

Data ONTAP® 7.1

MultiStore® Management Guide

Network Appliance, Inc.
495 East Java Drive
Sunnyvale, CA 94089 USA
Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 4-NETAPP
Documentation comments: doccomments@netapp.com
Information Web: <http://www.netapp.com>

Part number 210-01078_B0
December 2005

Copyright and trademark information

Copyright information

Copyright © 1994–2005 Network Appliance, Inc. All rights reserved. Printed in the U.S.A.

No part of this document 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 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 and Carnegie Mellon University is 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.

This software contains materials from third parties licensed to Network Appliance Inc. which is sublicensed, and not sold, and title to such material is not passed to the end user. All rights reserved by the licensors. You shall not sublicense or permit timesharing, rental, facility management or service bureau usage of the Software.

Portions developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright © 1999 The Apache Software Foundation.

Portions Copyright © 1995–1998, Jean-loup Gailly and Mark Adler

Portions Copyright © 2001, Sitraka Inc.

Portions Copyright © 2001, iAnywhere Solutions

Portions Copyright © 2001, i-net software GmbH

Portions Copyright © 1995 University of Southern California. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of Southern California, Information Sciences Institute. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

Portions of this product are derived from version 2.4.11 of the libxml2 library, which is copyrighted by the World Wide Web Consortium.

Network Appliance modified the libxml2 software on December 6, 2001, to enable it to compile cleanly on Windows, Solaris, and Linux. The changes have been sent to the maintainers of libxml2. The unmodified libxml2 software can be downloaded from <http://www.xmlsoft.org/>.

Copyright © 1994–2002 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>

Software derived from copyrighted material of the World Wide Web Consortium is subject to the following license and disclaimer:

Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be used within the body of any redistributed or derivative code: "Copyright © [\$date-of-software] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>.

Notice of any changes or modifications to the W3C files, including the date changes were made.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS

FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

Software derived from copyrighted material of Network Appliance, Inc. is subject to the following license and disclaimer:

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 or purchase of this product does 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

NetApp, the Network Appliance logo, the bolt design, NetApp—the Network Appliance Company, DataFabric, Data ONTAP, FAServer, FilerView, MultiStore, NearStore, NetCache, SecureShare, SnapManager, SnapMirror, SnapMover, SnapRestore, SnapVault, SyncMirror, and WAFL are registered trademarks of Network Appliance, Inc. in the United States, and/or other countries. gFiler, Network Appliance, SnapCopy, Snapshot, and The Evolution of Storage are trademarks of Network Appliance, Inc. in the United States and/or other countries and registered trademarks in some other countries. ApplianceWatch, BareMetal, Camera-to-Viewer, ComplianceClock, ComplianceJournal, ContentDirector, ContentFabric, EdgeFiler, FlexClone, FlexVol, FPolicy, HyperSAN, InfoFabric, LockVault, Manage ONTAP, NOW, NOW NetApp on the Web, ONTAPI, RAID-DP, RoboCache, RoboFiler, SecureAdmin, Serving Data by Design, SharedStorage, Simulate ONTAP, Smart SAN, SnapCache, SnapDirector, SnapDrive, SnapFilter, SnapLock, SnapMigrator, SnapSuite, SnapValidator, SohoFiler, vFiler, VFM, Virtual File Manager, VPolicy, and Web Filer are trademarks of Network Appliance, Inc. in the United States and other countries. NetApp Availability Assurance and NetApp ProTech Expert are service marks of Network Appliance, Inc. in the United States. Spinnaker Networks, the Spinnaker Networks logo, SpinAccess, SpinCluster, SpinFS, SpinHA, SpinMove, and SpinServer are registered trademarks of Spinnaker Networks, LLC in the United States and/or other countries. SpinAV, SpinManager, SpinMirror, SpinRestore, SpinShot, and SpinStor are trademarks of Spinnaker Networks, LLC in the United States and/or other countries.

Apple is a registered trademark and QuickTime is a trademark of Apple Computer, Inc. in the United States and/or other countries. Microsoft is a registered trademark and Windows Media is a trademark of Microsoft Corporation in the United States and/or other countries. RealAudio, RealNetworks, RealPlayer, RealSystem, RealText, and RealVideo are registered trademarks and RealMedia, RealProxy, and SureStream are trademarks of RealNetworks, Inc. in the United States and/or other countries.

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

Network Appliance is a licensee of the CompactFlash and CF Logo trademarks.

Network Appliance NetCache is certified RealSystem compatible.

