 17, 20). If the argument is omitted, the filer creates an hourly snapshot.

The default snapshot schedule is

`snap sched` *`volume_name`* `0 2 6@8,12,16,20`

Example 2 of snap sched command

Following is an example of the `snap sched` command:

`snap sched` *`volume_name`* `2 6 8@8,12,16,20`

Snapshots created by this schedule: The following list describes the snapshots created by the example:

- 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 8:00 p.m., and keeps the eight most recent

The following list shows the snapshots that are 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
```

Result: This schedule keeps the eight most recent hourly snapshots, created at 8 a.m., 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. 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 are not necessarily all 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.

The `snap sched` command is persistent across reboots. There is no need to put the command in the */etc/rc* file.

User-defined automatic snapshots

You can create snapshots at predefined times instead of using the hourly, daily, and weekly schedules.

Example: For example, if you want to create two snapshots for the volume named vol1 each week, you can set up a `cron` job on the administration host to run twice each week at an appropriate time to execute the following `snap create` command:

```
rsh toaster snap create vol1 filename
```

Note

The `snap create` command does not accept a snapshot name containing a slash (/).

Understanding snapshot disk consumption

About snapshot disk consumption

It is important to understand the amount of disk space snapshots consume and the amount of disk space they are likely to consume. The following sections explain 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 might be referenced by several 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.

Using the df command to display snapshot use

To provide information about snapshot disk utilization, the `df` command on the filer treats snapshots as a partition different from the active file system.

Sample df command output: Following is a partial sample `df` command output:

```
df

Filesystem            kbytes    used    avail   capacity
/vol/vol0             3000000  2000000  1000000  65%
/vol/vol0/.snapshot   1000000  500000   500000   50%
```

Note

The numbers in this example were rounded off to make the example easier to understand. Also, to make the output easier to read, the "Mounted on" column is not included in the sample `df` output in the following sections.

In this example, the *vol0* volume contains 4 GB of disk space. It has 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 `/vol/vol0/.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 is misleading if the filer charged the space to snapshots. In the example, half of the 1 GB reserved for the snapshot is used.

How the snapshot reserve works

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 276.

This section uses several examples to explain the advantages of reserving disk space for snapshots.

Snapshots use deleted active file disk space: 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.

The following example shows how disk space being freed by deleting files in the active file system ends up in the snapshot:

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` command output is as follows:

```
Filesystem              kbytes    used   avail   capacity
/vol/vol0/              3000000 3000000       0     100%
/vol/vol0/.snapshot     1000000  500000  500000      50%
```

If you remove 100 MB of files, the disk space used by these files is no longer part of the active file system, so the space is reassigned to the snapshots instead. If you enter the `df` command, the output is as follows:

```
Filesystem              kbytes    used   avail   capacity
/vol/vol0/              3000000 2900000  100000      97%
/vol/vol0/.snapshot     1000000  600000  400000      60%
```

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 space for new files.

Administering snapshot disk space: 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 the example if all files in the active file system were deleted. Before the deletion, the `df` output was as follows:

```
Filesystem              kbytes    used   avail   capacity
/vol/vol0/              3000000 3000000       0     100%
/vol/vol0/.snapshot     1000000  500000  500000      50%
```

After removing all the data in the file system, the `df` command generates the following output:

```
Filesystem              kbytes    used   avail   capacity
/vol/vol0/              3000000 2500000  500000      83%
/vol/vol0/.snapshot     1000000 3500000       0     350%
```

Explanation: 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 there.

Recovering disk space for file system use: 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.

Example: For example, if 500 MB of data were added to the active file system in the preceding example, a `df` command generates the following information:

```
Filesystem              kbytes    used   avail capacity
/vol/vol0               3000000 3000000      0     100%
/vol/vol0/.snapshot     1000000 3500000      0     350%
```

As soon as the filer creates a new snapshot, every block in the system is referenced by some snapshot. Therefore, no matter how many files you remove from the active file system, there is still not 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 274 for information about how to use the `snap list` command to determine which snapshot to delete.

Effects of snapshots on quotas

Quotas do not count disk 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.

Managing snapshot disk consumption

About snapshot management

This section describes

- how to schedule snapshots to suit your environment
- how to determine a reasonable snapshot reserve
- how to adjust the amount of space snapshots use

The examples in this section are for a volume named *home*.

Scheduling snapshots

The best way to manage the amount of space consumed by snapshots in each volume is to use the `snap sched` command to adjust the schedule of snapshot creation.

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. For example, this is the schedule that creates a snapshot every day and keeps two:

```
snap sched home 0 2 6@8,12,16,20
```

- 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. Following is the recommended snapshot schedule, which keeps six daily snapshots and two weekly snapshots:

```
snap sched home 2 6 8@8,12,16,20
```

- If the data changes very quickly, reduce the number of snapshots scheduled. For example, if a volume is filled and emptied each day, for example, a volume 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 volume, schedule snapshots every hour and keep them for just a few hours, or turn off snapshots. For example, the following schedule creates a snapshot every hour and keeps three:

```
snap sched home 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.

- When you create a new volume on a filer, the new volume inherits the snapshot schedule from the root volume. After you use the volume for a while, check how much disk space the snapshots consume in the volume.

Displaying snapshot statistics

The `snap list` command shows the amount of disk space used by snapshots in a specified volume. This command enables you to see how much disk space each snapshot uses, and helps you determine an appropriate snapshot reserve.

Command output

Following is an example of the command output. If you don't specify a volume name in the command, the output contains statistics about each volume.

```
snap list vol0

Volume vol0
%/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 : The %/used column shows space consumed by snapshots as a percentage of disk space being used in the volume. The first number is cumulative for all snapshots listed so far, and the second number is for the specified snapshot alone.

- The first number is equal to

$$\frac{\text{cumulative snapshot space} \times 100\%}{\text{cumulative snapshot space} + \text{file system space}}$$

- The second number is equal to

$$\frac{\text{this snapshot} \times 100\%}{\text{this snapshot} + \text{file system space}}$$

The %/total column : The %/total column shows space consumed as a percentage of total disk space in the volume.

- The first number is equal to

$$\frac{\text{cumulative snapshot space} \times 100\%}{\text{total disk space in this volume}}$$

- The second number is equal to

$$\frac{\text{this snapshot} \times 100\%}{\text{total disk space in this volume}}$$

  "Cumulative snapshot space" is the total space used by this snapshot and all other more recent snapshots (the ones preceding this snapshot in the `snap list` output).

Output summary: The %/used number is more useful for planning the snapshot reserve because it is more likely to remain constant as the file system fills.

The example shows a volume that keeps five nightly snapshots and four hourly snapshots. That is, the volume uses the following command for creating snapshots regularly:

```
snap sched vol0 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. Assuming that this pattern of change holds up, a reserve of 12% to 15% provides a safe margin to ensure that removing files frees disk space when the active file system is full.

The values in parentheses, that is, 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 disk space indicated, because other snapshots might be referring to the same blocks. Refer to "Adjusting disk space used by snapshots" on page 276 for further information about how to select a snapshot file for deletion to reclaim disk space.

If you do not want the total amount of disk 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 preceding example, 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 the following command:

`snap reserve volume_name percent`

For example:

```
snap reserve vol0 25
```

With no arguments, the `snap reserve` command displays the percentage of disk space reserved for snapshots in each volume.

Note

Snapshots can exceed the snapshot reserve space.

Adjusting disk space used by snapshots

This section describes how to use the `snap list` output to determine which snapshot file to delete to free the most disk space.

In the sample `snap list` output in "Displaying snapshot statistics" on page 274, 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.

Example: 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 output:

```
%/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 disk space used by snapshots does not increase because no changes were made to the file system. However, 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 output:

```
%/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
```

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* are different if you delete and create data between snapshots in the following way:

1. Delete 20 MB of data and create 20 MB of new data after *hourly.2*.
2. Delete the 20 MB of data created in Step 1 after *hourly.1*.

After the data deletions and additions, the `snap list` command displays the following output:

```
%/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. However, this time they reference different data blocks. Cumulatively, they consume 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 much disk 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. When you create a new volume on a filer, remember that the new volume inherits the snapshot schedule from the root volume. After you use the volume for a while, check how much disk space the snapshots consume in the volume. If the disk space 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 help you gain a better understanding of how snapshots work.

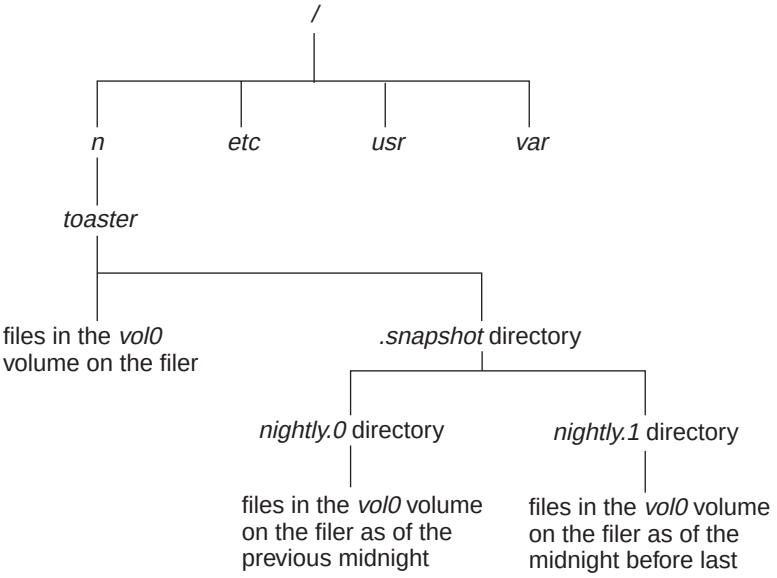
Accessing snapshots from clients

About client access to snapshots

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 users can access old versions of files in that directory. How users gain access to snapshots depends on the file-sharing protocol used: NFS or CIFS.

NFS client access to snapshots

The following illustration shows the directory structure on a client with the *vol0* volume of a filer named toaster mounted on */n/toaster*.



Explanation: In this example, the client can obtain access to snapshots by way of */n/toaster/.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* directories in all other directories are "magic"; that is, can be accessed when they are referenced by name but do 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     dir2
```

The same command entered in a directory below the mount point does not show the *.snapshot* directory; for example:

```
cd dir1
ls -a

.     ..    file1     file2
```

If you enter the `ls` command with the directory name *.snapshot*, you can see a listing of the snapshots for the *dir1* directory:

```
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 commands for removing files 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.

CIFS client access to snapshots

To CIFS clients, the snapshot directory appears only at the root of a share. For example, if a user's home directory is a share named bill that corresponds to the */vol/vol0/home/bill* directory, only the */vol/vol0/home/bill/.snapshot* directory is visible. When this user displays the contents of the home directory, the snapshot directory is displayed as *~snapshot* if the operating system supports long file names and as *~snapsht* if the operating system supports only short file names.

Note

The snapshot directory is visible in that it is displayed in a directory listing or File Manager display if the client operating system is configured 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

From an NFS client: the best way to find all versions of a particular file preserved in snapshots is to use the `ls` command. The following example shows how to find all versions of *foo*:

```
ls -l foo .snapshot/*/foo
-rw-r--r--  1 smith 0 Jan 14 09:40  foo
-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.

From a CIFS client: use the `Find` command to search for the file in the *~snapshot* directory. For example, if a user maps the home share to drive F: and wants to find all versions of *foo* in snapshots, the user can use the `Find` command to search for *foo* in the *f:\~snapshot* folder.

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.

From an NFS client: you can use the `ls -lu` command, which shows the access times of files, to determine when snapshots were created. Following is an example of the `ls -lu` command:

```
ls -lu foo .snapshot/*/foo
-rw-r--r--  1 smith 0 Jan 14 09:40  foo
-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
```

From a CIFS client: you can determine the access time of a file by checking its properties.

Administering Qtrees 11

About qtrees

A qtree is a special subdirectory of the root directory of a volume.

Qtree parameters: You can set the following parameters on a qtree:

- security style: NTFS (Windows NT file system), UNIX, or mixed
- oplocks setting: On or Off
- disk space and file limits, as described in Chapter 12, "Managing Quotas and Maximum Number of Files."

Note

Qtrees used to be called quota trees. Any quota trees created in previous releases of Data ONTAP are now qtrees.

Volumes and qtrees: A volume has all the properties of a qtree except

- It can have qtrees under it.
- It has different defaults than a qtree.

Unless expressly mentioned otherwise, whatever applies to qtrees also applies to volumes.

Note

You cannot create a qtree inside another qtree.

Uses of qtrees: You use qtrees to group files that have similar characteristics, much in the way that you use volumes. However, qtrees are much more flexible than volumes. For details, see "Using qtrees" on page 283.

Chapter contents

This chapter discusses how to administer qtrees from the filer command line. You can also administer qtrees from FilerView.

This chapter discusses the topics listed in the following table.

Read...	To learn about...
"Using qtrees" on page 283	What qtrees are and what you can do with them.
"Qtree security styles" on page 285	Security styles and how to set and change the security style of a qtree.

Read...	To learn about...
"Qtree file access models" on page 288	How to look at qtrees using file access models.
"Creating a qtree" on page 290	How to create qtrees.
"Modifying the security style of a qtree" on page 291	How to set and change the security style of a qtree.
"Modifying qtree oplocks settings" on page 292	How to set and change the oplocks setting of a qtree.
"Displaying qtree information" on page 294	How to display information about all qtrees on a filer so you can see how many there are and what their properties are. This enables you to make informed decisions about qtree management.

Using qtrees

What you can do with qtrees

You can use qtrees in the following two ways:

- Group files that have the same security style and oplocks setting, such as files related to a particular activity, for example, a project, without having to create a separate volume for them.
- Perform quick and easy backups.

Using a qtree for a project: One way to group files is to set up a qtree for a project, such as one maintaining a database. Setting up a qtree for a project enables you to do the following actions:

- Set the security style of the project without affecting the security style of other projects.

  For example, you use NTFS-style security if the members of the project use Windows files and applications. Another project in another qtree can use UNIX files and applications, while yet another project can use both Windows and UNIX files.
- Set oplocks (if the project uses Windows) as appropriate to the project without affecting other projects.

  For example, if one project uses a database that requires no oplocks, you can turn oplocks Off on that project's qtree. If another project uses oplocks, it can be in another qtree that has oplocks set to On.
- Limit the disk space and number of files available to a project so that it does not use up resources that other projects and users need. As the needs of the projects and available resources change, you can easily change the limits on the qtree. For instructions about managing space using qtrees, see Chapter 12, "Managing Quotas and Maximum Number of Files."

Using a qtree for backups: You can back up individual qtrees. You would do so to

- add flexibility to backup schedules
- modularize backups
- keep the size of each backup to one tape

For details, see Chapter 13, "Backing Up and Restoring Files."

Qtree and volume defaults

Volumes and qtrees have the default values shown in the following table.

Parameter	Qtree default	Volume default
`oplocks`	On	On
`security`	The style of the volume's root directory	UNIX

Note

Quota trees that were created in previous releases of Data ONTAP have, as defaults, UNIX security style and oplocks On.

Moving files between qtrees

In UNIX, you cannot move a file into or out of a qtree with a rename operation. As a result, the `mv` command on some UNIX systems fails if you try to move a file into or out of a qtree. You can always move the file by copying it, then deleting the original.

In Windows, you can move a file into or out of a qtree.

Qtree security styles

Types of security styles

There are three kinds of security styles, described briefly in the following table.

Style	Behavior
NTFS	Exactly like Windows NT NTFS: Files and directories have Windows NT file-level permission settings. **Note** To use NTFS security, make sure that the filer is licensed for CIFS.
UNIX	Exactly like UNIX: Files and directories have UNIX permissions.
mixed	Both NTFS and UNIX security are allowed: a file or directory can have either Windows NT permissions or UNIX permissions.

Qtree security styles in detail

The following table describes the security styles in detail and the effects of changing to each style.

Security style	Description	Effect of changing to the style
NTFS	User access depends on the protocol, as follows: **CIFS requests:** Windows NT permissions determine user access if Windows NT permissions have been set on a file. **NFS requests:** UNIX-style permission bits that the filer generates and stores determine access in conjunction with a mapping of the Windows owner to a UNIX identity, where possible. Security permissions for Other are based on the permissions for the Everyone group. Security permissions for Owner are usually based on the owner of the file, with the following exceptions: ◆ The owner is a member of the Administrators group. ◆ The owner is a generic user. In these cases, the user is mapped to root with restricted owner rights. **Note** You cannot change permissions in an NTFS qtree from a UNIX client.	Windows NT permissions determine file access for a file that had them if the change is from a mixed qtree. Otherwise, UNIX-style permission bits determine file access for files created before the change.

Security style	Description	Effect of changing to the style
UNIX	User access depends on the protocol, as follows: **CIFS requests:** Windows users are mapped to a UNIX UID and UNIX permissions determine access. In a UNIX qtree, a user cannot set Windows NT permissions. A Windows user can change UNIX permissions from Windows using SecureShare Access, as described in "Viewing and changing UNIX permissions from Windows" on page 203. **NFS requests:** Only the user's UID, GID, and UNIX-style permission bits of the file or directory determine user access.	The filer disregards any Windows NT permissions established previously.
mixed	Both NTFS and UNIX style permissions are permitted. The security style of a file is the style most recently used to set permissions on that file. **Caution** Changing NTFS permissions on a file recomputes UNIX permissions on that file. Changing UNIX permissions or ownership on a file deletes any NTFS permissions on that file.	None.

Qtree file access models

Kinds of file access models

You can use the following four file access models in working with qtrees:

- CIFS user accessing a file with Windows NT security
- CIFS user accessing a file with UNIX security
- NFS user accessing a file with Windows NT security
- NFS user accessing a file with UNIX security

CIFS access to Windows files

CIFS accesses to Windows files obey Windows security rules.

CIFS access to UNIX files

The following principles apply to accessing UNIX files from CIFS:

- All CIFS users are mapped to UNIX UIDs and GIDs.
- File accesses use UNIX security or PC security, as chosen during the `cifs setup` program.
  - UNIX-style permissions are determined by the rights associated with the UNIX UID and GID.
  - PC-style permissions are determined by the rights assigned in a share's Access Control List (ACL) and are limited by the UNIX permissions assigned to a file.
- PC security is like FAT (File Allocation Table) file system security with per-file permissions:
  - If the owner of a file or directory accesses an item, the owner permissions are checked to see whether they allow access.
  - If someone other than the owner of a file or directory accesses an item, the group permissions are checked to see whether they allow access.

NFS access to Windows files

The following principles apply to accessing Windows files from NFS:

- Windows NT permissions are mapped to UNIX permissions.
- Each Windows NT user who sets Windows NT permissions is mapped to a UNIX user and UNIX group, except that if the owner is a generic user, the owner is mapped to root with restrictions.

- Windows NT permissions for Owner are mapped to UNIX owner permissions.
- Windows NT permissions for Everyone are mapped to UNIX Group and UNIX Other permissions.

NFS access to UNIX files

NFS accesses to UNIX files obey UNIX security rules.

Creating a qtree

How to create a qtree

To create a qtree, use the following command:

```
qtree create pathname
```

Result: The qtree *pathname* is created, with the following properties:

- Volume: the root volume, unless you specify another volume
- Name: *pathname*
- Security style: that of the root directory of the volume
- Oplocks settings: On

Creating a qtree in the root volume

If *pathname* does not begin with a slash (/), the qtree is created in the root volume. For example:

```
qtree create news
```

creates the qtree */vol/vol0/news*, where */vol/vol0/* is the default name for the root volume. For information about volumes, see.“Volume concepts” on page 83

Creating a qtree in a volume other than the root volume

If you want to create a qtree (for example, *news*) in a particular volume (for example, *users*), use the following command:

```
qtree create /vol/users/news
```

Modifying the security style of a qtree

When to change the security style of a qtree

There are many circumstance under which you might want to change qtree style. Two examples are

- Because the default security style of a qtree is that of its root volume, you might want to change the security style of a qtree after creating it to the style you want.
- You might also want to change the security style to accommodate other users or files; for example, if you start with an NTFS qtree and subsequently want to include UNIX files and users, you might want to change the qtree to a mixed qtree.

How to change the security style of a qtree

To change the security style of a qtree, use the following command:

```
qtree security [pathname [mixed | ntfs | unix]]
```

Example with a qtree: To change the security model of */vol/users/docs* to be the same as Windows NT, use

```
qtree security /vol/users/docs ntfs
```

Example with a volume: To change the security model of the root directory of the *users* volume to mixed so that, outside of a qtree in the volume, one file can have NTFS security and another UNIX security, use

```
qtree security /vol/users/ mixed
```

Note

When you create an NTFS qtree or change a qtree to NTFS, by default, every Windows user is given full access. You must change the permissions if you want to restrict access to the qtree for some users. If you do not set NTFS file security on a file, UNIX permissions are enforced.

Modifying qtree oplocks settings

When to change oplocks settings

You might want to change qtree oplocks settings when you add or remove software, or when the kind of data you are using changes. For detailed information about oplocks, see “Using oplocks” on page 207.

Changing oplocks settings

To change the oplocks setting of a qtree, follow these steps:

1. Make sure that the `cifs.oplocks.enable` option is set the way you want.
2. Use the qtree oplocks command, as follows:

```
qtree oplocks [name [enable | disable]]
```

The command takes effect immediately.

Note

If you disable the oplocks feature on a qtree, existing oplocks in the qtree are not broken.

Example with a qtree: To enable oplocks in the */vol/users/docs* qtree if oplocks are disabled and the `cifs.oplocks.enable` option is set to On, enter the following command:

```
qtree oplocks /vol/users/docs enable
```

Example with a volume: To disable oplocks in the entire *users* volume if oplocks are enabled and the `cifs.oplocks.enable` option is set to On, enter the following command:

```
qtree oplocks /vol/users/ disable
```

This disables only files and directories that were not in a qtree when oplocks were enabled.

Effect of the cifs.oplocks.enable option

Setting the `cifs.oplocks.enable` option has the following effects:

- If the `cifs.oplocks.enable` option is set to Off, all oplocks on the filer are turned off.

- If the `cifs.oplocks.enable` option is set back to On, the setting for each qtree comes into effect and oplocks are turned on for those qtrees where oplocks are enabled.

Displaying qtree information

How to display qtree information

To display all attributes of all qtrees on a filer, including quota trees created in previous versions of Data ONTAP by the `quota qtree` command, use the `qtree` command with no arguments.

The qtree command display

The `qtree` command lists for a filer the items described in the following table.

Field	Contents
`Volume`	The volumes on a filer. Bear in mind that a volume is itself a qtree.
`Qtree`	Qtrees that are not volumes; each is listed next to its volume.
`Style`	The security style of each qtree.
`Oplocks`	The oplocks setting of each qtree.

Example qtree display: For example:

```
qtree

Volume   Qtree    Style Oplocks
-------- -------- ----- --------
bagels            unix  enabled
bagels   sesame   unix  enabled
muffin            ntfs  enabled
muffin   bran     unix  enabled
```

Explanation of example qtree display: In the example

- Because *bagels* and *muffin* are volumes, each has a security style and oplocks setting.
- *Sesame* is a qtree in the *Bagels* volume. Its security style and oplocks setting happen to be the same as that of its parent volume, bagels.
- *Bran* is a qtree in the *muffin* volume. Its security style is different from that of its parent volume, *muffin*. Files in *bran* have UNIX-style permissions; files in *muffin* but not in *bran* have NTFS-style permissions.

Managing Quotas and Maximum Number of Files 12

Chapter contents

This chapter describes how to manage disk space on the filer. The following table lists the types of information available.

Read...	To learn about...
“Restricting disk usage by using disk quotas” on page 296	Restricting disk usage by using quotas.
“Increasing the maximum number of files” on page 307	Increasing the maximum number of files in a volume.
“The df command” on page 308	Using the `df` command to monitor the current disk space usage.

Restricting disk usage by using disk quotas

About 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 tree.

- You create and manage disk quotas by editing the */etc/quotas* file and using the `quota` command.
- You create and manage qtrees with the `qtree` command.

Format of the quotas file

To set up disk quotas, create a *quotas* file in the */etc* directory.

Note

Keep a record of your *quotas* file in a safe place and update it as you change it, in case you must do a restore without having access to the root volume.

Following is a sample *quotas* file:

```
#Quota Target       type                disk     files
/vol/home/user/joe  user                500M      10K
21                  group               750M      75K
/vol/eng/proj1      tree                750M      75K
writers             group@/vol/eng/proj1 300M     50K
*                   user                50M       10K
```

Note

To specify a qtree in the */etc/quotas* file, you must use tree and not qtree. The displays of the quota command use `tree` for qtree.

Keep a record of your *quotas* file in a safe place and update it as you change it, in case you must do a restore without having access to the root volume.

Quota Target field: Specifies the user, group, or qtree on which you want to impose restrictions. You can assign more than one quota to a user or group, but only one quota to a qtree. The entries can be in any order.

Quota target for a user quota: You specify a user with one of the following targets:

- a file or subdirectory whose UID matches the user

- the user's 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 quota implications for that file or directory. The UID of the user must not be 0.

Any file or subdirectory you use in the Quota Target field is 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 account remains on the system. For example, use a user's home directory for a user quota.

Quota target for a group quota: You specify a group with one of the following targets:

- 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 quota implications for that file or directory. The GID of the group must not be 0.

Quota target for a tree quota: To create a tree quota, use the `quota qtree` command to create a directory in the root directory of a volume. The quota target in the quotas file for a tree quota is the complete path name of this directory.

Quota target for default quotas: Use an asterisk (*) in the Quota Target field to specify a default for the user or group quotas. Defaults do not apply to tree quotas. The default value applies to the following users or groups:

- New users or groups that are created after the default entry has taken effect. For example, if the maximum disk space for default user quotas is 500 MB, any new user can use up to 500 MB of disk space.
- Users or groups that are not explicitly mentioned in the *quotas* file. For example, if the maximum disk space for default user quotas is 500 MB, users for whom you have not specified a user quota in the *quotas* file can use up to 500 MB of disk space.

To override a default for a specific user or group, specify a quota for that user or group.

Type field: You can enter one of the following values in the Type field to define the quota type:

- **user:** If a user quota applies just to a tree and not to the entire volume, specify `user@`*`tree`*, where *tree* is the name of a qtree. If a user quota applies to a volume other than the root volume, append `@/vol/`*`volume`* to the quota type. For example, `user@/vol/`*`marketing`* means that the user quota applies to the *marketing* volume.
- **group:** If a group quota applies just to a tree and not to the entire volume, specify `group@`*`tree`* where *tree* is the name of a qtree. If a group quota applies to a volume other than the root volume, append `@`*`volume`* to the quota type. For example, *`group@/vol/marketing`* means that the group quota applies to the *marketing* volume.
- **tree:** A tree quota is similar to a disk partition, but you can increase or decrease the size of a tree quota at any time.

Disk field: Specifies the maximum amount of disk space that the quota target can use. In this field, K is equivalent to 1,024 bytes, M means 2^20 bytes, and G means 2^30 bytes. If you omit the K, M, or G, the default is K.

Note

Do not put a blank in the Disk field; it acts as white space. The filer regards the following entries as equivalent:

```
#Quota Target      type      disk      files
/export            tree                75K
/export            tree      75K
```

Files field: Specifies the maximum number of files that the quota target can use. Use K to indicate 1,024 files. For example, 75K means 76,800 files. Use M to mean 2^20 and G to mean 2^30. You can omit the K, M, or G. For example, if you enter 100, it means the maximum number of files is 100. A blank in this field means there is no restriction on the number of files that the quota target can use.

Sample quotas file

Following is a sample quotas file that includes different kinds of quotas:

```
#Quota Target         type                  disk      files
/vol/home/user/jdoe   user                  500M      10K
108                   user                  500M      10K
jsmith                user@/vol/rls         500M      10K
publications          group                 750M
/vol/home/eng         group@/vol/cad        750M      75K
/vol/cad/proj1        tree                  750M      75K
writers               group@/vol/cad/proj1  150M
*                     user                   50M      15K
*                     user@/vol/cad/proj1    50M      10K
```

```
*                   group                 750M      85K
*                   group@/vol/cad/proj1  100M      75K
mhoward             user                  150M     100K
mhoward             user@/vol/cad/proj1    75M      75K
```

Any operation that creates files or writes to them must satisfy all applicable quotas. The following list describes the effects of the sample quotas file:

- The owner of */vol/home/user/jdoe* and the user whose UID is 108 can each use 500 MB of disk space and 10,240 files in the root volume.
- The user whose user name is jsmith can use 500 MB of disk space and 10,240 files in the *rls* volume.
- The group publications can use 750 MB of disk space with no restrictions on the number of files in the root volume.
- The group that owns */vol/home/eng* can use 750 MB of disk space and 76,800 files in the *cad* volume.
- The qtree proj1 in the *cad* volume can use 750 MB of disk space and 76,800 files.
- The writers group can use 150 MB of disk space and an unlimited number of files in the proj1 qtree provided that the quotas on the proj1 qtree are not exceeded.
- Any user not otherwise mentioned in this file can use 50 MB of disk space and 15,360 files in the root volume.
- In the proj1 qtree, any user not otherwise mentioned in this file can use 50 MB of disk space and 10,240 files.
- Any group not otherwise mentioned in this file can use 750 MB of disk space and 87,040 files in the root volume.
- In the proj1 qtree, any group not otherwise mentioned in this file can use 100 MB of disk space and 76,800 files.
- The user mhoward can use 150 MB of disk space and 102,400 files in the root volume. In the proj1 qtree, mhoward can use 75 MB of disk space and 76,800 files.

Both user and group quotas apply to the entire specified volume (or the root volume if no volume name is specified). This is true even if the quota target in the *quotas* file is specified in the form of a path name. For example, if the quota target for the user named jdoe is */vol/home/user/jdoe*, the filer imposes the quotas on all files written by jdoe, not just the ones written to */vol/home/user/jdoe*.

The quota command

The quotas file specifies what the restrictions are on users, groups, and trees. Whether these restrictions take effect depends on the `quota` command. The `quota` command enables you to do the following tasks:

- enable and disable quotas on a per-volume basis
- resize quotas on a per-volume basis
- display information about all active quotas or about quotas that apply to a specified path
- create a qtree

Some of the information from `quota` commands is available through SNMP using the Network Appliance custom MIB. For more information about the MIB, refer to "About the Network Appliance custom MIB" on page 98.

Enabling or disabling quotas: The `quota on|off` command enables or disables quotas for all volumes or a specific volume. Use the following syntax when using the command:

```
quotas [on|off] volume
```

Use the following command to enable quotas for a volume:

```
quota on volume
```

This command computes the disk usage of each quota target. The computation can take a few minutes to complete for a large number of quotas. To find out how much quota initialization the filer has completed, use the `quota` command without any arguments. For example:

```
quota

vol0: quotas are on.
vol1: quotas are initializing (24% done).
vol3: quotas are off.
```

Use the following command to disable quotas for a volume:

```
quota off volume
```

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*.

Resizing quotas: 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 output of the `quota report` command, discussed in "Displaying information about quotas" on page 303.

You use the `quota resize` command after changing limits for a group or user. For example, jdoe has a disk quota of 500 MB, as shown in the following example.

```
#Quota Target         type         disk      files
/vol/home/user/jdoe   user         500M       10K
*                     user          10M       10K
```

Use the `quota resize` command to update the quota if jdoe's disk quota were increased, as shown in the following example:

```
#Quota Target         type         disk      files
/vol/home/user/jdoe   user         600M       10K
*                     user          10M       10K
```

How quota resize affects newly added quota targets: The `quota resize` command usually ignores newly added quota targets. For each entry that it skips, the `quota resize` command prints the following message:

`quota resize: new entry on line` *n* `in /etc/quotas`

where *n* is a line number.

The `quota resize` command does not ignore newly added quota targets in the following situation:

If a default quota applies to the creator of a file, an active quota record is created for the owner of the file. If a newly added quota target is the user or group that has an active quota record, the `quota resize` command does not ignore the newly added quota target. That is, if a user or group has written to a file that is under the control of a default quota, a newly created entry in the quotas file for this user or group takes effect after a `quota resize` command.

After you edit the quotas file, if you want to make sure that all entries take effect, enter the `quota off` command followed by the `quota on` command. In this way, all quotas listed in the file become active.

Creating an active quota for a quota target: You can use the `quota resize` command so that quotas can take effect on targets that have not created any files.

Because a quota becomes active when a quota target has written a file, you can make an entry in the quotas file an active quota by following these steps:

1. Create a file.
2. Change ownership of the file to the quota target.

Because the quota target now has an active quota record, you can enter the `quota resize` command, including the volume name, to make the quota entry go into effect. This procedure takes less time than executing the `quota off` command followed by the `quota on` command because only the newly active quota is recalculated.

Displaying information about quotas: The `quota report` command displays the current consumption of files and space for each quota target and for each user and group that is under quota restrictions.

For example, if a user quota is specified in the quotas file for a user named jdoe, the quota report shows how many files and how much disk space have been used by jdoe. The Quota Specifier column in the quota report shows the same information as the Quota Target column in the quotas file, with the following exception:

If a user or group is under default quota restrictions, `quota report` displays information about the user or group as if the user or group had an entry in the quotas file. In this case, the Quota Specifier column in the quota report is blank.

With a path argument, `quota report` displays information about all quotas that apply to the specified file.

Creating a user quota

Follow these steps to limit disk space used by a user named jdoe.

Step	Action
1	Decide who should be limited by a disk quota. In this example, the user is the owner of */vol/home/user/jdoe*. The filer restricts disk usage of files with the same UID as */vol/home/user/jdoe*.
2	Add the following line to the quotas file: **`/vol/home/user/jdoe user 300M 20K`** Substitute 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 on the filer or the NIS database. If you want the restrictions to apply to a volume other than the root volume, for example, the home volume, enter the following line: **`/vol/home/user/jdoe user@/vol/home 300M 20K`** Substitute the values you want for 300M and 20K.

Step	Action
3	Use the `quota on` command, including the volume name. For example, if quotas are already on for the home volume, enter the following commands: **`quota off home`** **`quota on home`** Alternatively, if jdoe is already under quota restriction, for example, if his files were restricted by a default user quota, enter the following command so that the user quota you just created can take effect: **`quota resize home`**

Creating a group quota

Follow these steps to limit disk space used by a group named service.

Step	Action
1	Decide which group should be limited by a disk quota. In this example, the group is the owner of */vol/home/user/service*. That is, the filer restricts disk usage of files with the same GID as */vol/home/group/service*.
2	Add the following line to the quotas file: **`/vol/home/group/service group 700M 100K`** Substitute the values you want for 700M and 100K. You can also use the GID of the group or the name of the group. If you want the restrictions to apply to a volume other than the root volume, for example, the home volume, enter the following line: **`/vol/home/group/service group@/vol/home 700M 100K`** Substitute the values you want for 700M and 100K.

Step	Action
3	Use the `quota on` command. If quotas are already on for the *home* volume, enter the following commands: **`quota off home`** **`quota on home`** Alternatively, if the service group is already under quota restriction, for example, if the group's files are already restricted by a default group quota, enter the following command so that the group quota you just created can take effect: **`quota resize home`**

Removing quota restrictions

To remove quota restrictions, follow these steps.

Step	Action
1	Remove the appropriate line or lines from the *quotas* file.
2	Use the `quota off` and `quota on` commands, including the volume name. For example, enter the following commands: **`quota off home`** **`quota on home`**

When quotas are exceeded

This section describes how the filer responds when quotas are exceeded and what users see on the client systems.

Messages displayed by the filer when quotas are exceeded: When it receives a write request, the filer first determines whether the file to be written is in a qtree. If the write would exceed the qtree quota, the filer logs the following error message:

`tid` *`tree_id`*`: tree quota exceeded on` *`volume_name`*

If the qtree is not full but the write would cause either the user or group quota to be exceeded, the filer 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 and *group_id* is the group's GID.

To the client, the filer returns an "out of disk space" error to the NFS write request or a "disk full" error to the CIFS write request. The following sections describe how the clients notify the users about quotas being exceeded.

Messages displayed on NFS clients: If a write from an NFS client to a filer causes a quota to be exceeded, the user experience depends on the operating system version and the application.

If a UNIX client mounts a filer without the `noquota` option, the `login` program on the client checks to see whether the user has reached the disk quota and file quota each time the user tries to log in to the client. The client displays a message to alert the user before displaying the system prompt if a quota was reached. In the following example, a user reached the disk quota on the filer mounted as */t/toaster* on a client named client2:

```
rlogin client2
You have mail.
Block limit reached on /t/toaster
client2%
```

Not all versions of UNIX perform the quota check as described in this section. Also, the exact message printed varies from one version to another.

If a write causes a quota to be exceeded, the error message seen by the user depends on the application. For example, on a SunOS 4.x client, if a user tries to save a file using `vi` when his or her disk quota is reached, the error message is

```
Disc quota exceeded [Warning - /t/toaster/home/jdoe/file1 is
incomplete]
```

Messages displayed on CIFS clients: If a write from a CIFS client to a filer causes a quota to be exceeded, the user experience depends on the operating system and the application. Following are two examples:

- An application might display a message as follows:

```
Cannot write file filename
```

- When a user tries to copy a file to the filer using the Explorer in Windows 95, the error is as follows:

```
Cannot create or replace filename: Cannot read from the source file
or disk.
```

Increasing the maximum number of files

About 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. The increase is determined by the filer, and is not a user-specified value.

Unlike UNIX, which requires that you specify the maximum number of files in a file system when you create the file system, the filer enables you to use the `maxfiles` command to increase the number of files for each volume at any time.

Note

Use caution when increasing the maximum number of files because after 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.

Viewing the number of files in a volume

To see how many files are in a volume, use the `df -i` *`pathname`* command, which shows how many inodes have been used, or use the `maxfiles` *`volume`* command.

For example, both of the following commands show that the home volume has used 2,872 inodes:

```
df -i /vol/home

Filesystem     iused     ifree    %iused   Mounted on
/vol/home/      2872      118090      2%    /vol/home

maxfiles home

Volume home: maximum number of files is currently 120962 (2872
used)
```

Information generated by the `maxfiles` command is available through SNMP using the Network Appliance custom MIB. For more information about the MIB, see “About the Network Appliance custom MIB” on page 98.

The df command

About the df command

To verify the amount of free disk space on the filer, enter 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 “About the Network Appliance custom MIB” on page 98.

With the `-i` option, the command displays the number of used inodes and the number of available inodes. Following is an example of the `df -i` command:

```
df -i /vol/home

Filesystem    iused      ifree    %iused   Mounted on
/vol/home/    240843     121525     66%     /vol/home
```

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 volume. Consider the following sample `df` command issued on a volume with seven 2-GB disks installed:

```
df /vol/vol0

Filesystem          kbytes    used       avail   capacity
/vol/vol0/          9886448   8426196    1460252 85%
/vol/vol0/.snapshot 1098492   427008     671484  39%
```

When you add the numbers in the kbytes column, notice that the sum, 10,984,940 KB (about 11 GB), is significantly less than the total disk space installed (14 GB). The following reasons account 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 is restricted only by the amount of disk space installed in the volume. For individual files, the maximum size is 8 TB. Many NFS clients restrict the file size to 2 GB or 4 GB.

Using the df command with qtrees

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, if the filer is mounted on the client as */t/toaster*, the `df` command on the client displays the disk information about the */t/toaster* file system as follows:

```
df /t/toaster/engineering/jdoe

Filesystem            kbytes    used   avail capacity  Mounted on
toaster:/            2097151 1646923  450228    79%    /t/toaster
```

However, if you defined qtrees on the filer, the information about available space could be misleading because the actual space available might be less. For example, if */engineering* is a qtree with a disk quota of 1,800 MB, the space available in the */t/toaster/engineering* directory is less than that in the `df` command output shown in the preceding example.

If you have qtrees on the filer, Network Appliance recommends that you mount each qtree separately. For example, if the filer named toaster has two qtrees, */vol/home/engineering* and */vol/home/marketing*, mount *toaster:/vol/home/engineering* and *toaster:/vol/home/marketing* on two mount points, for example, */t/toaster/engineering* and */t/toaster/marketing*.

In this way, the filer takes the qtrees into consideration when responding to a `df` command from a client and returns the amount of free space in each qtree, as opposed to the space available in the entire file system.

Backing Up and Restoring Files 13

Chapter contents

This chapter describes how to back up files from and restore files to a filer.

The chapter is divided into three sections:

- Dump and restore limitations
- Backup topics
- Restore topics

Dump and restore limitations

Read "Dump and restore limitations" on page 314 to learn how permissions are affected when files are dumped and restored between qtrees and volumes that use NTFS and UNIX-style permissions.

Backup topics

The following table describes the backup topics.

Read...	To learn about...
"Methods for backing up a filer" on page 315	The three methods you can use to back up the filer.
"Introduction to filer backup" on page 317	The filer `dump` command and the benefits of using it instead of other backup methods.
"The dump command syntax and options" on page 319	The syntax and parameters of the `dump` command, and information that is important to creating successful backups.
"Examples of the dump command" on page 322	Eight examples of how to use the `dump` command.
"Backing up from the filer" on page 326	Step-by-step instructions for using the `dump` command to back up the filer.
"Determining the number of tape volumes required" on page 328	Step-by-step instructions for determining how many tapes a backup requires.

Read...	To learn about...
"How the dump command uses snapshots" on page 330	How the `dump` command uses snapshots to enable backups to occur without taking the filer off-line.
"Backing up from a remote host" on page 332	Information you need to know to determine whether you want to use the UNIX `tar` or `cpio` commands to back up the filer data.
"Backing up the filer using NDMP" on page 333	Using the `ndmpd` command to back up the filer, including the command syntax and options and examples of the command.

Restore topics

The following table describes the restore topics.

Read...	To learn about...
"Methods for restoring files" on page 337	How to decide between using the `restore` command on the filer and using the UNIX `ufsrestore` command on a client.
"The restore command syntax and options" on page 339	The syntax and options for the `restore` command.
"Examples of the restore command" on page 342	Several examples of how to use the `restore` command.
"Restoring from the filer" on page 348	How to restore files by entering the `restore` command at the filer console or through a `telnet` session to the filer.
"Restoring from a remote host" on page 349	How to restore files by running the `restore` command through the `rsh` command.

Read...	To learn about...
“Using a filer tape drive from another system” on page 350	How to use a filer tape drive to dump and restore files on a UNIX system or on another filer.
“Controlling tape devices” on page 352	How to control tape devices with the `mt` command.

Dump and restore limitations

How qtree and volume permissions are affected

The following table shows how permissions are affected when files are dumped and restored between NTFS and UNIX-style qtrees and volumes.

Dump a file that has only...	Restore to...	Results...
Windows NT permissions	UNIX qtree or volume	File gets the default UNIX permissions for the qtree or volume to which it is restored.
UNIX permissions	NTFS qtree or volume	The file gets the default Windows permissions for the qtree or volume to which it is restored.

Other dumps and restores preserve permissions.

Methods for backing up a filer

Supported backup methods

The filer supports three backup methods:

- Backing up from the filer using the filer `dump` command
- Backing up from a remote host
- Backing up from a remote host using an application that supports the Network Data Management Protocol (NDMP)

Backing up from the filer

Backing up from the filer using the `dump` command is the preferred method. With this method, you can

- initiate the backup session from the filer console, from a `telnet` session, or from a trusted host using the `rsh` command
- use a tape drive attached to the filer as the device to which the backup is written
- use a tape drive attached to another host as the device to which the backup is written

Backing up from a remote host

Backing up from a remote host enables you to

- back up from an NFS client. Backing up from an NFS client enables you to
  - use the `tar` or `cpio` commands, or third-party backup software that is not compliant with NDMP, to back up filer volumes that are mounted to the remote host
  - use the tape drive attached to the remote host as the device to which the backup is written
  - use a tape drive attached to the filer as the device to which the backup is written
  - preserve NFS-specific file information, such as symbolic links and file names not understood by CIFS computers
- back up from a CIFS client. Backing up from a CIFS client enables you to
  - use the backup software on a CIFS host to perform a backup
  - preserve CIFS-specific file information, such as DOS-style 8.3 names and DOS attributes

Caution

You should use these backup methods only if your filer is licensed for a single protocol: NFS or CIFS. If you use `tar` or `cpio`, CIFS attributes are not preserved; if you use a backup program from a CIFS client, UNIX attributes are not preserved.

Backing up using NDMP

Backing up using NDMP enables you to use third-party backup management software to perform the backup operation either to local or remote attached tape drives. NDMP uses `dump` and backs up both NFS and CIFS attributes.

Introduction to filer backup

Use the dump command

You use the filer `dump` command to perform a backup from the filer.

Benefits of using the dump command

When you use the `dump` command, you gain the following benefits:

- You can back up active file systems; you do not need to take the filer off-line.
- You can back up existing snapshots.
- You ensure that all the NFS and CIFS attributes for files are preserved in the backup.
- You can direct the dump to a local tape device attached to the filer, to the standard output, or to a remote tape device on a host supporting the remote magnetic tape (`rmt`) protocol.
- You can back up a large subtree to multiple files on a single tape, or to multiple files on multiple tapes.
- You can back up a large subtree to a remote host even if the subtree size exceeds the maximum size allowed for a single tape file on the remote host.

  Each file on the tape is called a *tape volume*. They differ from disk volumes, which are file systems on the filer.
- You can use the tape created by the filer tape drive on clients that support the Solaris `ufsrestore` command.

Dump and restore for large volumes

You can segment individual dumps so that they can be completed sooner by setting up qtrees within each volume. Qtrees enable you to ensure that the file system does not grow beyond a predetermined size.

Note

Full qtree and volume backups are optimized in the `dump` command to make the first two `dump` passes faster than backing up subdirectories.

For information about qtrees, see “Restricting disk usage by using disk quotas” on page 296 and Chapter 11, “Administering Qtrees.”

Remote backup using shell scripts

Because the filer supports remote shell (`rsh`), you can automate system backups through shell scripts. Backing up the filer using `rsh` offers another advantage over starting `dump` on the console: you can be sure that `dump` is not aborted by a Ctrl-C inadvertently entered from a host connected to the filer through `telnet`.

Blocking factors for dump and ufsdump differ

The filer `dump` command is similar to the Solaris `ufsdump` command. A major difference between the commands is that the blocking factor in the filer `dump` command is in kilobytes, while in the Solaris `ufsdump` command, the blocking factor is in 512-byte units.

Verify dumps to ensure success

You should verify that dumps are completed successfully to ensure that you do not lose data because of tape problems.

Example: If you execute `dump` through the `rsh` command and `dump` needs more tapes than are available in your tape device, it displays 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.
```

Caution

If you do not verify that the dump succeeded and keep putting in new tapes and executing additional dumps, you will have tapes containing no dumps, and you might lose data.

Adding a tape drive to a filer

If you need to add an external tape drive to your filer, you can purchase one from Network Appliance or another vendor. Refer to the hardware guide for your filer model for instructions about attaching an external tape drive and a list of supported tape drives; using an unsupported tape drive might cause the filer to hang.

The dump command syntax and options

About this section

This section describes the syntax and options for the `dump` command, and includes other information important to creating successful backups.

The dump command syntax

The syntax for the `dump` command is as follows:

```
dump [ options [ arguments ]] [ subtree ]
```

Rules for using the dump command

Follow these rules when you enter the `dump` command:

- Specify multiple options without intervening spaces.
- Enter the arguments for each option in the order that you specify the options.
- Specify the volume of a subtree if the subtree does not exist on the default volume. If you do not include the volume name in the subtree, the `dump` command assumes that the subtree is within the filer root volume.

The dump command options

The following table describes the options for the `dump` command.

Option	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, Solaris. 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 Solaris `ufsdump` command.

Option	Argument	Meaning
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. Following are legitimate file names: • local tape drive names, for example, `nrst0a`. For more information about 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, specified as a dash (–).
b	*factor*	Specifies the blocking factor, in kilobytes. If you dump to local tape drives on the filer, 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.
B	*blocks*	Specifies the maximum size of each dump volume in 1,024-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 to load a new tape in the last device. 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 tape drives, consult your tape drive documentation. Use this option only if you plan to restore the dump volume on a host that has a size limit for a dump volume.

Option	Argument	Meaning
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.
Q	*vol_name*	Specifies a dump containing all data in a volume that does *not* reside in a qtree.
n	*name*	Specifies the name of the dump volume for */etc/dumpdates*. **Note** This is mandatory when dumping multiple subtrees.
A		Specifies a dump containing all data but no ACLs.

Use the norewind tape device when dumping multiple subtrees

When you dump multiple subtrees on one tape, you must use the norewind tape device in the dump command.

Caution

If you do not use the norewind tape device in the dump command when you dump multiple subtrees on one tape, you might overwrite data.

UIDs and GIDs longer than 16 bits might not be restored correctly

The filer uses the Solaris 2.5.1 scheme for putting files and directories on the dump tape when the files have UIDs or GIDs longer than 16 bits. If these files and directories are restored on systems that use Solaris earlier than version 2.5.1, they might not be restored with the correct UIDs or GIDs.

Examples of the dump command

About this section

This section provides several examples of how to use the `dump` command.The following table describes the examples.

For this task...	See...
Dumping one directory to the local tape device.	“Example 1” on page 322
Dumping a qtree to a single tape volume on the local tape device.	“Example 2” on page 323
Dumping one directory to a single tape volume on a remote tape device.	“Example 3” on page 323
Dumping a single directory to multiple tape volumes on a tape stacker or library.	“Example 4” on page 323
Dumping a single directory from an existing snapshot to a single tape volume on the local tape device.	“Example 5” on page 323
Dumping the entire root volume to a tape stacker using four tapes.	“Example 6” on page 324
Dumping multiple subtrees to a single tape volume on the local tape device.	“Example 7” on page 324
Dumping non-qtree data to a single tape volume on the local tape device.	“Example 8” on page 325

Example 1

The following example dumps one directory, */vol/vol0/home/user*, to the local tape device, *rst0a*:

```
dump 0uf rst0a /vol/vol0/home/user
```

The following table describes the options and arguments in this example.

Options and arguments	Description
`0uf`	The options for the dump. `0` = Do a level 0 dump. `u` = Update the *dumpdates* file. `f` = Output file or device.
`rst0a`	Argument for the `f` option—the output file or device. In this example, the device is the local rewind-on-close tape drive.
`/vol/vol0/home`	The directory to be dumped.

Example 2

The following example dumps a qtree to a single tape volume on a local tape device:

```
dump 0uf rst0a /vol/vol0/home
```

Example 3

The following example dumps one directory to a single tape volume on a remote tape device:

```
dump 0uf adminhost:/dev/rst0 /vol/vol0/home/user
```

Example 4

The following example dumps a single directory to multiple tape volumes on a tape stacker without operator intervention:

```
dump 0uf urst0a,urst0a,urst0a,urst0a,rst0a /vol/vol0/home/user
```

Example 5

The following example dumps a single directory from an existing snapshot to a single tape volume on a local tape device:

```
dump 0uf /home rst0a /vol/vol0/.snapshot/weekly.0/home/user
```

Example 6

The following example dumps the entire root volume in sequential mode to a tape stacker when four tapes are required:

```
dump 0uf urst0a,urst0a,urst0a,nrst0a /vol/vol0/
```

Example 7

You can dump multiple subtrees with a single `dump` command by using

- The `n` option to specify the name of the dump volume for the */etc/dumpdates* file.
- The `l` option to specify that `dump` reads path names from standard input.
- A directory name as the last command argument. This directory contains the subtrees to be dumped.

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 as the last argument of the `dump` command. Do not specify ".." as a subtree or specify subtrees that contain symbolic links. To end the list of subtree names, use a blank line as shown in the following example, or press Ctrl-D.

```
dump 0ufnl nrst0a user.1.3.5 /vol/vol0/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 /vol/vol0/home to nrst0a
DUMP: mapping (Pass I) [regular files]
DUMP: Reading file names from standard input
user1
user3
user5

DUMP: 96% done, finished in 0:07
DUMP: 3252426 tape blocks
DUMP IS DONE
DUMP: deleting "snapshot_for_backup.0" snapshot.
deleting.....................................................
```

Caution

When you do a level 0 dump for multiple subtrees, 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 cannot do incremental restores.

Example 8

The following example dumps non-qtree data to a single tape volume on a local tape device:

```
dump 0ufQ rst0a /vol/vol0/
```

In this example, entries of the following format are written to the */etc/dumpdates* file:

```
/vol/home/all_non_quota_tree_data
```

Backing up from the filer

What it means

Use the procedure in this section to back up from the filer. Backing up from the filer means

- entering the `dump` command
  - from the filer console
  - from a `telnet` session into the filer console
  - using the `rsh` command from a remote host
- using a dump device, such as a tape drive, attached
  - to the filer
  - to a remote filer
  - to a remote host

Preparing for a backup

Before you use the `dump` command, follow these guidelines to ensure success:

- Turn off the `nosnap` and `nosnapdir` options for each volume.
- If you dump to a remote host, check the maximum size of the dump file supported by the host.

  Reason: Some UNIX systems, for example, systems running SunOS 4.1.1 or earlier) don't support dump files larger than 2 GB. You can specify the dump file size in blocks to avoid exceeding the limit imposed by the remote host.
- The `dump` command creates tapes compatible with the Solaris 2.5.1 format; you can restore the tapes from Solaris computers.
- If you plan to restore tapes created on the filer tape drive from a tape drive on a remote host
  - Determine whether the host supports dump files larger than 2 GB.

    The filer can create dump files larger than 2 GB. If the host you plan to use to restore cannot support dump files larger than 2 GB, you might not be able to restore the dump files from that host.
  - Verify the blocking factor.

    Some systems don't support blocking factors larger than 63 KB, the default `dump` block size. 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.

Procedure for packing up a filer

Complete the steps in the following table to back up a filer.

Step	Action	
1	**If backing up to the filer tape drive...**	**If backing up to a remote host's tape drive...**
	Determine the capacity of the tape drive attached to the filer. Enter `sysconfig -t` Remember to consult your tape drive documentation for information about compression.	Check the maximum capacity of the tape drive on the remote host.
2	Create a quota report to list the usage in each of the qtrees to be backed up. Enter `quota report` Refer to "Use quota report to determine amount of data" on page 328 for more information. **Note** If any directories to be backed up are not in the quota report, because they are not qtrees or they are unlisted qtrees, find out the size of each directory by checking its properties from a CIFS client, or use the `du -s` command from an NFS client.	
3	Determine the number of tape volumes needed to back up the filer subtrees. Refer to "Determining the number of tape volumes required" on page 328 for instructions.	
4	Start the `dump` command. Enter `dump [ options [ arguments ]] [ subtree ]` **Note** If you dump multiple subtrees on one tape, you must use the `norewind` tape device in the `dump` command.	
5	Verify the dump.	

Determining the number of tape volumes required

Commands to use

You use the `sysconfig` and `quota` commands to gather the information you need to determine the number of tape volumes required to back up the filer. You can use the `du` command on a UNIX host in place of the `quota` command to determine the amount of data to be backed up.

Use sysconfig to determine dump tape size

The filer `sysconfig` command enables you to determine the tape drives attached to the filer, their formats, and their capacities.

Example: The following example shows how to verify the dump tape size for a filer containing a Quantum DLT7000 tape drive. The tape drive is attached to the SCSI adapter in slot 6 and has SCSI ID 4.

```
sysconfig -t

Tape drive (6.4)  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. Only the tape drive, as in the preceding example for the DLT7000, is visible.

Use quota report to determine amount of data

The filer `quota report` command enables you to determine the amount of data stored in filer qtrees.

Example: The following example showing the output from the quota report assumes

- Each tape stores up to 10 GB of data.
- The host used to dump and restore files supports dump files larger than 2 GB.

Using the information in the example, the dump would use two tape volumes:

- The first tape volume would hold a dump of the *mkt* and *news* subtrees.
- The second tape volume would hold a dump of the *etc*, *finance*, *home*, and *users* subtrees.

```
quota report

                              K-Bytes          Files
Type  ID  Volume Tree      Used Limit       Used Limit      Quota Specifier
----- --- ----- ----       ---- -----       ----- -----     ---------------
tree  6   vol1   finance   21608 1048576     1350 10240     /vol/vol1/finance
user 1086 vol2   -          9172 1048576      129 1048576   /vol/vol2/eirik
tree  3   vol2   -       3018520 10485760  403451 1048576   /vol/vol1/mkt
tree  4   vol1   news    6247100 20971520 1561775 2097152   /vol/vol1/news
tree  1   vol1   users   3018328 10485760  406421 1048576   /vol/vol1/users
tree  5   vol1   home    1546219 1048576   203946 1048576   /vol/vol1/home
```

Use du to determine amount of data to back up

The UNIX du command enables you to determine the amount of data used in directories.

Example: The following example checks the disk space used by directories in the filer root directory (*/vol/vol0*) mounted on an NFS client as */n/toaster*.

```
cd /n/toaster

du -s etc home

4108 etc
5510100 home
```

How the dump command uses snapshots

When dump uses snapshots

The `dump` command uses snapshots during the backup process.

When dump creates snapshots

The `dump` command creates a snapshot when the *subtree* parameter for the command specifies a file or directory that resides within an active file system.

The `dump` command names the snapshots that it creates *snapshot_for_backup.n*. The *n* at the end of the snapshot name

- is an integer that is incremented by one for each snapshot file the `dump` or `image` command creates
- gets reset to zero whenever you boot the filer

When dump does not create snapshots

The `dump` command does not create a snapshot when the subtree parameter for the command specifies an existing snapshot.

When dump deletes snapshots

The `dump` command automatically deletes snapshots that it creates when it finishes the dump.

If the system is rebooted during the dump: After the system reboots, do one of the following:

- delete the snapshot manually and run `dump` again.
- restart dump specifying the snapshot created by the previous dump command to get the same dump as you would have if the system had not rebooted

Multiple simultaneous dumps create multiple snapshots

You can initiate up to five current `dump` sessions, for example, four using `rsh` plus one using the console. Each `dump` session creates its own snapshot.

For example, the following files might exist on your filer when several `dump` commands are taking place: *snapshot_for_backup.0*, *snapshot_for_backup.1*, and *snapshot_for_backup.2*.

Do not delete snapshots during dumps

Do not delete a snapshot while `dump` is backing up the snapshot; this causes the filer to crash. If you invoke `dump` 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.

Avoiding crashes: To avoid such crashes, do one of the following actions:

- Ensure that `dump` finishes before the snapshot being dumped comes up for deletion by the snapshot scheduler.
- Rename the snapshot to protect it from the snapshot scheduler.

Specify dump names when dumping multiple subtrees

When dumping multiple subtrees, you must also specify a dump name. For more information about dumping multiple subtrees, see “Example 7” on page 324.

Backing up from a remote host

Backing up with tar and cpio

If the filer does not use the CIFS protocol, you can back up the filer using commands, such as `tar` and `cpio`, from a client system.

Caution

Some versions of `tar` and `cpio` have bugs that cause them to fail with file systems that contain long path names, unusual file names, or hard links.

Backup using dump recommended

Using the filer `dump` command is recommended as a more efficient and reliable way of backing up the file system.

The `dump` command creates backup files compatible with the SunOS 5.*x*/Solaris 2.*x* `ufsdump` format.

Backing up the filer using NDMP

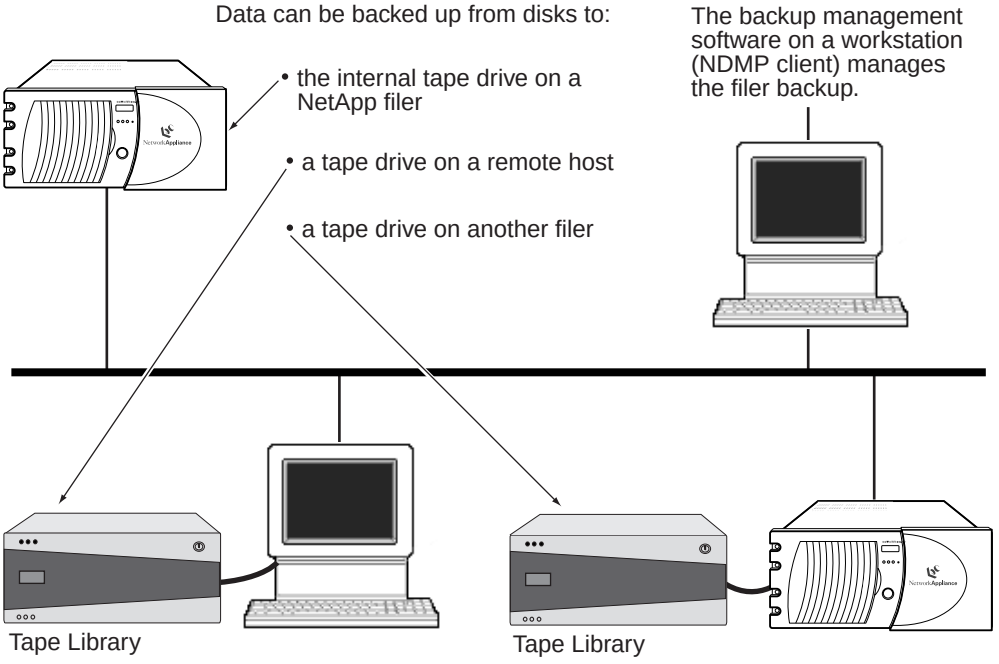
Devices you can back up the filer to

You can use third-party backup management programs that support NDMP, Version 1 or 2, to back up the filer. The filer can be backed up by NDMP to the following devices:

- tape drives, libraries, or stackers attached to the filer
- tape drives, libraries, or stackers attached to the workstation that runs the backup management software
- tape drives, libraries, or stackers attached to a workstation or filer on your network

Example

The following figure shows an example of backing up the filer to various devices using NDMP.



Introducing the ndmpd command

Use the `ndmpd` command for enabling NDMP support on the filer and for obtaining information about backup and restore operations that use NDMP.

The ndmpd command syntax

The syntax for the `ndmpd` command is as follows:

```
ndmpd [on|off|status|probe [session]|kill [session]|version [version]]
```

The ndmpd command options

The following table describes the `ndmpd` command options.

Option	Description
`on`	Enables NDMP. **Running NDMP after each reboot:** If you want the filer to run NDMP after each reboot, put the `ndmpd on` command in the */etc/rc* file.
`off`	Disables NDMP. After you enter `ndmpd off`, the filer continues processing existing requests but rejects new ones.
`version`	Enables you to change the version of NDMP used by the filer. Available versions are `1` and `2`. All sessions must be inactive to change the version.
`status`	Displays the current state of each of the five possible NDMP sessions. Sessions are numbered 0 to 4
`probe`	Displays detailed information about the NDMP session specified. Using the `probe` option without a `session` parameter displays NDMP information for all five sessions.
`kill`	Stops all processing in the specified session and puts the session in an inactive state. Use this option if you want to clear a hung NDMP session.

Example of version option

In the following example, the `version` command sets the NDMP version to `2`:

```
ndmpd version 2
```

Example of status option

In the following example, the `status` command shows that there is only one active session:

```
ndmpd status

ndmpd ON.
Session: 0
  Active
  version:    2
  Operating on behalf of primary host.
  tape device:    not open
  data state:     Idle
  data operation: None
  mover state:    Idle
Session: 1
  Inactive
Session: 2
  Inactive
Session: 3
  Inactive
Session: 4
  Inactive
```

Example of probe option

In the following example, the `ndmpd probe 0` command shows detailed status of active session 0:

```
ndmpd probe 0

ndmpd ON.
Session: 0
  pid:                    0xfffffc0000d75760
  isActive:               TRUE
  protocol version:       2
  effHost:                SK_HOST_LOCAL
  eof:                    FALSE
  authorized:             FALSE
  spt.device_id:          -1
  spt.ha:                 -1
  spt.scsi_id:            -1
  spt.scsi_lun:           -1
  tape.id:                -1
  tape.device:            not open
  recordCount:            0
  tape.mode:              Read only
  mover.state:            Idle
  mover.mode:             Write
```

```
mover.pauseReason:      N/A
mover.haltReason:       N/A
mover.dataWritten:      0
mover.seekPosition:     0
mover.bytesLeftToRead:  0
mover.windowOffset:     0
mover.windowLength:     18446744073709551615
mover.position:         0
mover.discardLength:    0
mover.recordSize:       0
mover.recordNum:        0
mover.rIndex:           0
mover.wIndex:           0
data.operation:         None
data.state:             Idle
data.haltReason:        N/A
data.abort:             FALSE
data.mover.addr_type    LOCAL
data.readOffset:        0
data.readLength:        0
```

Methods for restoring files

Two ways to restore files

You can restore a dump file created by the filer `dump` command in 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 drive, 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 Solaris `ufsrestore` command only if your filer runs NFS exclusively. If your filer runs the CIFS protocol, do not use the Solaris `ufsrestore` command. Doing so results in data loss.

When to use restore

Use the `restore` command in the following situations:

- Files have been deleted from the filer but have been backed up to tape.
- Some files on the filer are corrupted.
- No disk slots are available for storage expansion.
- The entire filer is damaged and unusable.

If files are corrupted: If some files are corrupted, you can restore the subtree containing the files.

If no disk slots are available for expansion: If the filer runs out of storage space, you can

- back up the entire filer
- replace the current disks with disks of greater capacity
- set up volumes on the new disks
- restore the filer from tapes

If the entire filer is damaged and unusable: If the entire filer is damaged and unusable, call Network Appliance Technical Support to see whether you

- can repair the filer. If you can repair the filer but some files are deleted during the repair process, you need only restore the missing files.

- cannot repair the filer. If you cannot repair the filer, you need to reinitialize the disks and restore all file systems from tape.

Caution

Before initializing disks on your filer, call Network Appliance Technical Support for instructions. Initializing disks destroys all data on your disks.

Restoring using NDMP

Restoring using NDMP enables you to use third-party backup management software to perform the restoration of data from either local or remote attached tape drives. NDMP uses `restore` and, therefore, restores both NFS and CIFS attributes.

The restore command syntax and options

About this section

This section describes the syntax and options for the `restore` command, and includes other information important to restoring files successfully.

The restore command syntax

The syntax for the `restore` command is as follows:

`restore [`*`function_key`*`] [`*`options`*`] [`*`arguments`*`] [`*`subtree`*`]`

Rules for using the restore command

Follow these rules when you enter the `restore` command:

- Specify no more than one function key.
- Specify multiple options without intervening spaces.
- Enter the arguments for each option in the order that you specify the options. Separate each argument from the next with a space.
- Place the subtree parameter after the last option argument.

The restore command function keys

The following table describes the function keys for the `restore` command.

Key	Meaning
`r`	Rebuilds the file system or subtree. If you are applying incrementals, this must be the only option specified.
`R`	Specifies a particular tape if the dump file is on a multitape-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 an individual file or subtree from the backup tape.

The restore command options

The following table describes the options for the `restore` command.

Option	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.
`D`	*pathname*	Specifies the absolute path name of a directory into which the files are restored. Without the path name, 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. The filer can restore files faster without the `v` option.
`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. This option is particularly useful if you use `restore` through `rsh`. This is because without the `y` option, if `restore` through `rsh` encounters a read error, it aborts immediately.
`A`		Specifies that `restore` does not restore ACLs.

The ufsrestore i argument is not supported

The filer `restore` command is similar to the Solaris `ufsrestore` command, except that the filer `restore` command doesn't support the `i` argument of the Solaris `ufsrestore` 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 `restore` command.

Incremental-only restores not supported

The `restore` command doesn't support incremental-only restores. You must begin the restoration from a level 0 dump.

Do not restore from tapes created by other computers

Do not attempt to restore files from tapes created by other servers or workstations.

Examples of the restore command

About this section

This section provides several examples of how to use the restore command.

Example: restoring a subtree

This example shows how to restore a subtree named */vol/vol0/etc/home* to a filer named toaster. The following information helps you understand the example:

- The subtree was backed up as a subtree, not as a directory within a subtree.
- This procedure uses the following tape device name:

  `unixhost:/dev/rst0a`

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 drive, specify the remote tape in the following format:

`hostname``:`*`device_name`*

Procedure for restoring /vol/vol0/etc/home/user: The following procedure restores */vol/vol0/etc/home/user*.

Step	Action
1	Delete all files in the subtree before the restore.
2	Install the tape that contains the level 0 dump file for the *home* subtree in a local or remote tape drive.
3	Enter the following command if the subtree is dumped to only one tape: **`restore rfD rst0a /vol/vol0/etc/home/user`** You do not have to specify the `D` option and */vol/vol0/etc/home/user* if the destination directory for the restore is the same as the directory that was backed up. This command restores the dump file in *rst0a* to the */vol/vol0/etc/home/user* directory, using the same block size as when the subtree was dumped to the tape. If backup of the subtree is contained in multiple tape volumes, `restore` prompts you for the next tape volume when appropriate.

Step	Action
4	After the restore is finished, remove the tape from the drive. Install another tape that contains the next lowest level of incremental dump. **Example:** If you have dump tapes containing level 1, level 2, and level 3 dumps of */vol/vol0/etc/home/user*, 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.
5	After the last incremental backup is restored, from a client, 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. The restore process is now complete.

Example: Restoring the entire filer

To restore the entire filer, repeat the procedure described in this section for each volume. This section assumes that you already initialized the disks on your filer.

If each disk volume was backed up to one tape volume: If each disk volume was backed up to one tape volume, follow the procedure in "Example: restoring a subtree" on page 342 to restore each volume. The only difference is that you use */vol/volume_name* as the directory to which the dump file is restored.

If each volume was backed up as subtrees or qtrees: If each volume was backed up as subtrees, restore each subtree. For example, if the root volume contains two directories, */vol/vol0/etc* and */vol/vol0/home*, and you backed them up as subtrees, follow these steps to restore the entire volume.

Step	Action

1	If you want to restore qtrees, create directories on the filer using the `qtree` command that is used as the top of the subtrees to be restored. For example:
	`qtree etc`
	`qtree home`

2	Install the tape that contains the level 0 dump file for the */vol/vol0/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.
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 the following format: *hostname*:*device_name* Following is the sample tape device name used in this procedure: `unixhost:/dev/rst0a`
4	Enter the following command on the filer: **`restore rfD rst0a /vol/vol0/etc`** This command restores the dump file in `rst0a` to the */vol/vol0/etc* directory. If the dump file for the subtree is in multiple tape volumes, `restore` prompts you for the next tape volume when appropriate. See "Examples of restoring from multiple tapes" on page 345 for a sample screen display when more than one tape contains the dump volume.

Step	Action
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 */vol/vol0/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.
6	From the client, 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.
7	Repeat Steps 4 through 6 for the */vol/vol0/home* subtree.
8	If the filer contains multiple volumes, repeat Steps 1 through 7 for each volume. **Note** If your filer currently contains multiple volumes but the system backup was made before multiple volumes were installed, specify the path name into which the dump file is to be restored. For example, when the filer contained a single volume, you backed up the */home* directory. Now you want to restore */home* from tape to */vol/engineering/home*. Specify */vol/engineering/home* as the target directory in the `restore` command. If you don't specify the directory, the *home* dump file is restored to */vol/vol0/home.*

Examples of restoring from multiple tapes

This section provides additional examples illustrating how to restore a volume or a subtree that was backed up to multiple tapes.

Restoring a volume from multiple tapes to a directory: In this example, you backed up the *vol0* volume to two tapes using the following command:

```
dump 0fB rst0a,rst1a 600 /vol/vol0

DUMP: creating "snapshot_for_backup.1" snapshot.
DUMP: Date of this level 0 dump: Wed Jan 21 20:28:07 1998
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /vol/vol0/ to rst0a,rst1a
DUMP: mapping (Pass I)[regular files]
```

```
DUMP: mapping (Pass II)[directories]
DUMP: estimated 857 tape blocks on 1.40 tapes
DUMP: Dumping volume 1 on rst0a
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: Dumping volume 2 on rst1a
DUMP: Volume 2 begins with blocks from inode 2880
DUMP: 821 tape blocks
DUMP IS DONE
DUMP: deleting "snapshot_for_backup.1" snapshot.
```

At the time of the restore, the filer prompts you for the next tape device.

Restoring a volume to a directory from multiple tapes using two tape drives: In this example, the volume is restored into a directory named */vol/vol0/myexample*, and the two tapes are in different tape drives:

```
restore rfD rst0a /vol/vol0/myexample

Mount tape volume 2
Enter “none” if there are no more tapes
otherwise enter tape name (default: rst0a) rst1a
```

Restoring a volume to a directory from multiple tapes using one tape drive: If the two tapes use the same tape device, `rst0a` for example, remove the tape currently in the tape drive and load the next tape. Then accept the default tape device name, in this example `rst0a` when the filer prompts you for the name of the device for the second tape.

Restoring a named file from multiple tapes

If you want to restore a specific file or directory from a subtree, use the `x` option in the `restore` command. Specify the path name of the file or directory relative to the subtree that was backed up.

Example: If the *vol0* volume was backed up as a subtree and you want to restore all contents of the */vol/vol0/test* directory, specify */test* as the path name in the `restore` command.

Example: The subtree containing the directory to be restored was backed up to two tapes:

```
restore xf rst0a /test
```

In this command, you must specify the tape device for the first tape; it is `rst0a` in this example. This is necessary because `restore` needs to read information about the directory structure of the subtree from the first tape before restoring the data.

The filer then displays the following messages to let you specify the tape devices containing the multiple tape volumes. When the filer prompts you for the tape volume number, start with the last tape volume, as shown in this example:

```
You have not read any tapes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 2
Mount tape volume 2
Enter “none” if there are no more tapes
otherwise enter tape name (default: rst0a) rst1a
You have read volumes: 2
Specify next volume #: 1
Mount tape volume 1
Enter “none” if there are no more tapes
otherwise enter tape name (default: rst1a) rst0a
```

After prompting for the tape devices, the filer also displays the following question:

```
set owner/mode for '.'? [yn] y
```

To keep the original owner and permission modes for the restored files and directories, enter **y**.

Restoring from the filer

About this section

This section describes how to restore files by entering the `restore` command at the filer console or through a `telnet` session to the filer.

Procedure

Follow the steps in the following table to restore files to the filer.

Step	Action
1	If the filer contained qtrees, re-create the qtrees that existed when you backed up the filer by entering the following command: **`qtree` *`name`*** where *name* is the name of a qtree. **Note** You can enter the following command to generate a complete list of qtrees in an active file system: **`qtree`** If you cannot access an active file system, use the record of the */etc/quotas* file that you made when you created qtrees.
2	Place the tape containing the first volume of the backup in the tape drive.
3	Enter the `restore` command as follows: **`restore [`*`function_key`*`] [`*`options`*`] [`*`arguments`*`] [`*`subtree`*`]`** Specify the appropriate function key, options, arguments, and subtree.
4	If prompted, insert the next tape in the backup.
5	Repeat Step 4 until the restore is complete.

Restoring from a remote host

About this section

This section describes how to restore files by running the `restore` command through the `rsh` command.

Restoring with rsh from trusted hosts

You can enter the `restore` command on a trusted host through `rsh`. In addition, you can run multiple `restore` commands by starting each from a remote shell. The filer supports up to two simultaneous restores.

Remote restore requires host name lookup

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.

Using a filer tape drive from another system

About this section

This section discusses how you can use a filer tape drive to dump and restore files on a UNIX system or on another filer.

Remote host access required

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:

- The */etc/hosts.equiv* file on the filer with the tape drive. Alternatively, you can specify both the host and user in */etc/hosts.equiv.* The filer */etc/hosts.equiv* file contains entries in this format:

  ```
  hostname [ username ]
  ```

  For more information about */etc/hosts.equiv*, see the *na_hosts.equiv*(5) man page.
- The */etc/hosts* file 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* file. See the *Getting Started Guide* for information about how to add the filer to the client's */etc/hosts* file.

Format for specifying filer tape drive

In the commands for dumping and restoring, specify the filer tape drive in the following format:

`filer``:`*`device_name`*

Examples

The following examples show how to use a filer tape drive from an SVR4 UNIX system such as Solaris 2.*x*:

- To dump a file system that resides on a UNIX system

  ```
  ufsdump 0uf toaster:nrst0a /dev/sd0g
  ```
- To restore a directory from a filer tape drive to a UNIX system

  ```
  cd /filesystem
  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

  ```
  ufsdump 0uf toaster:nrst0a /dev/sd0g
  ```

- To restore a directory from a filer tape drive to a UNIX system

  ```
  cd /filesystem
  ufsrestore rf toaster:rst0a
  ```

To use a filer tape drive from another filer, follow the directions in "Example 7" on page 324, "Example: restoring a subtree" on page 342, and "Example: Restoring the entire filer" on page 343 to dump or restore files, using *filer*:*device_name* as the tape drive name in these procedures.

Controlling tape devices

Use the mt command

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 (`rewind`).
- Take a tape off-line (`offline`).
- Display status information about a tape device (`status`).

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 as follows:

```
mt [-f|-t] tapedevice command [ count ]
```

This section discusses only the `eom`, `fsf`, `rewind`, `offline`, and `status` commands. Keep in mind the following information:

- The `-f` and `-t` options are interchangeable as far as the filer is concerned. Only the `-f` option is shown here.
- For additional information about 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 device. To do so, use the `eom` command, which moves the tape to the end of data (end of media if tape is full), as in the following example:

```
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 on a tape device, use the `fsf` command, as in the following example:

```
mt -f nrst0a fsf 1
```

Rewinding a tape

To rewind a tape, use the `rewind` command, as in the following example:

```
mt -f nrst0a rewind
```

Taking a tape drive off-line

To rewind the tape and, if appropriate, take the tape drive off-line by unloading the tape, use the `offline` command, as in the following example:

```
mt -f nrst0a offline
```

Displaying status information

To display status information about the tape unit, use the `status` command, as in the following example:

```
mt -f nrst0a status

Tape drive: Quantum DLT7000
Status: ready, write enabled
Format: 85937 bpi 70 GB (w/comp)
fileno = 0  blockno = 95603  resid = 0
```

Copying One Volume to Another 14

Chapter contents

This chapter describes the filer's commands for copying one volume to another. The following table lists the topics discussed in this chapter.

Read...	To learn about...
"Introduction to the filer's commands for copying volumes" on page 356	The possible scenarios where you might want to copy a volume to another volume and the advantages of using the commands for copying volumes.
"Requirements and recommendation for copying a volume" on page 358	What you need to do before entering a command for copying a volume.
"Procedure for copying one volume to another" on page 361	The procedure for copying one volume to another.
"Managing a volume copy operation when it is in progress" on page 366	Checking the status of a volume copy operation, stopping a volume copy operation, and changing the speed of a volume copy operation.

Introduction to the filer's commands for copying volumes

Purpose of the vol copy command set

The filer's SnapCopy™ feature enables you to copy one volume to another through the `vol copy` command set. The commands in this command set control copying both data in the active file system and data in snapshots from one volume to another. The source and destination volumes of the copy can reside on the same filer or on different filers.

For more information about snapshots, see Chapter 10, "Snapshots."

When to copy volumes

The following table describes some situations where you might find copying volumes useful.

Situation	Reasons for copying one volume to another
You want to copy data from one filer to another regularly to ensure high data availability.	After you copy the data, clients can switch to the destination filer in the following scenarios: ◆ When you shut down the source filer for software or hardware upgrades. ◆ If a network client process accidentally deletes a large number of files on the source filer, requiring you to shut down network service on the filer to restore data.
You want to migrate data from one filer to another.	The destination filer has more storage.

Situation	Reasons for copying one volume to another
You want to move a volume from one set of disks to another on the same filer.	◆ Splitting a volume. **Example:** You can copy the *vol0* volume to the *vol1* volume and then delete duplicated files and directories in these volumes so that the original contents of *vol0* are split into two volumes. ◆ Expanding filer storage. **Example:** You have eight 4-GB disks for the *vol0* volume and four 9-GB spare disks. You can migrate *vol0* to the four 9-GB disks and replace all 4-GB disks with larger capacity disks. ◆ Avoiding reduced performance due to disk errors. **Example:** If the disks in a particular volume have been reporting errors, you might want to migrate the volume to a new set of disks to reduce the number of read and write retries or to reduce the likelihood of RAID reconstruction. After the filer moves the data to the new disks, you can replace those disks that report errors.

Benefits of the vol copy command set

Although you can copy data on the filer using client programs such as `cpio` or using the filer's `dump` and `restore` commands, the `vol copy` command set offers the following benefits:

- When a `vol copy` command reads and writes data, the filer does not traverse directories on the filer. Data is copied block for block directly from the disks, which means that the filer can finish the copying faster than it could with other methods.
- Using a `vol copy` command, the filer preserves the snapshot data of the source volume. If, in the future, users might need to use snapshots that were taken before data was copied from one volume to another, for example, if users accidentally delete files and need to recover them, use a `vol copy` command for migrating data.

Requirements and recommendation for copying a volume

Requirements for copying a volume

The filers involved in a volume copy operation must meet several requirements for data to be copied successfully. The following list is a brief description of these requirements. The rest of this section provides more detailed information about verifying whether the source and destination volumes meet these requirements.

- The source volume must be on-line and the destination volume must exist and be off-line.
- The destination volume must not be the root volume because the destination volume must be off-line when the filer executes the `vol copy` command, and a root volume must always be on-line.
- The capacity of the destination volume must be greater than or equal to the capacity of the source volume.
- The maximum number of files for the destination volume must be equal to or greater than the maximum number of files for the source volume.

  To check the maximum number of files, enter the following command for the source and destination volumes, replacing *pathname* with the path to a volume:

  `df -i` ***`pathname`***

  To increase the maximum number of files on a destination volume, enter the following command, replacing *volume_name* with the path to the volume and *number* with the maximum number of files desired:

  `maxfiles` ***`volume_name number`***
- To copy a volume between filers, the destination filer must have at least as much RAM as the source filer.

  If the destination filer does not have as much RAM as the source filer, the source filer displays the following message:

  ```
  VOLCOPY: Error writing to standard output : Broken pipe
  ```

  The destination filer displays the following message:

  ```
  VOLCOPY: No buffers available.
  ```
- The destination volume must not contain data that you want to preserve.
- The source and destination filers must have a trusting relationship with each other.
- The localhost interface must be specified as a trusted host if data is copied on the same filer.

Verifying the status of each volume

The destination volume must exist before you enter the `vol copy start` command to start copying a volume. If the volume does not exist, the command does not create the volume and the command returns an error. See "Creating volumes" on page 91" for information about how to create a volume.

After you verify that the destination volume exists, check the status of the source volume and the destination volume.

Checking the status of a volume: The source volume must be on-line and the destination volume must be off-line. To verify whether a volume is on-line or off-line, use the following command syntax:

`vol status` *`volume_name`*

Changing the status of a volume: If you need to change the status of a volume, use one of the following command syntaxes:

- `vol offline` *`destination_volume`*
- `vol online` *`source_volume`*

Note

The `vol offline` command takes effect only after you reboot the filer.

Verifying the size of each volume

The capacity of the destination volume must be greater than or equal to the capacity of the source volume, regardless of how many snapshots you try to copy. To verify the capacity of a volume, follow these steps:

1. Enter the following command syntax:

   `df /vol/`*`volume_name`*

   The command displays information about disk space in the active file system and in the snapshot for the specified volume.

2. Add the numbers in the kbytes column in the `df` output. The result is the number of kilobytes of data that the volume can hold.

Verifying the contents of the destination volume

If the destination volume is not a new volume, make sure that the destination volume does not contain data that you might need in the future. This is because after the filer starts copying the source volume, it overwrites the entire destination volume. That is, all data in the active file system and in the snapshot of the destination volume will be lost after the filer starts copying the data.

Verifying the relationship between filers

If the source and destination volumes in a volume copy operation reside on two filers, the filers must have a trusting relationship with each other. That is, you must specify each filer as a trusted host in the */etc/hosts.equiv* file of the other filer. The */etc/hosts.equiv* file contains a list of host names, each of which is on a separate line.

Verifying localhost as a trusted host

If you want to copy data between volumes on the same filer, you must specify localhost on the filer as a trusted host in the filer's */etc/hosts.equiv* file. Localhost is the interface through which the filer sends packets to itself.

If you have not already listed localhost as a trusted host, follow these steps to include localhost in */etc/hosts.equiv*:

1. Enter the loopback address, which is 127.0.0.1, in the */etc/hosts* file.
2. Type `localhost` on a separate line in the */etc/hosts.equiv* file.

Recommendation for copying a volume

To avoid performance problems when copying to a different filer, you might want to set up a private network for copying between the source and destination filers. This is because when a filer copies data between two volumes, it floods the network with packets. Users of the filers involved in a volume copy operation might notice a degradation in response time during the copy.

Procedure for copying one volume to another

Command syntax for copying one volume to another

To copy one volume to another, use the following command syntax:

`vol copy start [ -S | -s` *`snapshot`* `]` *`source destination`*

Specifying the snapshots to copy: The `-S` and `-s` arguments specify the snapshots to copy. The following table describes the snapshots to copy and the resulting snapshots on the destination volume, depending on the argument.

Argument used	Snapshots to copy from the source volume	Snapshots in the snapshot file system of the destination volume
None	The snapshot taken after you enter the `vol copy start` command	A snapshot named *snapshot_for_backup.n*, where *n* is a number starting at 0
`-S`	All snapshots in the snapshot file system of the source volume and the snapshot taken after you enter the `vol copy start` command	All snapshots in the source volume and *snapshot_for_backup.n*, where *n* is a number starting at 0
`-s` followed by the name of the snapshot	The specified snapshot	The specified snapshot

Note

The `vol copy start -S` command does not copy any snapshots that are created when the copying is in progress. For example, if the copying lasts from 11:45 p.m. to 1:00 a.m. the next day, and the filer creates a snapshot named *nightly.1* at midnight, the filer does not copy the *nightly.1* snapshot.

Specifying the volumes involved in the copy: The *source* and *destination* arguments are the names of the volumes. If a volume is on a different filer, precede the volume name with the filer name and a colon. For examples illustrating how to specify volume names, see "Examples of the vol copy start command" on page 362.

Where to enter the vol copy start command

If the copying takes place between two filers, you can enter the command on either the source or destination filer. You cannot, however, enter the command on a third filer that does not contain the source or destination volume.

Examples of the vol copy start command

The following table shows several examples of the `vol copy start` command.

If you want to...	Enter...
Copy all snapshots from the *vol0* volume to the *vol1* volume on the same filer	`vol copy start -S vol0 vol1`
Copy a nightly snapshot from the *vol0* volume to the *vol1* volume on the same filer	`vol copy start -s nightly.1 vol0 vol1`
Create a snapshot in the *vol0* volume to be copied to the *vol1* volume on the same filer	`vol copy start vol0 vol1`
Copy all snapshots from the *vol0* volume to the *vol1* volume on a different filer named toaster1	`vol copy start -S vol0 toaster1:vol1`

Results of the vol copy start command

The `vol copy start` command generates volume copy operations and produces screen messages showing the progress of the operations.

Volume copy operations: Each `vol copy start` command generates two volume copy operations, as described in the following list:

- One operation is for reading data from the source volume. Screen messages displayed by a `vol copy` command refer to this operation as the `volcopy dump` operation.
- One operation is for writing data to the destination volume. Screen messages displayed by a `vol copy` command refer to this operation as the `volcopy restore` operation.

The filer assigns a volume copy operation number to each operation.

When to use the volume copy operation number: You need the volume copy operation number if you want to stop a volume copy operation or change the volume copy operation speed.

For more information about obtaining the volume copy operation number, see "Checking the status of a volume copy operation" on page 366.

Screen messages from the vol copy command: When the filer is copying a volume, the filer displays messages indicating the percentage of the volume copy operation completed and the number of minutes remaining. When the filer finishes copying all data, it displays the filer prompt.

Maximum number of simultaneous volume copy operations

Each filer supports up to four simultaneous volume copy operations. Whether a filer can execute a `vol copy start` command depends on how many volume copy operations are already in progress on the filer or filers specified in the `vol copy start` command, as illustrated in the following examples.

Example: You can enter the following two `vol copy start` commands on a filer to copy volumes locally:

```
vol copy start vol0 vol1
vol copy start vol2 vol3
```

When these commands are in progress, you cannot enter additional `vol copy start` commands because four volume copy operations are already running on the filer. Two of the operations are for reading the *vol0* and *vol2* volumes, and two of the operations are for writing the *vol1* and *vol3* volumes.

Example: Suppose you enter the following three `vol copy start` commands on a filer named toaster1 to copy volumes to another filer named toaster2.

```
vol copy start vol0 toaster2:vol0
vol copy start vol1 toaster2:vol1
vol copy start vol2 toaster2:vol2
```

When these commands are in progress, toaster1 runs three volume copy operations to read the volumes, and toaster2 runs three volume copy operations to write the volumes.

You can enter an additional `vol copy start` command to copy between toaster1 and toaster2 because the command adds one more volume copy operation to each filer.

However, you cannot enter an additional `vol copy start` command to copy volumes locally on either toaster1 or toaster2. This is because the additional command would create two volume copy operations, one for reading and one for writing, on the filer that performs the local copying. The filer cannot support these two additional volume copy operations because three operations are already in progress.

Possible errors

If your filer does not meet a requirement described in "Requirements and recommendation for copying a volume" on page 358, the `vol copy start` command generates one or more error messages. The following table explains the meanings of the possible error messages.

Error message	Meaning

`Permission denied.`
 	
`VOLCOPY: Could not connect to filer 127.0.0.1.`
	The source filer does not have permission to copy to the destination filer.
	
	Action: Make sure that the filers have a trusting relationship with each other. If both the source volume and destination volume are on the same filer, remember to include localhost in the */etc/hosts.equiv* file.

Error message	Meaning
`VOLCOPY: volcopy restore: volume is online, aborting`	The destination volume is on-line. **Action:** Take the destination volume off-line and reboot the filer that contains the destination volume.
`VOLCOPY: volcopy restore: volume is too small, aborting`	The destination volume is smaller than the source volume. **Action:** Add more disk space to the destination volume or choose another destination volume of sufficient capacity.

Managing a volume copy operation when it is in progress

Checking the status of a volume copy operation

You can use the following command syntax to check the status of one or more volume copy operations:

```
vol copy status [ operation_number ]
```

This command displays the status for a specified volume copy operation. If you do not specify the operation number, the command displays the status of all volume copy operations in progress. In the command output, the operations are differentiated from one another with unique volume copy operation numbers, ranging from 0 to 3. For more information about volume copy operation numbers, see “Results of the vol copy start command” on page 362.

Where to enter the vol copy status command: When a volume copy operation is in progress, you can enter the `vol copy status` command only through `rsh` because you do not have access to the filer prompt on the console.

If data is being copied between two filers, you can enter this command through an `rsh` connection to either filer. The operation numbers displayed on the source filer and the destination filer are different because the reading and the writing are considered two different operations.

Example of a vol copy status command

The following example illustrates a `vol copy start` command that copies the *vol0* volume to the *vol1* volume on the same filer:

```
vol copy start -S vol0 vol1

Copy Volume: vol0 on machine 127.0.0.1 to Volume: vol1
Reading the dump stream
VOLCOPY: Starting on volume 1.
This dump contains 257 blocks
10:04 pm : volcopy restore 1 : begun.
10:04 pm : volcopy restore 1 : 5 % done. Estimate 3 minutes
remaining.
.
.
.
10:04 pm : volcopy restore 1 : 95% done. Estimate 1 minutes
remaining.
```

Before the filer prompt is displayed again, you can use the `vol copy status` command on a trusted host of the filer, as shown in the following example:

```
rsh toaster vol copy status

10:04 pm : volcopy dump 0 : 99 % done. Estimate 1 minutes remaining.
10:04 pm : volcopy restore 1 : 99 % done. Estimate 1 minutes
remaining.
No operation 2 in progress.
No operation 3 in progress.
```

In this example, volume copy operation 0, shown as `volcopy dump 0` in the display, is for reading the data from the *vol0* volume; volume copy operation 1, shown as `volcopy restore 1` in the display, is for writing the data to the *vol1* volume.

Aborting a volume copy operation

To stop a volume copy operation, use the following command syntax:

`vol copy abort [ operation_number ]`

The *operation_number* parameter specifies the volume copy operation to be aborted. You can obtain the operation number from the `vol copy status` output.

Caution

If you specify no operation number in the command, the filer aborts *all* volume copy operations. It does not display a help string for the `vol copy abort` command.

If data is being copied between two filers, you can execute this command on either filer.

You can enter the `vol copy abort` command only through `rsh` because you do not have access to the filer prompt on the console.

Caution

An incomplete volume copy operation leaves unusable data in the destination volume.

Controlling the speed of a volume copy operation

You can control the speed of a volume copy operation in two ways.

- Before you enter the `vol copy start` command, use the following command syntax to set the default speed for all volume copy operations:

```
options vol.copy.throttle value
```

The *value* variable specifies the speed, which ranges from 10 (full speed) to 1 (one-tenth of full speed).

- If a volume copy operation is in progress, use the following command syntax to set the speed of a specific operation:

  ```
  vol copy throttle [ operation_number ] value
  ```

  In this case, because a volume copy operation is in progress and you do not have access to the filer prompt, you must enter the `vol copy throttle` command through `rsh`.

  The *operation_number* parameter specifies the volume copy operation whose speed you want to adjust. If you do not specify an operation number, the command applies to all volume copy operations that are in progress. The *value* variable specifies the speed, which ranges from 10 (full speed) to 1 (one-tenth of full speed).

Note

The speed for reading data from the source volume and the speed for writing data to the destination volume can be different. In this case, the smaller of the two values determines the time required for the filer to finish copying the data.

Displaying the default speed for copying a volume

Before starting the `vol copy start` command, you can verify the default speed for all volume copy operations using the following command:

```
options vol.copy.throttle
```

It displays the value (1 through 10) to be used by all volume copy operations. The value of the `vol.copy.throttle` option was set at 10 at the factory.

Example of controlling the speed of copying a volume

The following example illustrates changing the speed of all volume copy operations in progress to one-tenth of full speed through `rsh`:

```
rsh toaster vol copy throttle 1

volcopy operation 0: Throttle adjusted from 100% to 10%.
volcopy operation 1: Throttle adjusted from 100% to 10%.
```

System Information and Performance 15

Chapter contents

This chapter discusses the topics described in the following table.

Read...	To learn about...
"Displaying the Data ONTAP version" on page 370	How to use the `version` command to display the Data ONTAP version.
"Displaying filer configuration information" on page 371	How to use the `sysconfig` command to display filer configuration information.
"Displaying volume information" on page 373	How to use the `vol` command to display information about a volume's configuration.
"Displaying filer statistics" on page 374	How to use the `sysstat` and `uptime` commands to display filer statistics.
"Displaying network statistics" on page 376	How to use the `netstat` command to display network statistics.
"Displaying interface statistics" on page 377	How to use the `ifstat` command to display interface statistics.
"Improving filer performance" on page 387	How to improve filer performance by adding hardware or setting software options.

Displaying the Data ONTAP version

How to display 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 follows:

```
version

NetApp Release 5.1: Fri May 12 03:06:00 PDT 1998
```

Displaying filer configuration information

Use the sysconfig command

The `sysconfig` command displays information about the filer's hardware configuration. The exact types of information displayed depend on the command options.

Displaying disk information using sysconfig -d

The `sysconfig -d` command displays product information about each disk in the filer.

Displaying RAID information using sysconfig -r

The `sysconfig -r` command displays RAID configuration information about the parity disk, 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 "Using disks of various sizes" on page 79 for more information about disk identifiers.
- Determining how much space on each disk is available to the filer. Refer to "Understanding usable space on each disk" on page 80 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.

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 97.

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.

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. This information varies according to the devices on the filer. 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.

Displaying overall filer information using sysconfig

If you enter `sysconfig` without any options, information similar to what you get with `sysconfig -v` is displayed, but the information is abbreviated. When you report a problem to Network Appliance, provide the information displayed by `sysconfig -v`. This information is useful for diagnosing system problems.

Displaying volume information

Use the vol status command

The `vol status` command displays information about a volumes's configuration. The types of information displayed depend on the command options. When you specify a volume, the information for that volume is displayed; when you do not specify a volume, the status of all volumes in the filer is displayed.

Displaying volume state information with vol status

With no options, the `vol status` command displays a one-line synopsis of volume states. This includes the volume name, whether it is on-line or off-line, and other states, for example, partial, degraded, and so on.

Displaying disk information using vol status -d

The `vol status -d` command displays information about disks. The disk information is the same as the information from the `sysconfig -d` command.

Displaying RAID information using vol status -r

The `vol status -r` command displays a list of the RAID information. The display is the same as the `sysconfig -r` display.

Displaying RAID information for each group using vol status -v

The `vol status` -v command displays information about each RAID group.

Displaying filer statistics

Use the sysstat and uptime commands

You use the `sysstat` and `uptime` commands to display filer statistics.

About 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.

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 might improve your filer's performance:

- Typically, 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 might help improve system performance. If there are short spurts of writes, adding NVRAM does 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 the Cache age column are consistently below 5, the filer might 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 "About the uptime command" on page 375. 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 and 10 MB per FDDI adapter), you might need to add another network adapter card to the system.

About 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.

Example: An example of the display is