Table of Contents

	Preface	.xi
Chapter 1	About the MultiStore Software Product.	1
	Understanding MultiStore	2
	System administration for a storage system running MultiStore.	7
Chapter 2	Managing MultiStore	9
	Establishing a vFiler unit	10
	Enabling or disabling the MultiStore license	11
	Planning for vFiler unit creation	13
	Creating a vFiler unit.	16
	Understanding the vFiler command family	21
	Setting up a vFiler unit manually	22
	Administering vFiler unit resources from the hosting storage system	24
	Changing resources for a vFiler unit	25
	Adjusting the vFiler unit limit	29
	Renaming a vFiler unit	32
	Stopping and starting a vFiler unit	33
	Destroying a vFiler unit.	35
	Re-creating a destroyed vFiler unit.	37
	Allowing or disallowing protocols for a vFiler unit	38
	Displaying vFiler unit status	41
	AutoSupport and syslogd messages about a vFiler unit	42
	Executing commands and setting options for a vFiler unit.	43
	Effects of storage system reboot on a vFiler unit.	48
	Understanding volumes and qtrees on a vFiler unit	49
	Backing up vFiler units.	51
	Understanding LUNs on a vFiler unit	52
	Networking considerations	56
	Understanding SnapMirror on the hosting storage system	57

	Understanding SNMP when used with a vFiler unit	58
	Performance monitoring and statistics	59
Chapter 3	Using vFiler Units in Different IPspaces	61
	Understanding IPspaces	62
	Advantages of using VLAN tagging for IPspaces	66
	Requirements for using clusters and IPspaces	67
	Case studies for IPspaces	69
	Managing IPspaces on a storage system	70
	Creating an IPspace	71
	Listing IPspaces on a storage system	72
	Assigning an interface to an IPspace.	73
	Destroying an IPspace	75
	Creating a vFiler unit in a nondefault IPspace	76
Chapter 4	File Access Using NFS and CIFS	79
	Accessing a vFiler unit's file system with CIFS and NFS	80
	How to specify path names for NFS exports or CIFS shares.	81
	Preparing the vFiler unit for NFS.	82
	Preparing the vFiler unit for CIFS	83
Chapter 5	Disk Space Management Using Quotas	85
	Understanding quotas.	86
	Turning quotas on and off	87
	How to resize quotas	90
	Understanding the /etc/quotas file	91
	Displaying quota status	92
	Understanding quota reports	93
	Messages from exceeded quota thresholds and soft quotas	95
Chapter 6	Disaster Recovery and Data Migration	97
	Understanding disaster recovery and data migration.	98

	Preparing the destination storage system	99
	Storage and network checklists	107
	How to create, activate, or delete a disaster-recovery vFiler unit	110
	Creating the disaster-recovery vFiler unit	112
	Activating the disaster-recovery vFiler unit	116
	Deleting the disaster-recovery vFiler unit	118
	Reactivating the original vFiler unit	119
	Resynchronizing the vFiler unit	120
	Reactivating the original vFiler unit using SnapMirror commands	126
	Reactivating the original vFiler unit using vFiler dr commands	129
	Recreating the vFiler unit on a replacement storage system	132
	How to move (migrate) a vFiler unit	133
	Migrating the vFiler unit by copying data	134
	Migrating a vFiler unit between clustered nodes with SnapMover	140
Chapter 7	Virus Protection for CIFS	149
	Understanding virus scanning for a vFiler unit.	150
	Configuring virus scanning for a vFiler unit	152
Appendix A	Building secure networks on a storage system	153
	Index	155

Preface

About this guide

This guide describes how to use the MultiStore® software product on storage systems that run Data ONTAP® 7.1 software.

Audience

This guide is for system administrators who are familiar with operating systems that run on the Data ONTAP storage system clients, such as UNIX®, Windows 95™, Windows NT®, and Windows® 2000. It also assumes that you are familiar with how to configure these storage systems and how the NFS, CIFS, and HTTP protocols are used for file sharing or transfers. This guide does not describe basic system or network administration topics, such as IP addressing, routing, and network topology; it emphasizes the characteristics of the Data ONTAP software.

Terminology

Network Appliance's storage products (filers, FAS appliances, and NearStore® systems) are all storage systems—also sometimes called *filers* or *storage appliances*.

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.

Command conventions

You can enter Data ONTAP commands on either the system console or from any client computer that can access the storage system through a Telnet session.

In examples that illustrate commands executed on a UNIX workstation, the command syntax and output might differ, depending on your version of UNIX.

FilerView as an alternative to commands

Tasks you perform as a storage system administrator can be performed by entering commands at the system console, in configuration files, or through a Telnet session or a Remote Shell (rsh) connection.

Another method of performing many common storage system tasks is to use the FilerView® graphical management interface for viewing and managing a storage system from a Web browser. FilerView comes with every Data ONTAP storage system, is easy to use, and includes Help that explains Data ONTAP features and how to work with them in FilerView.

For more information about accessing a storage system with FilerView, and about FilerView Help, see the *System Administration Guide*.

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
<i>Italic font</i>	Words or characters that require special attention. Placeholders for information you must supply. For example, if the guide says to enter the <code>arp -d hostname</code> command, you enter the characters “arp -d” followed by the actual name of the host. Book titles in cross-references.
Monospaced font	Command and daemon names. Information displayed on the system console or other computer monitors. The contents of files.
Bold monospaced font	Words or characters you type. What you type is always shown in lowercase letters, unless 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.

Attention

An attention contains instructions that you must follow to avoid damage to the equipment, a system crash, or loss of data.

About this chapter This chapter introduces you to the MultiStore software product.

Topics in this chapter This chapter discusses the following topics:

- ◆ “[Understanding MultiStore](#)” on page 2
- ◆ “[System administration for a storage system running MultiStore](#)” on page 7

Understanding MultiStore

What MultiStore is

MultiStore is an optional software product that enables you to partition the storage and network resources of a single storage system so that it appears as multiple storage systems on the network. Each “storage system” created as a result of the partitioning is called a vFiler™ unit. A vFiler unit, using the resources assigned, delivers file services to its clients as a storage system does.

The storage resource assigned to a vFiler unit can be one or more qtrees or volumes. The network resource assigned can be one or more base IP addresses or IP aliases associated with network interfaces. You can add or remove resources at any time.

Licensing MultiStore

Because MultiStore is an optional software product, you must enable the software license for it. The name of the license is *MultiStore*.

Reasons for using MultiStore

MultiStore is typically used for the following reasons:

- ◆ You want to consolidate multiple servers to one storage system.
- ◆ You use the storage system to host data for multiple customers, such as clients of a service provider or different organizations within an enterprise.
- ◆ You use the SnapMirror® technology to migrate data from one storage system to another transparently.

MultiStore for server consolidation: The following scenarios illustrate how you can use MultiStore to consolidate servers:

- ◆ Suppose you currently manage multiple servers and you want to store all data on one storage system for easier administration.

To consolidate the servers, you can partition the storage system into vFiler units and then copy the data from the servers to the vFiler units. If the vFiler units are for Common Internet File System (CIFS) users, you can set up the vFiler units to use the same computer names as the servers. This enables CIFS clients to share resources without having to remap their drives or search for the new server in Network Neighborhood. If the vFiler units are for Network File System (NFS) users, the NFS clients might need to remount the file systems, or they can use the automounter to automatically mount the file systems from the new locations.

- ◆ A storage system without MultiStore can participate in only one security domain, so if your environment requires that different groups of CIFS users be in different domains, you must use multiple storage systems. MultiStore, however, enables you to install each vFiler unit in the appropriate domain while keeping all the data on the same physical storage system.
- ◆ Because you can set up Network Information Service (NIS) and Domain Name Service (DNS) servers for individual vFiler units, after you consolidate the servers on one storage system, network clients of the vFiler units can continue to use the same NIS and DNS servers as before.

MultiStore for service providers and enterprises: Service providers, such as Internet service providers (ISPs) and storage service providers (SSPs), can partition the resources of a storage system to create many vFiler units for client companies. Similarly, the information technology (IT) department of an enterprise can create vFiler units for various organizations, or customers, within the enterprise.

The administrator for each customer can manage and view files only on the assigned vFiler unit, not on other vFiler units that reside on the same storage system. In addition, there is no data flow between vFiler units. A customer using a vFiler unit can be assured that no sensitive information is exposed to other customers that store data on the same storage system.

For example, an SSP can create the following vFiler units on a storage system:

- ◆ A vFiler unit named vFilerA. It uses the /vol/vol1 volume and the e0 interface on the storage system. It is leased to CompanyA.
- ◆ A vFiler unit named vFilerB. It uses the /vol/vol2 volume and the e1 interface on the storage system. It is leased to CompanyB.

Although both CompanyA and CompanyB store data on the same storage system, network traffic for each company is restricted to the specified interface. The administrator at CompanyA, which uses NFS to access data, cannot use the `showmount` command on a UNIX client to view directories on the storage system that are outside the /vol/vol1 volume. Similarly, the administrator at CompanyB, which uses CIFS to access data, cannot browse the shares that are outside the /vol/vol2 volume.

See Chapter 3, “[Using vFiler Units in Different IPspaces](#),” on page 61 for a more detailed discussion.

MultiStore for data migration: MultiStore enables you to migrate data from one storage system to another without extensive reconfiguration on the destination storage system.

Without MultiStore, you can use the SnapMirror technology to copy data from one storage system to another, but you then need to edit the files' access control lists (ACLs), local user group definitions, user mapping information, and so on, before users can access the data. If the data being copied is stored on a vFiler unit, however, all user, group, and ACL information is encapsulated in the vFiler unit. You re-create a vFiler unit on the destination storage system with the encapsulated information, and data can be served from the destination storage system without further reconfiguration. UNIX clients do not experience any disruption in service when the vFiler unit on the destination storage system starts serving data instead of the vFiler unit on the source storage system.

About the hosting storage system

The storage system on which you create vFiler units is called the hosting storage system. The storage and network resources used by the vFiler units exist on the hosting storage system.

About the default vFiler unit

When you license MultiStore, Data ONTAP automatically creates a default vFiler unit on the hosting storage system. The name of the default vFiler unit is vFiler0.

Initially, vFiler0 owns all the resources on the storage system. After you create additional vFiler units, vFiler0 owns the resources that are not owned by the vFiler units you create.