```
uptime

8:54am up 2 days 22:23, 3122520 NFS ops
```

Displaying network statistics

Use the netstat command

You use the `netstat` command to display network statistics.

About the netstat command

The `netstat` command displays network-related data in various output formats.

The `netstat -i` and `netstat -I` options show the state of all network interfaces or one specific interface, respectively. The `netstat -r` command shows the filer's routing table.

For information about troubleshooting network problems, see "Network problems" on page 412. For more information about the `netstat` command, see the *na_netstat*(1) man page.

Displaying interface statistics

Use the ifstat command

The `ifstat` command prints per-interface statistics not reported by commands such as `netstat`. This includes some statistics maintained by the networking code, as well as statistics maintained by the driver and by the networking card.

The output of the `ifstat` command might contain many fields, because different types of interfaces, for example, Ethernet, Gigabit (GB) Ethernet, FDDI, and ATM, have different statistics.

ifstat syntax: The syntax for the `ifstat` command is

`ifstat [-z]  -a | interface`

The `-z` option "zeros" (or clears) the statistics. The `-a` option lists statistics for all the filer's interfaces. The interface option indicates the type of interface for which you want statistics.

Explanation of interface statistics

Ethernet: The following table describes the statistics in the RECEIVE section of the `ifstat` command output when you use the command on an Ethernet interface.

Statistic	Meaning
`Packets`	Number of packets received on the interface.
`Bytes`	Number of bytes received on the interface.
`Errors`	Number of errors during Ethernet frame reception, including all kinds of receive errors.
`No buffers`	Number of received packets dropped due to the unavailability of buffers.
`Length err`	Number of frames truncated due to the shortage of receive descriptors.
`Runt frames`	Number of runt frames.

Statistic	Meaning
`Long frames`	Number of frames received that exceeded the maximum Ethernet-specified size of 1,518 bytes.
`CRC error`	Number of CRC errors that occurred on the received frames.
`H/w overflow`	Number of frames discarded because of receive FIFO overflow.
`Process stop`	Number of times the receive process stopped.
`List overflow`	Number of frames dropped due to the unavailability of descriptors.
`Process reset`	Number of times the receive process was reset.
`Rst frame drops`	Number of frames discarded due to the resetting of the receive process.

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output when you use the command on an Ethernet interface.

Statistic	Meaning
`Packets`	Number of packets attempted to be transmitted.
`Bytes`	Number of bytes attempted to be transmitted.
`Errors`	Number of hardware errors encountered while attempting to transmit.
`Collisions`	Number of collisions that occurred while transmitting frames.
`Late collisions`	Number of collisions aborted due to a late collision.
`Excess coll`	Number of times transmission was aborted due to excessive collisions.
`Queue full`	Number of times the queue was full.
`List full`	Number of frames that were dropped due to the unavailability of descriptors.

Statistic	Meaning
`No carrier`	Number of times the carrier signal was not present during transmission.
`Underflow`	Number of times the transmitter aborted the message because data arrived late from memory.
`Defer`	Number of times transmission had to be deferred.
`Time out`	Number of times the transmit jabber timer expired.
`Stopped`	Number of times the receive process stopped.
`List underflow`	Number of frames that had to be dropped due to the unavailability of descriptors.
`Loss of carrier`	Number of times the carrier was lost.
`No buffers`	Number of times buffers were unavailable.
`Requeue`	Number of times the frame was requeued due to list underflow.
`Threshold up`	Number of times the transmit threshold was increased.
`Threshold dn`	Number of times the transmit threshold was decreased.
`Threshold lvl`	Current threshold level.

The following table describes the statistics in the DEVICE section of the `ifstat` command output when you use the command on an Ethernet interface.

Statistic	Meaning
`Interrupts`	Number of times the Ethernet device interrupted the host.
`Resets`	Number of times the Ethernet device was reset.

The following table describes the statistics in the LINK INFO section of the `ifstat` command output when you use the command on an Ethernet interface.

Statistic	Meaning
Auto	Auto-Negotiation state.
Mediatype	Media type, such as twisted pair.
Link Partner	The Auto-Negotiation capability of the remote end. It is unknown if Auto-Negotiation is disabled.
Link State	Link status.

GB Ethernet: The following table describes the statistics in the RECEIVE section of the `ifstat` command output when you use the command on a GB Ethernet interface.

Statistic	Meaning
Packets	Number of packets received on the interface.
Bytes	Number of bytes received on the interface.
Errors	Number of errors during Ethernet frame reception, including all kinds of receive errors.
Queue full	Number of received packets dropped due to the unavailability of buffers.
Unicast packets	Number of unicast packets received.

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output when you use the command on a GB Ethernet interface.

Statistic	Meaning
Packets	Number of packets attempted to be transmitted.
Bytes	Number of bytes attempted to be transmitted.
Errors	Number of hardware errors encountered while attempting to transmit.

Statistic	Meaning
`Collisions`	Number of collisions that occurred while transmitting frames.
`Unicast packets`	Number of unicast packets transmitted.

The following table describes the statistics in the DEVICE section of the `ifstat` command output when you use the command on a GB Ethernet interface.

Statistic	Meaning
`Received errors`	Number of errors encountered during reception by the interface.
`Transmit errors`	Number of errors encountered during transmission by the interface.
`Collisions`	Number of collisions encountered during transmission by the interface.

FDDI: The following table describes the statistics in the RECEIVE section of the `ifstat` command output when you use the command on an FDDI interface.

Statistic	Meaning
`Packets`	Number of packets received on the interface.
`Bytes`	Number of bytes received on the interface.
`Errors`	Number of errors during Ethernet frame reception, including all kinds of receive errors.
`Queue full`	Number of received packets dropped due to the unavailability of buffers.
`H/w faults`	Number of hardware faults.
`SA matches`	Number of SA matches.
`DA nonmatches`	Number of DA non-matches.

Statistic	Meaning
PDQ aborts	Number of PDQ aborts.
Invalid lengths	Number of invalid lengths.
Fragments	Number of fragments.
Format errs	Number of frames received that contained an odd number of symbols; that is, the number of non-integral octets.
MAC resets	Number of MAC resets.
CRC Errors	Number of frames received with a bad frame check sequence.
Status Errors	Number of frames received that had the E (error detected) indicator set or a missing E indicator.
Long Frames	Number of frames received that were larger than the maximum legal size.
Short Frames	Number of short frames received.
Copied buffers	Number of copied buffers.
No buffers	Number of times there were insufficient buffers.
LLC errors	Number of LLC errors.
Blk check errs	Number of block check errors.
Length errors	Number of length errors.
Overruns	Number of times that the interface received a frame loss indication from the hardware on the interface.

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output when you use the command on an FDDI interface.

Statistic	Meaning
Packets	Number of packets attempted to be transmitted.
Bytes	Number of bytes attempted to be transmitted.

Statistic	Meaning
`Errors`	Number of hardware errors encountered while attempting to transmit.
`Queue full`	Number of received packets dropped due to unavailability of buffers.
`No buffers`	Number of times buffers were not available.
`No buf resets`	Number of times the process was reset due to the unavailability of buffers.
`No descriptors`	Number of times descriptors were unavailable.
`I/f down`	Number of times the interface was down.
`Underruns`	Number of times an underrun occurred because the interface's transmit FIFO became empty due to insufficient memory. That is, the device tried to transmit but could not get data into memory fast enough.
`Failures`	Number of frames that were transmitted with an error.

The following table describes the statistics in the DEVICE section of the `ifstat` command output when you use the command on an FDDI interface.

Statistic	Meaning
`Interrupts`	Number of times the FDDI device interrupted the host.
`Spurious ints`	Number of spurious interrupts received from the device.
`Resets`	Number of times the device was reset due to unrecoverable hardware faults.
`Non-existing mem`	Number of times the device was asked to address nonexisting memory. This might indicate a software or hardware problem.
`Parity errs`	Number of parity errors detected on the interface.

Statistic	Meaning
`Host parity errs`	Number of parity errors detected on the host.
`Spurious Unsols`	Number of unknown unsolicited error events reported by this link entity.
`Ring inits in`	Number of ring claim processes or ring beacon processes initiated by this link entity.
`Ring beacon out`	Number of ring beacon processes initiated by this link entity.
`Ring beacon in`	Number of ring beacon processes recognized by this link entity.
`Dupl. address`	Number of times the duplicate address test failed.
`Dupl. token`	Number of times this link entity detected a duplicate token.
`Ring purge errs`	Number of times the MAC PurgeError Event bit was set.
`FCI strip errs`	Number of times a Frame Content Independent Strip operation was terminated.
`Trace init`	Number of PC Traces initiated by this station.
`Selftest failures`	Number of failed self- tests during device resets.
`PC trace rcvd`	Number of PC traces received by this station.
`Config changes`	Number of config changes.
`LEM rejects`	Number of LEM rejects.
`Elastic buf errs`	Number of ELM Chip EBUFF_ERROR events.
`Link test rejects`	Number of link test rejects.

ATM: The following table describes the statistics in the RECEIVE section of the `ifstat` command output when you use the command on an ATM interface.

Statistic	Meaning
`Packets`	Number of packets received on the interface.
`Bytes`	Number of bytes received on the interface.
`Errors`	Number of errors during Ethernet frame reception, including all kinds of receive errors.
`Vpi out of range`	Number of packets received on an out-of-range VPI.
`Vci out of range`	Number of packets received on an out-of-range VCI.
`Packets dropped`	Number of received packets dropped because of errors reported by the adapter.
`No connection`	Number of packets received for a disconnected or nonexistent virtual circuit.
`Small buffers`	Number of small received buffers supplied to the adapter.
`Large buffers`	Number of large received buffers supplied to the adapter.

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output when you use the command on an ATM interface.

Statistic	Meaning
`Packets`	Number of packets attempted to be transmitted.
`Bytes`	Number of bytes attempted to be transmitted.
`Errors`	Number of hardware errors encountered while attempting to transmit.
`Queue full`	Number of packets dropped because they could not be put in the transmit queue.
`List full`	Number of packets that could not be transmitted because the transmit queue was full.
`Iface unavailable`	Number of packets that could not be transmitted because the referenced interface was down.

For more information about the `ifstat` command, see the *na_ifstat*(1) man page.

Improving filer performance

About this section

This section 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.

The `wafl.maxdirsize` 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 4K 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*. 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 the *Getting Started Guide*.

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 the `no_atime_update` option On. Consider turning this option On 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.

Improving performance on directory lookups

Turning the `nfs.big_endianize_fileid` option On improves performance on directory lookups for clients that use the file ID in the file handle as a hash key in certain ways. Enable this option if your NFS clients are mainly running HP-UX or IRIX.

Note

If you turn the `big_endianize_fileid` option On, all NFS clients that have mounted directories from the filer must unmount and remount them; otherwise, they get "stale file handle" errors on all references to files already opened on the filer until they unmount and remount all directories.

Improving read-ahead performance

If the file access patterns of your clients are random (nonsequential), turning minimal read-ahead On might improve performance. By default, the filer uses aggressive read-ahead, which enhances sequential access, which is more commonly used by UNIX clients and applications. To specify minimal read-ahead, turn the `minra` option On. By default, the option is Off and the filer does very aggressive read-ahead.

Troubleshooting 16

Troubleshooting topics

Use the information in the following table to determine which troubleshooting topics to read.

Read...	To learn about...
"Getting technical assistance" on page 391	Getting technical assistance.
"Booting from system boot diskette" on page 392	Booting from diskette to recover from a lost password, configuration file errors, and some disk problems.
"Restarting a shut down filer" on page 395	Restarting a filer when it has shut down automatically.
"NVRAM problems" on page 396	How the filer handles NVRAM low battery conditions and inconsistent NVRAM contents.
"Volume problems" on page 398	Types of volume problems.
"Failed mounts and stale file handles" on page 399	How to fix mount and stale file handle problems caused by changing a volume's name.
"Volume name problems" on page 400	Using the rules for naming volumes, and about volume naming error messages.
"Disk problems" on page 401	Types of disk problems.
"Disk failure without a hot spare disk" on page 402	How the filer handles a disk failure when a hot spare disk is not available.
"Disk failure with a hot spare disk" on page 403	How the filer handles a disk failure when a hot spare disk is available.
"Disk errors" on page 404	Error messages the filer displays when it discovers certain disk problems.
"Inconsistent file system" on page 406	What to do if the filer displays a message that a file system is inconsistent.

Read...	To learn about...
"Disk operations in Maintenance mode" on page 407	Using maintenance-mode operations to correct disk problems.
"Configuration problems" on page 408	Booting the filer and recovering from errors in configuration files.
"How to reset the filer password" on page 410	Resetting the administration password for the filer.
"How to initialize all disks and create a new file system" on page 411	Initializing all the filer disks because you want to redeploy the filer or because Network Appliance Technical Support instructs you to do so.
"Network problems" on page 412	Diagnosing network problems.
"NFS problems" on page 414	Diagnosing NFS problems.
"Windows access problems" on page 416	Diagnosing problems with accessing the filer from Windows.
"UNIX cpio problems" on page 420	How 16-bit versions of the `cpio` command might cause problems when copying large files.
"UNIX df problems" on page 421	How the `df` command on some UNIX systems might report free space incorrectly.
"Cluster Messages" on page 422	What you can do to fix cluster problems.
"Serious error messages" on page 428	What to do if the filer halts with a serious error message.

Related information

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 start-up messages.

Getting technical assistance

Information to note before calling for support

If you encounter problems with the filer that you cannot solve, you might need to contact your service provider or Network Appliance Technical Support for assistance. If you do, have the following information available:

- the system serial number
- the system configuration as reported by the `sysconfig -v` command (if the filer still responds to the command)
- any diagnostic messages that were reported (Diagnostic messages are in the */etc/messages* file on the root volume.)

How to contact Network Appliance

The following table provides telephone and FAX numbers, and an email address that you can use to contact 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

Booting from system boot diskette

Boot from diskette to correct some types of problems

You might need to reboot the filer from the system boot diskette to correct configuration problems, recover from a lost password, or correct certain disk configuration problems.

Restrictions on booting from diskette

Do not boot the filer with a pre-5.0 system boot diskette. The attempt to boot will fail because pre-5.0 software cannot recognize system data from a filer running Data ONTAP 5.0 or later software.

If your file is using clustering, do not boot with a pre-5.2 system boot diskette. Releases earlier than 5.2 do not support clustering.

If you need to run an earlier release of Data ONTAP on the filer, contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391, for instructions about how to downgrade your filer.

Procedure for booting from diskette

Complete the steps in the following table to display the boot menu from a diskette. Booting from the diskette takes longer than booting from the hard disk.

Step	Action
1	Insert the diskette labeled System Boot Disk 1 into the filer's diskette drive. **Note** The Data ONTAP 5.2 system boot diskettes are specific to the model of filer that you are updating. You received diskettes appropriate to the model you are updating. If you have different models, be sure to check the diskette labels to make sure you're using the correct diskettes for the filer.
2	From the system console, enter the `reboot` command or, if the system is powered off, power it on. **Result:** The filer begins the boot process.

Step	Action
3	When the filer's LCD prompts you to, remove the diskette and insert the diskette labeled System Boot Disk 2 into the filer's diskette drive.

4	Press the Enter key on your console.
	
	Result: The filer boots and, if it can, displays the boot menu, shown below:
	
	`1) Normal Boot`
	`2) Boot without /etc/rc`
	`3) Change Password`
	`4) Initialize all disks`
	`5) Maintenance mode boot`
	
	`Selection (1-5)?`

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td>5</td><td colspan="2">Choose one of the boot types shown in the following table by entering the corresponding number.</td></tr>
<tr><td></td><td>Boot type</td><td>Description</td></tr>
<tr><td></td><td>Normal Boot (1)</td><td>Use Normal Boot to run the filer normally, but from a diskette.</td></tr>
<tr><td></td><td>Boot without /etc/rc (2)</td><td>Use Boot without /etc/rc to troubleshoot and repair configuration problems.
Note
Booting without /etc/rc causes the filer to use only default options settings, disregard all options settings you put in /etc/rc, and disable some services, such as <code>syslog</code>.</td></tr>
<tr><td></td><td>Maintenance mode boot (5)</td><td>Use Maintenance mode boot to go into Maintenance mode and perform some volume and disk operations and get detailed volume and disk information. Maintenance mode is special for the following reasons:
Most normal functions, including file-system operations, are disabled.
A limited set of commands is available for diagnosing and repairing disk and volume problems.
You exit Maintenance mode with the <code>halt</code> command.</td></tr>
</table>

Restarting a shut down filer

Procedure for restarting filer after unexpected shutdown

Complete the steps in the following table to restart your filer if it shuts down unexpectedly.

Step	Action	
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...**	**Then...**
	Showing the `mon>` prompt	Enter **`b`** (for boot) to reboot the filer.
	Showing the `ok` prompt	Enter **`boot`** to reboot the filer.
	Unresponsive	Reset the filer by turning it off, leaving it off for 30 seconds, and turning it back on.
	Results: The filer boots and displays the system prompt on the console.	
3	If the filer still does not boot, display the boot menu from diskette, as described in "Booting from system boot diskette" on page 392, then choose Normal Boot.	

NVRAM problems

Types of NVRAM problems

This section describes two types of NVRAM problems:

- NVRAM battery failure
- Inconsistent NVRAM contents

What F220, F330, and F540 filers do when NVRAM battery is low

This section applies only to the NetApp F220, NetApp F330, and NetApp F540 filers.

If the filer detects low batteries on the NVRAM card, it gives advance warning and automatically shuts down every 24 hours. You can change the time interval with the `raid.timeout` option to the `options` command.

If the filer is in degraded mode and the NVRAM batteries are low, it fails 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 turns off the NVRAM so that no battery power is used.

How the filer handles inconsistent NVRAM contents

The filer performs a number of checks to ensure that the NVRAM contents are consistent. If the contents are inconsistent, the filer performs one of following actions:

Inconsistency due to improperly updated volume: If the inconsistency is due to a failure to update a volume properly, the filer displays a message suggesting that you halt the filer with the `halt` command, take the offending volume off-line in Maintenance mode, and reboot the filer.

Inconsistency due to log updates for off-line volume: If the inconsistency is due to the log having updates for an off-line volume, the filer asks whether to discard them.

Inconsistency due to other reasons: If there are many inconsistencies that cannot be repaired, the filer discards the inconsistent contents and creates a core dump file. The file requests received during the last few seconds before the filer shuts down are 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 might be incorrect, the filer must do a parity check on the entire 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.

Volume problems

Types of volume problems described

This section describes the following types of volume problems:

- Failed mounts and stale file handles
- Volume name problems

Failed mounts and stale file handles

Changing volume names can cause mount and file handle problems

If mounts fail and clients see stale file handles, it might be because you renamed a volume and it is not exported anymore. This is because the new volume name was not in the */etc/exports* file on the root volume.

Procedure for fixing the /etc/exports problem

Complete the step in the following table to fix the problem.

Step	Action
1	Edit the */etc/exports* file on the filer default volume to change the old volume name to the new volume name. **Results:** The problem is resolved.

Volume name problems

Volume naming rules

A valid volume name has the following characteristics:

- The prefix is followed by either a letter or an underscore ("_").
- It contains only letters, digits, and underscores.
- It is not longer than 255 characters.

Examples of volume names

Examples of valid volume names are

- *_tech_pubs*
- *SW_Engineering*
- *Dept_32*

Error messages about volume names

You might get an error message that contains one of the following phrases:

- invalid volume name
- unrecognized volume name
- illegal volume name

If you get one of these messages, take one of the following actions:

- Type correctly the name of an existing volume.
- Type a valid volume name.

Disk problems

Types of disk problems described

This section describes how the filer reacts to a

- Disk failure without a hot spare disk
- Disk failure with a hot spare disk

Disk failure without a hot spare disk

About this section

This section describes how the filer reacts to a disk failure when a hot spare disk is available.

Filer runs in degraded mode

If a disk fails, the filer continues to run without losing any data but has a somewhat degraded performance.

Caution

Replace the disk as soon as possible, because a second disk failure could cause the filer to lose the entire file system.

Filer logs warning messages in /etc/messages

The filer logs a warning message in the */etc/messages* file on the root volume every hour after a disk fails.

Filer shuts down automatically after 24 hours

To ensure that you notice the failure, the filer automatically shuts itself off in 24 hours, by default, or in a period you set with the `raid.timeout` option to the `options` command. You can restart the filer without fixing the disk, but it continues to shut itself off periodically until you repair the problem.

Caution

Check the */etc/messages* file on the root volume once a day for important messages. You can automate checking this file with a script on a remote host that periodically searches the file and then alerts you.

Filer reconstructs data after disk is replaced

After you replace a disk, the filer detects the new disk when the system boots. The filer starts file service and reconstructs the missing data in the background with minimum interruption to service.

Disk failure with a hot spare disk

About this section

This section describes how the filer reacts to a disk failure when a hot spare disk is not available.

Filer replaces disk with spare and reconstructs data

If a disk fails, the filer

- replaces the failed disk with a hot spare disk
- reconstructs the missing data on the hot spare disk in the background, so that the interruption to file service is minimized
- logs the activity in the */etc/messages* file on the root volume

The filer does not shut down automatically.

Caution

After the filer is finished reconstructing data on the hot spare disk, replace the failed disk with a new hot spare disk as soon as possible so that there is always a hot spare disk available in the system. For information about replacing a disk, refer to "Disk management tasks" on page 88.

If a second disk fails and there is no hot spare disk available, contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391.

Related information

In addition to disk failure and hot spare disk replacement activity, the */etc/messages* file on the root volume logs any failure in a periodic check of the hot spare disk. For more information, refer to the *na_sysconfig*(1) and *na_messages*(5) man pages.

Disk errors

Types of disk errors described

The filer displays error messages when the following disk problems occur:

- A disk does not exist.
- A disk is in use.
- Disks are missing.

Error message: Nonexistent disks

If you get a message indicating that a disk does not exist, especially if you are adding a disk to a volume or a RAID group, make sure that

- The disk is specified correctly.
- The specified disk is a spare.

Error message: Disk in use

If you are adding a disk to a volume and you get a message indicating that a disk is in use, the disk you specified might already be a system disk. Make sure that

- The disk is specified correctly.
- The specified disk is a spare.

Error message: System cannot boot because disks are missing

You might get a message similar to the following:

```
The system cannot boot with more than one disk missing from a RAID
group.
```

This message indicates that a volume might be missing some disks because either not all the disks in a volume were transferred to a new filer or disks were damaged.

Make sure that all the disks in a volume were transferred. If the problem persists, it might mean that disks were damaged. Complete the steps in the following table to resolve the problem.

Step	Action
1	Display the boot menu from diskette, as described in "Booting from system boot diskette" on page 392, then choose Maintenance mode boot.
2	Use the `vol offline` command to take the volume off-line.
3	Reboot the filer; the volume specified in Step 2 is off-line.
4	Add missing disks, if possible, then bring them on-line; otherwise, follow these steps: 1. Replace any broken disks. 2. Destroy the old volume. 3. Create a new volume. 4. Use the `restore` command to restore the contents of the old volume from a backup tape.

Inconsistent file system

Inconsistencies seldom occur

The file system rarely becomes inconsistent. However, an inconsistent file system can be a result of combined disk and NVRAM failure.

Contact technical support if an inconsistency occurs

If your file system becomes inconsistent, contact Network Appliance Technical Support for assistance, as described in “Getting technical assistance” on page 391.

Disk operations in Maintenance mode

Maintenance mode operations

Maintenance mode enables you to perform the following operations to troubleshoot disk problems:

- Obtain detailed device information for each disk with the `disk_list` command.
- Check access to a particular disk with the `disk_check` command.
- Erase a disk label with the `disk_erase_label` command.

Note

For information about Maintenance mode, see "Booting from system boot diskette" on page 392.

Displaying detailed disk information

The `disk_list` command displays detailed device information for each disk on the system, such as drive type, and firmware revision level.

Checking access to a disk

The `disk_check` command checks access to a particular disk, issuing read requests to the disk. The disk is active for approximately 15 seconds. During this time, you can watch the disk's activity LED to verify that the disk is accessed.

Erasing a disk label

The `disk_erase_label` command erases a disk label on the specified disk drive.

Caution

Use the `disk_erase_label` command carefully because after a disk label is erased, RAID treats the disk as uninitialized and no longer recognizes it as a member of a RAID array.

Typically, use this command to remove an old RAID label on a previously used disk that you are adding 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.

Configuration problems

The /etc/rc, /etc/exports, and /etc/hosts files can contain errors

Configuration problems usually occur in one of the three configuration files on the root volume:

- */etc/rc*
- */etc/exports*
- */etc/hosts*

This section describes common configuration problems.

What to do when the filer is 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 on the root volume might have an IP address that is unreachable. Complete the steps in the following table to fix this problem.

Step	Action
1	Log in to the filer from the system console.
2	At the filer prompt, enter the following commands, replacing the information shown in italics with values appropriate for your filer: **`ifconfig`** ***`if`*** **`mediatype`** ***`type IP_address`*** **`netmask`** ***`netmask`*** **`exportfs -i -o root=`*`adminhost_IP_addr`* **`/`** **`nfs on`**
3	Mount the root file system from the administration host.
4	Edit the configuration files as described in Chapter 2, "Routine Filer Administration."

Filer runs setup when /etc/rc is damaged or missing

If the */etc/rc* file on the root volume is accidentally deleted, the filer automatically runs `setup` the next time it is booted.

If the system does not respond to network requests after a boot, 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 "Booting from system boot diskette" on page 392 for information about booting from the system boot diskette.

How to recover from configuration errors if NFS is the only licensed protocol

If you are running NFS only, complete the steps in the following table to recover from configuration errors in the */etc/rc* file.

Step	Action
1	Display the boot menu from diskette as described in "Booting from system boot diskette" on page 392.
2	Enter **2** to choose Boot without */etc/rc*. **Note** Booting the filer without the */etc/rc* file on the root volume automatically disables CIFS service. You cannot do this procedure using CIFS.

3	At the filer prompt, enter the following two commands, replacing the variables shown in italics with values appropriate for your filer:
	`ifconfig if mediatype type IP_address netmask netmask`
	`exportfs -i -o access=adminhost_IP_addr,root=adminhost_IP_addr nfs on`

4	Mount the root file system from the administration host.
5	Edit the configuration files, as described in the *Getting Started Guide*.
6	Remove the system boot diskette from the disk drive and reboot the filer to test the new configuration files.

Note

Although you can correct network problems using various keyboard commands, correct the */etc/rc* file on the root volume so that it initializes the system correctly if there is a power outage or system software failure.

How to reset the filer password

Reset the password if you forget it

If you forget your filer password, reset the password by using the system boot diskette. To avoid security problems, take care to limit access to the system boot diskette.

Procedure for resetting the password

Complete the steps in the following table to reset the filer password.

Step	Action
1	Reboot from diskette as described in “Booting from system boot diskette” on page 392.
2	When the boot menu appears, enter **3** to choose Change Password.

3	When the filer prompts you for a new password, enter it at the prompt.
	
	Results: The system prints the following message:
	
	`Password Changed`
	`Hit Return to reboot:`

4	Remove the diskette from the filer’s diskette drive and reboot the filer by pressing the Enter key.

How to initialize all disks and create a new file system

Initializing all disks erases all data

You might need to initialize all disks and create a single new file system in the following circumstances:

- You decide to redeploy an existing filer and need to completely reconfigure it.
- Network Appliance Technical Support advises you that the only way to recover from an error is to initialize all disks.

Caution

Initializing all disks causes all existing data to be lost.

Procedure for initializing all disks

Complete the steps in the following table to initialize all disks.

Step	Action	
1	**If the console is...**	**Then...**
	Displaying the filer prompt, for example, `toaster>`.	Place the system boot diskette into the diskette drive of the filer and enter **`reboot`**.
	Not displaying the filer prompt.	Reboot from diskette as described in "Booting from system boot diskette" on page 392.
2	When the boot menu appears, enter **4** to choose Initialize all disks. **Results:** The filer initializes all the disks and creates a single-volume file system.	

Network problems

Detect network problems using ping at the filer console

You can detect network problems by going to the filer console and using the `ping` command.

What the ping command does

The `ping` command checks whether the filer can communicate with other hosts on the network and that other hosts can communicate with the filer.

How to troubleshoot network problems

If the filer should be able to connect with a host but `ping` does not respond with a message indicating that the host is alive, complete the steps in the following table to troubleshoot the problem.

Step	Action
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 that the up and running flags are displayed.
3	Use the `arp -a` command to confirm that the filer has the correct IP-to-Ethernet address map for the host you are trying to reach.
4	Use the `netstat -r` command to examine the routing tables.

Step	Action
5	Use the `netstat -i` command to check for excessive errors on the interfaces. 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. **Note** 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.
6	Use the `routed status` command to determine 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 is 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.

Contact technical support about other network problem

If you encounter other problems, contact Network Appliance Technical Support for assistance, as described in "Getting technical assistance" on page 391.

NFS problems

Client's inability to mount directories indicates 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 volumes or directories from the filer.

How to troubleshoot NFS problems

Complete the steps in the following table to troubleshoot NFS problems.

Step	Action
1	Make sure that the filer is licensed for NFS by entering the `license` command at the filer prompt. If the following message appears `nfs not licensed` or if a message with protocols other than NFS appears but NFS is absent, you must get a license for NFS. **Note** For information about how to get a license, contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391.
2	Make sure that the filer can correctly look up the client host name.
3	Make sure that NFS service has been turned On using the `nfs on` command.
4	Make sure that the filer and the client are using correct IP addresses and names.

Step	Action
5	Sometimes a client can see the filer but gets a Permission Denied message when requesting a mount. If this happens, follow these steps: ◆ Make sure that you defined the file systems correctly in the filer's */etc/exports* file on the root volume and that you ran the `exportfs` command on the filer. ◆ 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, enter the following `options` command: **`options nfs.mount.rootonly off`** To make this change permanent, add the preceding command to your */etc/rc* file on the root volume.

Windows access problems

Kinds of access problems

This section describes preliminary troubleshooting steps, then describes how to troubleshoot the following problems:

- "Filer can't register with the Windows NT domain" on page 417
- "Incorrect password or unknown username" on page 418
- "Users can't map a drive" on page 419

Preliminary troubleshooting steps

Complete the steps in the following table to begin to troubleshoot Windows problems.

Step	Action
1	Make sure that the filer is licensed for CIFS by entering the `license` command at the filer prompt. If the following message appears `CIFS not licensed` or if a message with protocols other than CIFS appears but CIFS is absent, you must get a license for CIFS. **Note** For information about how to get a license, contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391.
2	If you are authenticating through a Windows NT domain, make sure that the filer has an account in the domain. **Note** You can verify that the filer is registered with a domain controller by using the `cifs testdc` command.
3	Make sure that CIFS service has been properly configured with the `cifs setup` command according to the instructions in the *Getting Started Guide*.

Filer can't register with the Windows NT domain

If you are using WINS, use the table in "Using Wins" to troubleshoot the problem; if you are not using WINS, use the table in "Not Using WINS" on page 418.

Using WINS: Use the following table if you are using WINS.

<table>
<tr><td colspan="5">Is the WINS server working?</td></tr>
<tr><td colspan="4">Yes</td><td>No</td></tr>
<tr><td colspan="4">Is the domain controller running?</td><td rowspan="8">Get the WINS server working.</td></tr>
<tr><td colspan="3">Yes</td><td>No</td></tr>
<tr><td colspan="3">Can you ping the domain controller?</td><td rowspan="6">Start the domain controller.</td></tr>
<tr><td colspan="2">Yes</td><td>No</td></tr>
<tr><td colspan="2">Can you ping the filer?</td><td rowspan="4">Check network connec-tivity</td></tr>
<tr><td>Yes</td><td>No</td></tr>
<tr><td>Contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391.</td><td>Check net-work con-nectivity</td></tr>
</table>

Not Using WINS: Use the following table if you are not using WINS.

<table>
<tr><th colspan="3">Is the PDC or BDC on the same subnet as the filer, and is the subnet connected to the first configured network interface card on the filer?</th></tr>
<tr><th colspan="2">Yes</th><th>No</th></tr>
<tr><td colspan="2">Is there a computer account created for the filer in the Windows NT domain?</td><td rowspan="2">Put the domain controller on the same subnet as the filer, and connect the subnet to the first configured network interface card on the filer.

Without a WINS server, the filer can talk only to the domain controller by broadcast. The filer broadcasts only from the first configured network interface. The domain controller must be on this subnet.</td></tr>
<tr><th>Yes</th><th>No</th></tr>
<tr><td>Contact Network Appliance Technical Support, as described in “Getting technical assistance” on page 391.</td><td>Create a computer account for the filer in the domain.</td></tr>
</table>

Incorrect password or unknown username

Use the following table to troubleshoot the problem.

<table>
<tr><th colspan="4">Are users in the same domain as the filer?</th></tr>
<tr><th colspan="2">Yes</th><th colspan="2">No</th></tr>
<tr><td colspan="2">Are you using a virtual LAN?</td><td colspan="2">Is there a trust relationship between the filer and the domain?</td></tr>
<tr><th>Yes</th><th>No</th><th>Yes</th><th>No</th></tr>
<tr><td>Have the users log in with DOMAINNAME \ USERNAME</td><td>Enter the users into the /etc/passwd file.</td><td>Have the users log in with DOMAINNAME \ USERNAME</td><td>Establish a trust relationship between the user’s account domain and the filer’s domain.</td></tr>
</table>

Users can't map a drive

If users get an Access Denied message, use the following decision table to troubleshoot the problem.

<table>
<tr><th colspan="3">Is the filer in the same domain as the PDC?</th></tr>
<tr><th colspan="2">Yes</th><th>No</th></tr>
<tr><th colspan="2">Does the user have access rights to the share?</th><td rowspan="3">Use Server Manager to add the filer to the domain or establish a trust relationship between the domains.</td></tr>
<tr><th>Yes</th><th>No</th></tr>
<tr><td>Contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391.</td><td>Give the user rights to the share.</td></tr>
</table>

UNIX cpio problems

The cpio version should support 32-bit inode definition numbers

If you copy large amounts of data using the UNIX `cpio` utility, 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 inode definition number. The problem generally occurs only on file systems with a large number of hard links.

Why the problem occurs: Some versions of `cpio` work by copying each file with hard links once and then re-creating 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. The `cpio` utility then overwrites the first file and creates files that no longer contain the original data.

Ask UNIX provider whether cpio version supports 32-bit inode definition numbers

Check with your UNIX provider to see whether 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.

UNIX df problems

The df version must support large file systems

Some UNIX versions of the `df` command have file system limits considerably smaller than the file system size supported by the filer. This can cause the UNIX `df` command to show an incorrect and useless 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.

Enable NFS option to avoid displaying useless data

To avoid a useless display of disk space on a client system that uses NFS version 2, enable the `nfs.v2.df_2gb_lim` option, as described in "Configuring filer options" on page 42. For additional information, refer to the *na_options*(1) man page.

DOS, Windows, and Macintosh clients might have display problem

Some DOS, Windows, or Macintosh clients might have a display problem similar to UNIX systems; in these cases, enable the `nfs.v2.df_2gb_lim` option.

Filer df command always shows correct disk space

At all times, the `df` command entered on the filer correctly shows the amount of disk space used and remaining.

Qtrees affect disk space displayed by df

If a directory in a qtree 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 qtree look fuller according to the client `df` command than it actually is.

Filer quota report command always displays correct usage

The filer `quota report` command shows the correct usage within that qtree.

Cluster Messages

Kinds of cluster messages

This section lists a console message, LCD messages, and system messages that indicate conditions that you can take steps to address.

What to do when you see a message that indicates trouble

If a filer or cluster ceases to function properly, save or record as many messages as you can. If the message is not listed here, contact Network Appliance Technical Support, as described in “Getting technical assistance” on page 391.

Where cluster LCD messages appear

LCD messages appear on two lines on the filer’s LCD. Cluster messages, if they appear at all, appear on the first line of the LCD; other messages appear on the second line.

Cluster LCD messages that indicate trouble are accompanied by an amber LED indicator.

Cluster LCD messages that you can address

The following messages pertaining to clusters can appear on the LCD of a takeover filer. These are messages that you can take steps to address

Message	Explanation	What you do
`cf disabled`	Takeover has been disabled.	Follow these steps: 1. Unless you want takeover to be disabled, issue the `cf enable` command. 2. Check the interconnect cables to make sure that they are seated properly. 3. Issue the `cf status` command to verify the link.

Message	Explanation	What you do
`Iconnect down!`	The interconnect link is broken.	Follow these steps: 1. Check the cluster interconnect cables to make sure that they are seated properly. 2. Issue the `cf status` command to verify the link.
`Iconnect error`	One of the interconnect links is broken.	Follow these steps: 1. Check the interconnect to make sure that it is functioning properly. 2. Issue the `cf status` command to verify the link.
`L: vol` *`volume n`*`% rebuilt`	The filer has taken over its partner and one of its own volumes is reconstructing itself.	Wait until the message disappears.
`P: vol` *`volume n`*`% rebuilt`	The filer has taken over its partner and one of its partner's volumes is reconstructing itself.	Wait until the message disappears.
`Partner down`	The partner is down and takeover is disabled.	If you want to take over the partner, issue the `cf enable` command, then the `cf takeover` command.
`Takeover`	The filer has taken over its partner.	Perform a giveback.
`Takeover in` *`nnn`*	The filer is counting down seconds until it takes over its partner.	Unless you want takeover to occur, issue the `cf disable` command.

The following message pertaining to a cluster can appear on the LCD of a failed filer.

Message	Explanation	What you do
`taken over`	The filer has been taken over.	Perform maintenance on the filer, then perform a giveback from the takeover filer.

Where syslog messages appear

Syslog messages appear by default in the */etc/messages* file. For additional information about syslog messages, see the Message logging section of Chapter 2 of the *System Administrator's Guide*.

Syslog messages that you can address

The following syslog messages are messages that you can take steps to address. By default they appear on the system console.

Message	Explanation	What you do
`cf.timed.enable: timed requires java which is not enabled on this system`	Essential Java files are missing from the filer.	Follow these steps: 1. Install the Java .zip files from the distribution CD-ROM onto the */etc/java* directory of your filers. 2. Reboot.
`Interconnect is down`	The cluster interconnect is not working properly. Takeover capability is disabled.	Follow these steps: 1. Check the cluster interconnect cables to make sure that they are seated properly. 2. If the connections are secure, and the message returns, check for Cluster Interconnect adapter messages in the */etc/messages* file, then contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391.

Message	Explanation	What you do
`No disks were detected on Fibre Channel port B; this filer will be unable to takeover correctly`	The B-port loop has not been connected to the local filer, which means that the local filer cannot take over its partner.	Follow these steps: 1. Check the B-port loop connections to the partner's disk shelves and make sure that they are securely connected. 2. Verify that the LRC cards are properly seated.
`Root volume` *`volume`*`, RAID group` *`n`* `is missing` *`n`* `disks and is unusable.`	A disk shelf or adapter failed, or the root volume's disks might be spread between A-port and B-port loops.	Check the disk shelves and make sure that they are wired properly.
`Volume` *`volume`* `is missing a data disk and has dirty parity.`	In this situation, the filer is refusing to perform a giveback because a volume is in degraded mode with dirty parity.	**Caution** You can lose data if you perform a giveback with a volume that is missing a data disk and has dirty parity. Follow these steps: 1. Wait until one if these events happen: ❖ Parity is clean. ❖ The data disk is reconstructed. 2. Verify that one of the events in Step 1 has happened using the `sysconfig -r` command. 3. Perform a giveback.
`Volume` *`volume`* `is missing` *`n`* `disks from RAID group` *`n`* `and is being taken offline for this boot.`	The volume's disks might be spread between A-port and B-port loops.	Follow these steps: 1. Verify that the disk shelves are wired properly. 2. Reboot the filer to clear this condition.

Message	Explanation	What you do
`WARNING cluster monitor fast timeout was blocked for` *n* `secs.`	The Data ONTAP kernel is supposed to schedule the cluster monitor's fast timeout thread at regular intervals to ensure that mailbox data is read and written within specific time frames. If a filer does not receive the mailbox data because the fast timeout thread is blocked, it might assume that its partner is down and attempt to perform a takeover.	If this message happens frequently, gather system utilization data (`sysstat`, profiling data, and so on), then contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391 so that they can characterize the system.
`X link is down`	The Cluster Interconnect has detected a failure in one of its two communication channels. This circumstance does not disable takeover but deserves investigation. **Note** This circumstance can also occur temporarily during the booting of a clustered filer.	Check the X (top) cable connections to make sure that the cables are seated properly.
`Y link is down`	The Cluster interconnect has detected a failure in one of its two communication channels. This circumstance does not disable takeover but deserves investigation. **Note** This circumstance can also occur temporarily during the booting of a cluster filer.	Check the Y (bottom) cable connections to make sure that the cables are seated properly.

About console messages

Some messages appear on the console but are not written to the */etc/messages* file. Most of them are self-explanatory.

Console message

The following table explains the console message "The cluster partner appears not to be operational."

Message	Explanation	What you do
`The cluster partner appears not to be operational. You will be asked whether you want to continue. If you answer "yes", the existing cluster monitor disk state will be overwritten and this node will be rebooted. Answering "no" will halt this node with no modification to the cluster monitor disk state.` `WARNING: Answering "yes" while the cluster partner is operational will have unexpected and potentially catastrophic results: YOUR FILESYSTEMS MAY BE DESTROYED`  `Do you wish to continue [y/n]?`	The local filer has been taken over but it appears its partner is not operational. This message is usually preceded by the following message: `This node was previously declared dead. Pausing to check cluster partner status ... partner not responding.` `Thu Aug 13 16:58:37 GMT [kbd_login_mgr]: Login from console`	Follow these steps: 1. Verify that the partner is operational or repair it. 2. Reboot it. 3. Perform a giveback from the partner. Giveback is the only method to ensure that data is consistent after a takeover. 4. If you cannot repair the partner, perform one of the following actions: ❖ Contact Network Appliance Technical Support, as described in "Getting technical assistance" on page 391. **Caution** If you enter **y** and override the takeover, client applications might lose data that was written before the reboot. ❖ Enter **y** to continue and override the takeover and allow the local filer to reboot.

Serious error messages

Panic messages mean serious problems

If your filer has a serious problem, such as a problem with the hardware or a bug in the system software, it issues a system panic message similar to the following one:

```
PANIC: system hung (NS0)!
Volume: volname
Version: verno
```

Message Component	Description
`system hung (NS0)`	Indicates the panic class of the message and is significant. The actual text of the message varies with circumstances.
volname	Is the name of the volume.
verno	Is the version number.

What to do after a panic message

Complete the steps in the following table when your system issues a panic message.

Step	Action
1	Write down the panic message.
2	Call Network Appliance Technical Support immediately, as described in “Getting technical assistance” on page 391.
3	Provide the panic message to Network Appliance Technical Support.

Glossary

ACL Access control list. A list that 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 through `telnet` or `rsh`.

appliance A device that performs a well-defined function and is easy to install and operate.

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 A filer daemon that triggers email messages from the customer site to Network Appliance or another specified email 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 A computer that shares 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 The technique for creating snapshots without consuming excess disk space.

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.
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. The bus architecture used in FAServers.
EISA Configuration Utility	A 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	The slots on the system board in which you insert expansion cards.
FDDI adapter	A Fiber Distributed Data Interface (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 The process of adding, removing, or replacing a disk while the filer is running.

hot swap adapter An expansion card 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.

interrupt switch A switch on some filer front panels used for debugging purposes.

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 The client host responsible for sending automatic email to Network Appliance when certain filer events occur.

Maintenance mode An option when booting a filer from a system boot 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 Asynchronous Transfer Mode (ATM) adapter.
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.
panic	A serious error condition causing the filer to halt. Similar to a software crash in the Windows system environment.
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. The bus architecture used in newer filer models.
pcnfsd	A 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 has been assigned as, the primary authentication server for the domain.
POST	Power-on self-tests. The tests run by the filer after the power is turned on.

PVC Permanent Virtual Circuit. A link with a static route defined in advance, usually by manual setup.

qtree A directory on which you can impose tree quotas, created by the `quota qtree` command. Formerly known as quota trees.

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 disk 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 An expansion card that supports the SCSI 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 An 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 enable users to restore files and enable you to back up the filer to tape while the filer is in use.

SVC Switched Virtual Circuit. A connection established through signaling. The user defines the endpoints when the call is initiated.

system board A printed circuit 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 WAFL™ file system was designed for the Network Appliance filer to optimize write performance.

WINS Windows Internet Name Service.

workgroup A collection of computers running Windows NT or Windows for Workgroups that is grouped for browsing and sharing.

Index

Symbols

/etc/crash directory 32, 37
/etc/dgateways file 107, 109
/etc/dumpdates file 319, 321, 324
/etc/exports file
    exporting directories in 32
/etc/fstab file
    balancing traffic among interfaces 387
    editing 387
/etc/hosts file 360
    accessing problems in 408
    contents of 101
    for IP address 31
/etc/hosts.equiv file 20, 360
/etc/httpd.access file 220
/etc/httpd.group file 221
/etc/httpd.hostprefixes file
    for virtual hosting 224
/etc/httpd.passwd file 221
/etc/messages file
    activity on the hot spare disk 82
    and files saved 41
    and messages from savecore 37
    and reporting to Network Appliance 391
    and warnings for disk failures 81
    configuring messages written to the 39–41
    for disk failures 402
/etc/netgroup file 145–146
/etc/nsswitch.conf file 27, 100
/etc/passwd file, copying for (PC)NFS 152
/etc/quotas file, configuring 296–299
/etc/rc file
    commands in 30–32
    damaged or missing 409
/etc/resolv.conf file 103
/etc/serialnum file 27
/etc/shadow file 27
    when using (PC)NFS 152
/etc/symlink.translations file 124
/etc/syslog.conf file 39–41
/etc/zoneinfo directory 32

Numerics

8.3 format for file names 128

A

access rights
    assigning rights to users 197–201
    CIFS shares 197–201
activity on the hot spare disk
    /etc/messages file 82
adding a foreign volume 92
adding disks 80
adding disks to a volume 91
administration host
    accessibility problems from 408
    modifying configuration files 26
    requirements for 15
ATM
    network mask for 31
    PVCs
        incoming on the filer 119
        outgoing, establishing on the filer 118
        tearing down 120
AuthName directive 220
automatic email to Network Appliance
    configuring 65–66
    contents of the messages 64
    designating the contact person 33
    using administration host for 14
autosupport daemon
    configuring with the options command 65–66
    how it works 63
autosupport options 46–47
autosupport.doit option 46, 66
autosupport.enable option 46, 65
autosupport.from option 46, 66
autosupport.mailhost option 46, 66
autosupport.noteto option 47
autosupport.to option 47, 66

B

benefits of multiple volumes 84
block size for the restore command 340
blocking factor
  for dump command 318, 326
  for ufsdump command 318
booting
  /etc/rc script 29–32
  and NVRAM 22
  from diskette 392–394
  with halt and reboot commands 22
  without /etc/rc file 393
broadcast filer address, setting using ifconfig 111

C

C$ share 19
cables, checking 412
cf disable command 247, 252
cf enable command 247, 252
cf forcegiveback command 252, 257
cf forcetakeover command 248, 252
cf giveback command 242, 252, 256
cf giveback -f command 256
cf partner command 245
cf status command 245
cf takeover command 239, 247, 252
cf.timed.enable option 246
CGI requests, redirecting 228
changing size of RAID groups 88
CIFS
  adding users 172–173
  assigning and changing access rights 197–201
  C$ share 19
  creating shares 183
  deleting a share 192
  displaying session information 210–211
  displaying share information 189–191
  displaying shares 180–182
  displaying statistics 209
  effect of filer reboot on clients 23
  file-naming conversions and NFS 130
  filer command-line only operations 165
  generic account
    creation by default 178
    disabling 179
    users 179
  guest access 178
  home directory shares, creating 194–196
  local groups, adding to filer 174–176
  open files, limits 169
  oplocks 207–208
  options 48–50
  remote shell use 177
  restoring files 337
  rsh use to enter filer commands 177
  session information, displaying 210–211
  sessions, limits 169
  shares
    changing description 190
    creating and changing 183–188
    deleting 192–193
    displaying 180–182
    displaying information 189–191
    limits 169
    renaming volume, effect on 167
  statistics, displaying 209
  symbolic links 123–125
cifs access command 197, 200–201
cifs access -delete command 201
CIFS guest access 178
CIFS guest account 178
cifs restart command 215
CIFS sessions
  starting 212–215
  stopping 212–215
cifs sessions command 210–211
cifs shares -add command 186
cifs shares -change command 187
cifs shares command 180, 181, 183, 190–191, 212
cifs shares -delete command 193
cifs stat command 209
cifs terminate command 213, 214
cifs.cifs.show_snapshot option 49
cifs.generic_account option 48
cifs.guest_account option 48
cifs.home_dir option 48, 195
cifs.netbios_aliases option 49
cifs.oplocks.enable option 49
cifs.scopeid option 49

cifs.symlinks.cycleguard option 50, 125
cifs.symlinks.enable option 50, 124
clients
  accessing snapshots from 278–280
  NFS statistics in the custom MIB 99
cluster
  commands that are different in partner mode 253–254
  commands unavailable in partner mode 252
  configuration information 245–246
  console messages 427
  defined 233
  disk reservations 239
  dump on virtual filer 254
  giveback
    performing 256–257
    understanding 242–243
  hardware components 235
  LCD messages 422–423
  management tasks 244
  messages
    console 427
    LCD 422–423
    syslog 424–426
  monitoring process 238
  normal mode 245–246
  partner mode 251–254
  partner name 245
  statistics in takeover mode 249–250
  status 245
  syslog messages 424–426
  takeover
    initiating 247–248
    managing 247
    process 239
    result 241
    understanding 239–240
  takeover mode 249–250
  terminology 236–237
  time synchronization 238, 246
  unlicensing 246
cluster interconnect 236
cluster monitor 236
collisions 413
commands
  aborted by Ctrl-C 21
  accessible through rsh 21
  unavailable in partner mode 252
configuration
  of volumes 84
  planning for multiple volumes 85
configuration files
  /etc/dgateways 109
  /etc/dgateways file 107
  /etc/hosts 101–103
  /etc/hosts.equiv file 20
  /etc/netgroup file 145
  /etc/rc 30–32
  /etc/resolv.conf file 103
  /etc/syslog.conf 39–41
  default 24–32
  editing 17
  errors in 409
configuration problems
  booting with diskette for 409
  filer accessibility 408
  lost passwords 410
  with /etc/rc file 408–409
console messages, cluster 427
console session 20
copying a volume
  aborting 367
  changing the speed 367
  checking status 366
  operation numbers 366, 368
  possible errors 364
  recommendation for 360
  requirements for 358
copy-on-write technique 260, 262
core files 37
cpio, copying files with 420
crash files 32
creating and changing shares 183–188
creating qtrees 290
creating volumes 91
cron and crontab 68–69

D

Data ONTAP

displaying version 370
initial volume configuration 83
data rebuild, on the hot spare disk 403
data reconstruction
speed 82
when filer is shut down (degraded mode) 82
date command 68
date date command 252
default
configuration 24–28
route in routing table 107
router 31, 107
degraded mode 81
meaning of 90
reasons for 81
timeout period for automatic shutdown 81
when a hot spare disk is available 82
when a hot spare disk is not available 81
deprecated MIB objects 98
destroying a volume 93
df command 269–271, 308–309, 421
diagnostic messages 39–41, 391
directories
created by snapshots 278–279
exporting 32
Directory directive 220
disk does not exist message 404
disk fail command 90
disk in use message 404
disk information, displaying 371, 373
disk remove command 90
disk reservations 239, 242
disk scrub command 78
disks
adding 80
addressing, use of 77
changing the size of RAID groups 88
concepts 76
degraded mode 81, 82, 90
different sizes in same filer 79
different types in RAID group 76
failures, effects of 81, 82
failures, handling 81
free space, accessing information through SNMP 99
free space, displaying 308–309
freeing space by deleting snapshots 276
hot swapping 79
information in the custom MIB 99
installing new 88
management tasks 88
maximum number of files 307
parity 76, 308
problems 401–406
quotas 70, 99, 272, 296–305
removing 89
replacing 80
restricting usage 296–306
right-sizing 80
SCSI ID number 77
setting size of RAID group 88
snapshots, space used 263–277
swapping 90
usable space 80
DNS
disabling 104
domain name in /etc/rc 32
enabling 103
options 51–52
querying the name server 102
resolving names with 100
dns.enable option 52
double disk failures 403
dump
/etc/dumpdates file 319, 321, 324
blocking factor 318, 320
default block size 326
levels 319
multiple subtrees 321, 324, 327
size of file 326
starting 327
subtrees 330
using filer tape drive from another system
SVR4 UNIX 350
UNIX (not SVR4) 351
using snapshot 260
verifying 318
volume size, maximum 320
dump command virtual filer 254

E

editing configuration files 17
effects of disk failures 81
email support 391
Enabling 103
errors
    caused by copying a volume 364
    caused by exceeding disk quotas 305
    displayed by netstat 413
Ethernet
    interface names and numbers 35
    network mask for 31
    setting media type on 111
explicit routes in routing table 107
exportfs command 32

F

failed filer
    defined 236
FAX support 391
FC-AL disk shelf 236
FDDI
    interface names and numbers 35
    network mask for 31
    setting the MTU 111–??, 112
file locking, differences between NFS and CIFS 122
file names
    conventions 130
    conversion 128
    legal characters 129
    maximum length 128
file names used by NFS and CIFS clients 129
file space, incorrect display 421
file system
    damaged 337
    inconsistent 406
    maximum number of files 307
    protection through RAID scrubbing 78
filer
    characteristics in a cluster 233
    description, changing and viewing 170–171
    halting without takeover 247
    restarting 395
    virtual 251–254
        accessing through a remote shell 252
        accessing through Telnet 252
        dump 254
        dump command 254
        restoring files 254
filer information, overall, displaying 372
filer system load
    systat command 82
FilerView 12
files
    copying with cpio 420
    large 96
    maximum size 96
    ownership
        changeable by root only 70
    same file criteria 261
    setting maximum number of 307
    working with large 96
firewall, virtual 223

G

generic account
    creation by default 178
    disabling 179
    users 179
giveback
    defined 236
    performing 256–257
    understanding 242–243
group quotas, creating 304
grouping Ethernet interfaces 116
guest
    access, CIFS 178
    access, NFS 178
guest account, CIFS 178

H

halt command 22, 252, 396
halt -f command 247
halting a filer without takeover 247
heartbeat
    defined 236
heartbeat transmission 238

home directory shares, CIFS, creating 194–196
host name resolution 100
host names
  for network interfaces 34
  resolving 101–105
hostname command 30
hot spare disk 88
  availability
    sysconfig command 82
  overview 79
  removing 89
  replacement activity (/etc/messages) 82
hot swapping a disk 79
hot swapping, overview of 79
hourly snapshots 266, 267
HTTP
  displaying connection information 231
  displaying statistics 232
  options 53–54
  password protection 220–221
  root directory 219
  starting service 218
  virtual hosting, enabling 224
httpd.admin.enable option 53
httpd.enable option 53
httpd.log.max_file_size option 53
httpd.rootdir option 53
httpd.timeout option 53
httpd.timewait.enable option 54
httpstat statistics in takeover mode 249–250

I

ICMP redirect messages 107
identifying disks
  SCSI ID 77
IERRS (input errors) displayed by netstat 413
ifconfig command 30, 31, 102, 111–115, 253, 412
ifconfig mtusize command 252
ifconfig -partner command 252
ifconfig partner command 252
ifstat command 249–250, 377
illegal volume name message 400
inconsistent file system 406
inodes, effects of maximum number of files on 307
installing new disks 88
interfaces
  errors on 413
  how packets are sent and received 109
  using ifconfig to configure 111–115
  using vif to link together 116
  See also ATM, Ethernet, FDDI
invalid volume name message 400
IP addresses
  setting in /etc/rc file 31
  setting using ifconfig 111
ip.path_mtu_discovery.enable option 61

L

large files 96
LCD messages, cluster 422–423
legal characters in file names 129
license command 71, 73, 252
licenses 71–73
limitations of multiple volumes 85
linking multiple interfaces 116
local filer 236
local groups, CIFS, adding to filer 174–176
localhost 360
logging in to the filer 20
logging out of the filer 20
lost data from disk failures 402
lost passwords 410
ls command, listing snapshot files 279, 280

M

mail host
  automatic email 14, 63
  designating other mail hosts 66
mailbox disk 237
Makefile, NIS 102, 146
making a volume inactive 92
management tasks
  for disks 88
  for volumes 91
maxfiles command 307
maximum number of files 307
media type for an Ethernet interface 111
messages

configuring the syslogd daemon 39–41
console, cluster 427
file of 39–41
for disk failures 402
LCD, cluster 422–423
severity of 39
syslog, cluster 424–426
MIB objects
deprecated 98
multivolume, locations 98
MIB, Network Appliance custom 98
MIB-II 97
MIME Content-Type, specifying 226
minra option 388
minra volume option 59
modes
normal 237, 245–246
partner 251–254
takeover 249–250
monitoring status
of volumes 91
mount_rootonly option 70
mounting files
if there are qtrees 309
problems with 414
restricting mount privilege to root 70
mounting volumes 84
mt command 253, 352
multiple RAID groups 76
multiple volumes
benefits of 84
configuration planning 85
limitations of 85
multivolume MIB objects, locations 98

N

name services, specifying the order in which contacted 100
names
resolving 100–106
volume naming conventions 83
NDMP 333–336
using to backup filer 316
ndmp command 334
netstat
statistics different in takeover mode 249–250
netstat command 107, 110, 376, 413
netstat -i command 253
network interfaces
balancing traffic among 387
configuring 111–115
naming conventions 34
network statistics, displaying 376
networks
connections, checking 412
how filer sends and receives traffic 109
management services, using SNMP 97–99
network mask
configuring using ifconfig 111
in /etc/rc file 31
problems with 412
statistics 412
using ifconfig to configure 111
NFS
file-naming conversions and CIFS 130
how interfaces respond to packets 109
problems with 414
statistics in custom MIB 98
statistics, displaying (nfsstat command) 160
turning on 32
NFS and (PC)NFS options 55–56
nfs command 32
NFS guest access 178
NFS over UDP requests 109
nfs.big_endianize_fileid option 388
nfs.mount_rootonly option 55
nfs.per_client_stats.enable option 55
nfs.tcp.enable option 55
nfs.v2.df.2gb.lim option 55
nfs.v2.df_2gb_lim option 55, 421
nfs.v3.enable option 56
nfs.webnfs.enable option 56
nfs.webnfs.rootdir option 56
nfs.webnfs.rootdirset option 56
nfsstat command 160–163, 249–250
nightly snapshots 265–267
NIS
changing domain name 106
disabling 106

domain name
set in /etc/rc 32
specifying with option 57
enabled in /etc/rc 32
enabling
during setup 105
without using setup 105
maps supported 105
options 57
propagating changes
/etc/hosts on filer 102
/etc/netgroup on filer 146
/etc/passwd on filer 153
/etc/shadow for (PC)NFS on filer 153
nis.domainname option 57
nis.enable option 57
no_atime_update option 387
no_atime_update volume option 59
nonexistent disks 404
Nonvolatile RAM (NVRAM)
batteries 82, 396
failures in 391
inconsistent contents 396
logging requests 22
mirroring, cluster 237, 238
when to increase 374
normal mode 237
normal mode, cluster 245–246
nosnap volume option 59
nosnapdir option 264, 265
nosnapdir volume option 59
numbering network interfaces 34

O

oplocks 207–208
options
autosupport.doit 46, 66
autosupport.enable 46, 65
autosupport.from 46, 66
autosupport.mailhost 46, 66
autosupport.noteto 47
autosupport.to 47, 66
cf.timed.enable 246
cifs.generic_account 48
cifs.guest_account 48
cifs.home_dir 48, 195
cifs.netbios_aliases 49
cifs.oplocks.enable 49, 208
cifs.scopeid 49
cifs.show_snapshot 49
cifs.symlinks.cycleguard 50, 125
cifs.symlinks.enable 50, 124
dns.domainname 52
dns.enable 52
httpd.admin.enable 53
httpd.enable 53, 218
httpd.log.max_file_size 53, 218
httpd.rootdir 53, 218, 224
httpd.timeout 53
httpd.timewait.enable 54
ip.path_mtu_discovery 61
minra 388
mount_rootonly 70
nfs.big_endianize_fileid 388
nfs.mount_rootonly 55
nfs.per_client_stats.enable 55
nfs.tcp.enable 55
nfs.v2.df_2gb_lim 55, 421
nfs.v3.enable 56
nfs.webnfs.enable 56
nfs.webnfs.rootdir 56
nfs.webnfs.rootdirset 56
nis.domainname 57
nis.enable 57
no_atime_update 387
nosnapdir 264, 265
pcnfsd.enable 56
pcnfsd.umask 56
raid.reconstruct_speed 58, 82
raid.scrub.enable 58, 78
raidtimeout 58, 81
root_only_chown 70
rps.status 61
telnet.hosts 61, 70
vol.copy.throttle 368
volume options
minra 59
no_atime_update 59
nosnap 59

nosnapdir 59
raidsize 60
root 60
vol.copy.throttle 62
wafl.maxdirsize 62, 387
wafl.root_only_chown 62
options command
making changes permanent 43
using 42
using options 1

P

packets, responses to 109
panic messages 428
parity disks 76
role 77
partner 237
partner command 251
partner mode 237, 251–254
commands that are different 253–254
unavailable comands 252
passwd command 21
password
HTTP pages 220–221
passwords
changing with passwd command 21
lost 410
used for shell session 20
(PC)NFS 149–153
pcnfsd daemon 151–152
pcnfsd.enable option 56
pcnfsd.umask option 56
performance, improving 374
permissions
for snapshots 261
on exported directories 138
ping command 412
protocol licenses 71–73
PVCs, ATM
establishing on a remote ATM host 119, 120
establishing on interconnecting ATM switches 119
incoming, establishing on the filer 119
outgoing, establishing on the filer 118

Q

qtrees
creating 290
displaying information about 294
moving files in and out of 284
oplocks settings 292
parameters 281
security style 285–287
use for backups 283
use in projects 283
what they are 281
quota
command 300
disk, setting up 296–305
displaying report 303
effects of snapshots on 272
effects on clients when exceeded 305
resizing 302
quotas, disk
information available through SNMP 99
information available through the custom MIB 99
when changing file ownership 70

R

RAID (Redundant Array of Independent Disks)
accessing information through SNMP 99
adding disks 80
data reconstruction speed 82
disk scrubbing 78
displaying information about disks 371, 373
displaying information for each group 373
displaying overall information 371
group size characteristics 76
groups 76, 83
information in the custom MIB 99
options 58
spare disk use 76
support for multiple groups 76
RAID information, overall, displaying 371, 373
raid.reconstruct_speed option 58, 82
raid.scrub.enable option 58, 78
raidsize volume option 60
raidtimeout command 81

raidtimeout option 58, 81
rdate command 68, 252
reactivating an off-line volume 92
read-ahead, minimal 388
read-only bit 126–127
reboot command 22, 252
rebooting
  and NVRAM 22
  automatic email to Network Appliance 38
  from diskette 392–407
  scripts for 30
  with halt and reboot commands 22
remote shell (rsh)
  accessing the filer from the administration host 20
  accessing virtual filer with 252
  for entering filer commands 21
  use with CIFS commands 177
removing
  a hot spare disk 89
  volumes from a filer 85
renamed volume not exported 399
renaming volumes 94
replacing
  disks 80
  failed disks 80
requests, replies to 109
require group directive 221
require user directive 221
reservations
  disk 239, 242
resolving host names 101
restarting a filer 395
restore command 312, 337–345
restore_symboltable file 343
restoring files from virtual filer 254
restoring files, remote access to a filer's tape drive
  SVR4 UNIX 350
  UNIX (not SVR4) 351
restoring the entire filer 343
right-sizing disks 80
role of parity disks 77
root volume 16, 83
  option 60
root_only_chown option 70
route command 107
routed command 108
routed daemon
  /etc/dgateways file 109
  purpose of 108
router
  default 31, 107
  problems with 413
Routing Information Protocol (RIP) 108
routing table, filer 107
rps.status option 61
rsh. See remote shell (rsh)

S

savecore command 32, 37
SCSI ID, identifying disks 77
security
  options for 70
  system password 21
serious error message 428
setting up the filer 24
shadow file 27
shares
  creating and changing 183–188
  deleting 192–193
  displaying 180–182
  displaying information about 189–191
  renaming volume, effect on 167
shelfchk command in partner mode 253
shut down filer, restarting 395
size of dump file 326
snap command 264–268
snapshot_for_dump snapshot 330
snapshots
  accessing 278–279
  automatic 265–267
  commands for 264
  definition 260
  deleting to free space 276
  directory name displayed on CIFS clients 279
  disk consumption by 269–277
  effects on disk quotas 272
  information in the custom MIB 99
  ls command 279, 280

magic directories 278
making snapshot directory invisible 264
operation of 261–262
options 58–60
reserving space for 270–271, 274–275, 276
scheduling 273
snapshot_for_backup file 330
types 265
SNMP
commands
examples of 97–98
in /etc/rc 32
configuring the agent 97
custom MIB 98
takeover mode 250
snmp command 97–98
software licenses 71–73
spare disks in RAID groups 76
sticky bit 28
subnets, exporting to 147–148
svtx bit 28
swapping out disks 90
symbolic links, CIFS 123–125
sysconfig command 245–246, 371–372
different in partner mode 253
hot spare disk availability 82
usable disk space 80
sysconfig -d command 371
sysconfig -r command 371
sysconfig -t command 371
sysconfig -v command 391
syslog messages, cluster 424–426
syslogd daemon 39
sysstat command 374
for filer system load 82
statistics in takeover mode 249–250
system crashes 32
system message files 39–41
system panics 37, 428

T

takeover
defined 237
disabling 247
enabling 247
initiating 247–248
managing 247
process 239
result 241
understanding 239–240
takeover filer 236
takeover mode 237, 249–250
prompt 249
statistics different 249–250
tape drives
controlling 352–353
displaying information about 371
inaccessible on virtual filer 252
remote access to filer 313, 350
TCP/IP, how interfaces respond to packets 110
technical assistance, getting 391
Technical Support
email address 63
telnet connection to the filer
administration host 15
limiting host access 61, 70
virtual filer 252
telnet.hosts option 61, 70
terminology, cluster 236–237
time
setting on the filer 68
synchronization, cluster 238, 246
synchronizing the filer with another system 68
time zone, setting 32
timezone command 32
timezone name command 252
troubleshooting
configuration problems 408–410
cpio problems 420
df problems 421
disk problems 401–403
network problems 412
NFS problems 414
NVRAM problems 396–397
UNIX cpio problems 420
UNIX df problems 421
volume problems 398–405
trunking Ethernet interfaces 116
trusted host 21, 360

U

unlicensing cluster feature 246
unrecognized volume name message 400
uptime command 253, 375
URL, how filer translates 228
usable space on disks 80
use of hot spare disks 79
user quotas, creating 303
users, CIFS, adding to filer 172–173

V

valid volume names 400
version command 370
version of Data ONTAP, displaying 370
vif command 116
    using to create a virtual interface 116
    using to display statistics about a virtual interface 117
    using to eliminate a virtual interface 117
virtual filer 251–254
    accessing through a remote shell 252
    accessing through Telnet 252
    defined 237
    dump 254
    dump command 254
    restoring files 254
    unavailable commands 252
virtual firewall 223
virtual host addresses, mapping 225
virtual hosting
    enabling 224
    setting up 224
virtual interface 116
    creating 116
    displaying statistics about 117
    eliminating 117
    listing names of 116
    primary interface of 117
vol command
    add 91
    create 88, 91
    destroy 93
    offline 92
    online 92
    options 92
    options raidsize 88
    rename 94
    status 91
vol copy start command 361–365
vol copy throttle command 368
vol options command 42
vol status command 245–246, 254, 373
vol status -d command 373
vol status -r command 373
vol status -v command 373
vol.copy.throttle option 62, 368
volcopy license command 72
volume
    aborting the copying of 367
    changing the speed of copying 367
    checking status of copying 366
    copy operation numbers 366, 368
    copying 355–368
    possible errors when copying 364
    recommendation for copying 360
    requirements for copying 358
volume information, displaying 373
volume name not valid message 400
volume names 16
volume not exported 399
volume options
    minra 59
    nosnap 59
    nosnapdir 59
    raidsize 60
    root 60
    vol.copy.throttle 62
volume state information, displaying 373
volumes
    adding disks 91
    adding foreign volume 92
    and management tasks 91
    concepts 83
    configuration of 84
    creating 91
    destroying 93
    handling failures 94
    making inactive 92
    monitoring status 91

mounting 84
naming conventions 83
problems with 398–405
reactivating 92
removing 85
renaming 94
renaming and effect on shares 167
setting options 92

W

wafl.maxdirsize option 62, 387
wafl.root_only_chown option 62
warnings for disk failures 81
WebNFS 156–158
weekly snapshots 265
working with large files 96

Y

ypwhich command 106