The default vFiler unit exists as long as the MultiStore license is enabled. On a storage system with the MultiStore license enabled, you cannot destroy vFiler0.

All information provided in this guide about the vFiler units is applicable to vFiler0, unless noted otherwise.

Some limitations of MultiStore

MultiStore has the following limitations:

- ◆ You can have a maximum of 33 vFiler units on a storage system. However, the maximum limit does depend on the memory capacity of the hosting storage system.

You can create 32 vFiler units on a storage system. The 33rd vFiler unit is vFiler0, which is created automatically when MultiStore is licensed on the storage system. vFiler0 continues to exist as long as MultiStore is licensed on the storage system.

In a cluster, you can create up to 32 vFiler units on each node of the cluster.

For more information about the vFiler unit limit, see [“Adjusting the vFiler unit limit”](#) on page 29.

- ◆ Only Ethernet interfaces are supported for vFiler units.
- ◆ Only the NFS, CIFS, Internet SCSI (iSCSI), HTTP, and FTP protocols are supported for the vFiler units.
- ◆ Not all Data ONTAP commands are supported on the vFiler units.

Hosting storage system's access to data contained in a vFiler unit

As the hosting storage system administrator, you do not have administrative, CIFS, or NFS access to the data owned by the vFiler units (other than data owned by the default vFiler unit, vFiler0). After you assign a qtree or volume to a vFiler unit, you lose access to the data in that qtree or volume.

Example: Before you create a vFiler unit with the /vol/vol1 volume, you can configure the /etc/exports file so that you can mount the /vol/vol1 volume. After you create the vFiler unit, an attempt to mount the /vol/vol1 volume from the hosting storage system results in the error message “.../vol/vol1 belongs to vFiler unit A, can't mount from vol0.”

To regain administrative control of storage owned by a vFiler unit, see [“Administering vFiler unit resources from the hosting storage system”](#) on page 24.

Sample vFiler unit

A storage system named “toaster” has the MultiStore license enabled and is configured with three volumes, which are /vol/vol0, /vol/vol1, and /vol/vol2.

Because the MultiStore license is enabled, a default vFiler unit, vFiler0, is created on toaster.

On toaster, you can create a vFiler unit named vFiler1. You can assign the IP address 123.123.123.1, the qtree /vol/vol0/qtree1, and the volume /vol/vol1 to vFiler1. Traffic destined for vFiler1 uses the IP address 123.123.123.1, and all data served by vFiler1 is stored in the assigned qtree and volume.

If /vol/vol0/qtree1 is the first path specified for vFiler1 on the `vfiler create` command line, this qtree becomes vFiler1's *primary storage*, containing the vFiler's /etc directory.

The storage system named toaster is the hosting storage system for vFiler1. vFiler0 uses all the IP addresses created on toaster except 123.123.123.1 and the following resources for storage:

- ◆ Space in the /vol/vol0 volume that is not in the /vol/vol0/qtree1 qtree

- ◆ The /vol/vol2 volume

For information about accessing a vFiler unit's file system using CIFS and NFS, see [“Accessing a vFiler unit's file system with CIFS and NFS”](#) on page 80.

System administration for a storage system running MultiStore

Types of system administration tasks

The following list describes the types of system administration tasks that you perform when managing a storage system running MultiStore:

- ◆ Hosting storage system tasks, which are the same as those for a storage system that is not supporting vFiler units
- ◆ MultiStore management tasks, which are related to establishing vFiler units and managing resources on vFiler units
- ◆ Tasks related to using Data ONTAP features that are supported by vFiler units

Hosting storage system tasks

You can perform tasks related to managing the resources on the hosting storage system in the same way that you perform them on a storage system without a MultiStore license. You can use either the command line or FilerView to perform these tasks. The following tasks are some examples:

- ◆ Managing volumes, disks, and RAID groups
- ◆ Increasing data availability through Snapshot™ management, SnapMirror management, and volume copy
- ◆ Backing up and recovering data

These tasks are covered in detail in the other volumes of the Data ONTAP System Administration set, particularly the *File Access Management Guide*, the *Storage Management Guide*, and the *Data Protection Guide*.

MultiStore management tasks

From the hosting storage system, you can manage MultiStore by doing the following tasks from the command line or FilerView:

- ◆ Enabling and disabling the MultiStore license
- ◆ Allowing and disallowing protocols to be run on a vFiler unit
- ◆ Creating a vFiler unit
- ◆ Setting up a vFiler unit
- ◆ Starting and stopping a vFiler unit
- ◆ Destroying a vFiler unit
- ◆ Moving resources to and from a vFiler unit
- ◆ Monitoring the status of a vFiler unit

For more information about these tasks, see Chapter 2, “[Managing MultiStore](#),” on page 9.

From a destination storage system, you can perform the following additional tasks:

- ◆ Moving a vFiler unit from one hosting storage system to another
- ◆ Creating a disaster-recovery vFiler unit

See Chapter 6, “[Disaster Recovery and Data Migration](#),” on page 97 for more information.

Tasks related to using Data ONTAP on vFiler units

You can use Data ONTAP features on individual vFiler units in the same way that you use them on a storage system. Some of the tasks related to these features can be performed by vFiler unit administrators. FilerView cannot be used to configure or use these features on vFiler units; you must use the command line. The tasks are as follows:

- ◆ Managing users

Information about managing users (for example, information about using the `/etc/usermap.cfg` file for mapping users) is the same for storage systems and vFiler units.

For more information, see the *System Administration Guide*.

- ◆ Managing iSCSI, LUNs, and initiator groups

For more information, see “[Understanding LUNs on a vFiler unit](#)” on page 52.

- ◆ Managing NFS exports and CIFS shares

For more information, see “[File Access Using NFS and CIFS](#)” on page 79.

- ◆ Managing quotas

For more information, see “[Disk Space Management Using Quotas](#)” on page 85.

- ◆ Preparing vFiler units for virus protection

For more information, see “[Virus Protection for CIFS](#)” on page 149.

About this chapter

This chapter provides information about MultiStore management tasks that can be performed only from the hosting storage system.

Topics in this chapter

This chapter discusses the following topics:

- ◆ [“Establishing a vFiler unit”](#) on page 10
- ◆ [“Administering vFiler unit resources from the hosting storage system”](#) on page 24
- ◆ [“Changing resources for a vFiler unit”](#) on page 25
- ◆ [“Adjusting the vFiler unit limit”](#) on page 29
- ◆ [“Renaming a vFiler unit”](#) on page 32
- ◆ [“Stopping and starting a vFiler unit”](#) on page 33
- ◆ [“Destroying a vFiler unit”](#) on page 35
- ◆ [“Re-creating a destroyed vFiler unit”](#) on page 37
- ◆ [“Allowing or disallowing protocols for a vFiler unit”](#) on page 38
- ◆ [“Displaying vFiler unit status”](#) on page 41
- ◆ [“AutoSupport and syslogd messages about a vFiler unit”](#) on page 42
- ◆ [“Executing commands and setting options for a vFiler unit”](#) on page 43
- ◆ [“Effects of storage system reboot on a vFiler unit”](#) on page 48
- ◆ [“Understanding volumes and qtrees on a vFiler unit”](#) on page 49
- ◆ [“Backing up vFiler units”](#) on page 51
- ◆ [“Understanding LUNs on a vFiler unit”](#) on page 52
- ◆ [“Networking considerations”](#) on page 56
- ◆ [“Understanding SnapMirror on the hosting storage system”](#) on page 57
- ◆ [“Understanding SNMP when used with a vFiler unit”](#) on page 58
- ◆ [“Performance monitoring and statistics”](#) on page 59

Establishing a vFiler unit

Tasks for establishing a vFiler unit

Before you can start protocol servers on a vFiler unit, you must establish a vFiler unit on your storage system.

For detailed information

For detailed information about how to establish a vFiler unit, see the following topics:

- ◆ [“Enabling or disabling the MultiStore license”](#) on page 11
- ◆ [“Planning for vFiler unit creation”](#) on page 13
- ◆ [“Creating a vFiler unit”](#) on page 16
- ◆ [“Understanding the vFiler command family”](#) on page 21
- ◆ [“Setting up a vFiler unit manually”](#) on page 22

Enabling or disabling the MultiStore license

Effects of enabling the MultiStore license

Enabling the MultiStore license has the following effects on the storage system:

- ◆ You can use the `vfiler` and `ipspace` commands.
- ◆ Data ONTAP starts logging vFiler unit status information and sends vFiler unit status information using the AutoSupport feature.
- ◆ The routed daemon is disabled.
- ◆ The `ip.match_any_ifaddr` option is set to Off.
- ◆ The “vFiler unit limit” (the number of vFiler units you can create on this storage system, including vFiler0) is set to a default value between 3 and 11, depending on the memory capacity of the hosting storage system. You can change this value; see “[Adjusting the vFiler unit limit](#)” on page 29 for more information.

Prerequisite for disabling the MultiStore license

You can disable the MultiStore license only when no vFiler units other than vFiler0 are on the storage system. If there are other vFiler units on the storage system, you must destroy them before disabling the MultiStore license. For information about destroying a vFiler unit, see “[Destroying a vFiler unit](#)” on page 35.

Effects of disabling the MultiStore license

Disabling the MultiStore license has the following effects:

- ◆ MultiStore becomes unavailable immediately. You can no longer use the `vfiler` and `ipspace` commands.
- ◆ The routed daemon is enabled after the next reboot.

Enabling or disabling the MultiStore license

To enable or disable the MultiStore license, complete the following step.

Step	Action	
1	If you want to...	Enter this command...
	Enable the MultiStore license	license add <i>license_key</i> Note _____ If you do not have a license key for the MultiStore technology, contact your service representative. _____
	Disable the MultiStore license	license delete multistore

Planning for vFiler unit creation

Prerequisites for vFiler unit creation

The following prerequisites must be met before you can create a vFiler unit:

- ◆ You must create at least one unit of storage (qtrees or volumes—traditional or flexible) before creating the vFiler unit.
- ◆ The storage unit for vFiler unit configuration information must be writable. It must not be a read-only file system, such as the destination volume or qtree in a SnapMirror relationship.
- ◆ The IP address used by the vFiler unit must not be configured when you create the vFiler unit.

Storage guidelines

When deciding which storage units to assign to vFiler units, remember the following guidelines:

- ◆ The first storage unit assigned to the vFiler unit contains the vFiler unit configuration information. It is called the primary storage unit. Although you can remove storage units from a vFiler unit at any time after the vFiler unit is created, the primary storage unit must remain for as long as the vFiler unit exists.
- ◆ The primary storage unit has the same security characteristics it had before it was transferred to the vFiler unit. When you create a new vFiler unit, C\$ share-level permissions are restricted to administrators only, but file-level security is not modified. The vFiler unit administrator can set more restrictive file-level permissions.
- ◆ If the qtree or volume to be used as the primary storage unit contains an /etc directory, the data in the directory is lost after you add the qtree or volume to a vFiler unit. Data in qtrees and volumes used as non-primary storage units is preserved.
- ◆ A volume assigned to a vFiler unit must not be the storage system's root volume. You can, however, assign qtrees in the root volume to a vFiler unit.
- ◆ A volume assigned to a vFiler unit can be a traditional volume or a FlexVol™ volume. You cannot assign aggregates to a vFiler unit.
For information about traditional volumes, FlexVol volumes, and aggregates, see the *Data ONTAP Storage Management Guide*.
- ◆ FlexCache volumes can be created on the default vFiler unit, vfiler0, but cannot be assigned or moved to any other vFiler unit.

- ◆ If the vFiler unit administrator needs to create qtrees on the vFiler unit, assign volumes instead of qtrees to the vFiler unit when creating the vFiler unit. This is because qtrees can be created only at the root of a volume.
- ◆ If you anticipate that you might have to move the disks that are used for the vFiler unit's storage from one storage system to another, assign volumes, not qtrees, to the vFiler unit.

Note

You can move traditional volumes and FlexVol volumes between vFiler units; however, you cannot move FlexCache volumes between vFiler units.

- ◆ When managing NFS exports, CIFS shares, quotas, and options, vFiler unit administrators need to enter the complete path names of the storage resources used by the vFiler units in commands and configuration files. Therefore, choose volume and qtree names appropriately, so that you can share with the vFiler unit administrators the complete path names beginning with *filer_name:/vol/vol_name*.

Naming guidelines

When deciding the vFiler unit name, remember the following guidelines:

- ◆ The name can contain up to 31 alphanumeric ASCII characters.
- ◆ The name can contain the dash (-) and the underscore (_) characters, but the dash character must not be the first character.
- ◆ The name is case-insensitive.
- ◆ The name is unique.

You might also want to incorporate the name of the physical storage system in the vFiler unit name to enable you to easily determine the physical storage system with which a vFiler unit is associated; for example, *mycompanyss1_vfiler1*.

- ◆ The name vFiler0 is reserved.

Language guideline

vFiler unit administrators need to edit the */etc/quotas* and */etc/usermap.cfg* files for their vFiler units. These files support Unicode and root volume UNIX encoding. To ensure that vFiler unit administrators who do not use Unicode-capable editors can edit the files, create vFiler units on a storage system whose root volume language can be used for editing.

Quotas guideline

Creating a vFiler unit involves changing the ownership of a volume or qtree from the hosting storage system to a specified vFiler unit. This change requires that quotas be turned off for the affected volume. You can turn quotas back on for the volume after the vFiler unit is created.

Clustering considerations

Remember the following considerations for establishing vFiler units on a cluster:

- ◆ You can create up to 32 vFiler units on each member of a cluster, depending on the memory capacity of the hosting storage systems. See [“Adjusting the vFiler unit limit”](#) on page 29 for details.
- ◆ Each member of a cluster must have a MultiStore license to take over its partner with a MultiStore license.
- ◆ The vFiler units hosted by the storage systems of the cluster are created and configured independently. That is, each storage system can host a different number of vFiler units, and the vFiler unit configurations on the storage systems can be different from each other.
- ◆ In takeover mode, the functioning storage system takes over all vFiler units created on the failed storage system. These vFiler units include the vFiler units you create and the unit called vFiler0. Therefore, for vFiler units on the failed storage system to work correctly after the takeover, each network interface used by a vFiler unit in a cluster must have a partner interface.

SAN considerations

When you create vFiler units on a storage area network (SAN), keep the following LUN and iSCSI considerations in mind:

- ◆ FCP LUNs and initiator groups are supported only on the hosting storage system, not on vFiler units.
- ◆ iSCSI LUNs and initiator groups are supported on all vFiler units and managed separately for each vFiler unit.
- ◆ When you create a vFiler unit on a storage system on which iSCSI is licensed, the iSCSI service is automatically started on the vFiler unit.

Note

If iSCSI is licensed on the hosting storage system and the storage to be allocated to the vFiler unit contains LUNs, unmap the LUNs.

- ◆ Starting a vFiler unit starts iSCSI packet processing for that vFiler unit.
- ◆ Stopping a vFiler unit stops iSCSI packet processing for that vFiler unit.

Creating a vFiler unit

About this section

This section describes how to create a vFiler unit in the default IP address space (IPspace). For the definition of IPspace and information about creating vFiler units in nondefault IPspaces, see Chapter 3, “[Using vFiler Units in Different IPspaces](#),” on page 61.

State of the vFiler unit after creation

Immediately after you create the vFiler unit, the vFiler unit is running.

The vFiler unit is configured with a set of options with default settings. These settings are not inherited from the hosting storage system. You or the vFiler unit administrator can modify the option settings.

The vFiler unit configuration information is stored in the vFiler unit’s /etc directory. It contains a subset of files that are in the hosting storage system’s /etc directory:

- ◆ cifsconfig.cfg
- ◆ cifssec.cfg
- ◆ exports
- ◆ filersid.cfg
- ◆ group
- ◆ hosts
- ◆ hosts.equiv
- ◆ passwd
- ◆ quotas
- ◆ files related to the registry
- ◆ rmtab
- ◆ sm
- ◆ usermap.cfg

Major steps for creating a vFiler unit

The following list describes the major steps for creating a vFiler unit:

- ◆ Ensure that the network interface to be configured with a vFiler unit IP address is ready for vFiler unit creation.

- ◆ If iSCSI is licensed on the hosting storage system and the storage to be allocated to the vFiler unit contains LUNs, unmap the LUNs.
See the *Block Access Management Guide* for instructions.
- ◆ Assign and configure vFiler unit resources.

Ensuring that the network interface is ready

To ensure that the network interface to be configured with a vFiler unit IP address is ready for vFiler unit creation, complete the following steps.

Step	Action	
1	If the IP address for the vFiler unit is...	Then...
	A base IP address for an interface	Go to Step 3 .
	An IP alias for an interface	Go to Step 2 .
2	If the IP alias is currently...	Then...
	Assigned to an interface	Enter the following command to remove the alias: ifconfig <i>interface</i> -alias <i>address</i> Example: The following example removes the IP alias from the e0 interface: ifconfig e0 -alias 123.123.123.123 You are done.
	Unassigned	You are done.

Step	Action	
3	If the base IP address for the vFiler unit is assigned to...	Then...
	An interface in the UP state	Enter the following command to change the state of the interface for the IP address to DOWN: ifconfig <i>interface</i> down Example: The following example changes the state of the e0 interface to DOWN: ifconfig e0 down
	An interface in the DOWN state	You are done.

Assigning and configuring vFiler unit resources

You can assign and configure resources for a vFiler unit using either the command line or FilerView. To configure the vFiler unit using the command line, complete the following steps.

Note

The procedure that follows assumes that you want to set up the vFiler unit for networking (and CIFS if necessary) immediately after creating the vFiler unit. If you want to defer setup, use the `-n` option of the `vfiler create` command, and then follow the directions in “[Setting up a vFiler unit manually](#)” on page 22 when you are ready to do the setup. For more information on the `vfiler create` command, see the `na_vfiler` manual (`man`) page.

Step	Action
1	Enter the following command: <pre>vfiler create vfiler_name -i ip_address [-i ip_address] ... path [path] ...</pre> <i>vfiler_name</i> is the name of the vFiler unit. <i>ip_address</i> is an IP address. <i>path</i> is the complete path name to an existing volume or qtree. The first path name is the storage unit that contains the /etc directory, which contains the configuration information about the vFiler unit. Example: The following example creates a vFiler unit using two IP addresses, one volume, and one qtree: <pre>vfiler create vfiler1 -i 123.123.123.123 -i 123.123.123.124 /vol/vol1 /vol/vol2/qtrees2</pre>
2	Respond to prompts to set up the storage system, and to set up CIFS if necessary. Results of the setup process: The setup process does the following on the new vFiler unit: ◆ Starts NFS (if NFS is licensed on the hosting storage system) and configures the vFiler unit's primary storage (root volume) to be exported to the vFiler unit's administrative host (using an entry in the /etc/exports file) ◆ Configures the vFiler unit's IP addresses and adds the appropriate entries to /etc/rc ◆ Creates a "pseudo-root" that allows CIFS clients to see all the storage that has been assigned to the vFiler unit as a single tree (see "Accessing a vFiler unit's file system with CIFS and NFS" on page 80 for more information) ◆ Starts the iSCSI service (if iSCSI is licensed on the hosting storage system)

What to do next

You, as the vFiler unit administrator, can now mount the root volume of the vFiler unit, edit `/etc/exports` to suit your needs and rerun the `export fs` command. See Chapter 4, “[File Access Using NFS and CIFS](#),” on page 79 for more information.

Understanding the vFiler command family

Purpose of the vFiler command

The `vfiler` command, which is supported only on the hosting storage system, enables the hosting storage system administrator to set up vFiler units, manage vFiler unit resources, and manage Data ONTAP features on individual vFiler units.

General syntax of the vFiler command

The `vfiler` command family has several commands, and each command has its own syntax. The following syntax is the general `vfiler` command syntax:

`vfiler command vfilertemplate options ...`

Meaning of vfilertemplate option

Some `vfiler` commands support the `vfilertemplate` option. Wherever you see `vfilertemplate` in this manual or in the `na_vfiler` man page, you can substitute a vFiler unit name, a comma-separated list of vFiler unit names, an IPspace declaration, or an asterisk as a wildcard. If you use the asterisk, the command takes effect on all vFiler units, including vFiler0 (the physical storage system), unless the command is inapplicable to vFiler0. See the `na_vfiler` man page for more information.

Format of path names in the vFiler command

Some `vfiler` commands include a path name. It is the complete path name for the qtree or volume assigned to the specified vFiler unit.

Setting up a vFiler unit manually

When to use this section

If you use the default form of the `vfiler create` command, you are prompted for setup information (and CIFS setup information) as soon as the vFiler unit has been created.

In that case, you can skip this section unless you need to change the networking configuration.

If you use the `-n` option of the `vfiler create` command, no setup is done, and no protocol servers will run on the vFiler unit until you set it up manually.

The procedure for setting up a vFiler unit manually is similar to that for setting up a storage system. You run the setup program, which enables you to do the following:

- ◆ Specify the administration host for the vFiler unit
- ◆ Specify the password for the root user on the vFiler unit
- ◆ Specify the bindings of IP addresses to network interfaces
- ◆ Enable DNS and specify the DNS domain and servers
- ◆ Enable NIS client service and specify the NIS domain and servers

Note

The setup program does not prompt you for the time zone. All vFiler units are in the same time zone as the hosting storage system.

If the vFiler unit is licensed to deliver CIFS service, you must run `cifs setup`, as you would for a storage system, in addition to running setup.

Setting up the vFiler unit

To set up the vFiler unit, complete the following steps.

Step	Action
1	Enter the following command: <pre>vfiler run vfiler_name setup</pre> Result: The setup command displays prompts for you to configure the vFiler unit. After you respond to all the prompts, configuration files, such as the /etc/exports file, are created in the /etc directory for the vFiler unit. Note _____ Unlike the setup command for the storage system, the setup command for a vFiler unit does not cause NFS to start running. For more information about starting NFS, see Chapter 4, “File Access Using NFS and CIFS,” on page 79. _____ If the vFiler unit runs the CIFS protocol, go to the next step. Otherwise, you are done.
2	Enter the following command: <pre>vfiler run vfiler_name cifs setup</pre> Result: The cifs setup command displays prompts for you to configure CIFS on the vFiler unit. After you respond to all the prompts, CIFS starts running.

Administering vFiler unit resources from the hosting storage system

Managing vFiler unit storage from the storage system

As the physical storage system administrator, if you need to manage storage resources that belong to a vFiler unit, but you do not have administrative access to the vFiler unit, you can do one of the following:

Note Before taking either of the following actions, unmap any LUNs that have been created in the affected storage resources. For instructions, see the *Block Access Management Guide*.

- ◆ Temporarily move the resources to the hosting storage system (but you cannot move the vFiler unit’s primary /etc volume).
- ◆ Temporarily destroy the vFiler unit.
This returns ownership of all resources to the hosting storage system. No user data is modified when you destroy a vFiler unit.

After you have done what you need to do, you can either move the storage resources back to the vFiler unit or, if you destroyed the vFiler unit, restore the vFiler unit.

Restoring a vFiler unit

To restore a vFiler unit, complete the following step.

Step	Action
1	Enter the following command: <pre>vfiler create oldname -r oldfirstpath</pre> <i>oldname</i> is the name of the original vFiler unit. <i>oldfirstpath</i> is the first path that was specified in the original <code>vfiler create</code> command that created the vFiler unit. For more information about the <code>vfiler create</code> command, see the <code>na_vfiler</code> man page.

Changing resources for a vFiler unit

What you can change

You can add, move, or remove resources for vFiler units. The resources can be network resources (that is, IP addresses) or storage resources (that is, volumes or qtrees).

Effects of changing resources

Adding, removing, and moving vFiler unit resources has the following effects:

- ◆ After you add storage resources to a vFiler unit, the resources are moved from the hosting storage system to a vFiler unit.
- ◆ After you remove storage resources from a vFiler unit, the resources are moved from the vFiler unit to the hosting storage system.
- ◆ After you add an IP address to a vFiler unit, you can assign the address as an IP alias of an interface or assign the address to a network interface that has not been configured.
- ◆ After you remove an IP address from a vFiler unit, the IP address becomes an unassigned IP address.
- ◆ After you move resources from one vFiler unit to another, the resources are removed from the source vFiler unit and added to the destination vFiler unit.

Note

Adding, removing, or moving vFiler unit resources only affects the association between the vFiler unit and the resources. It does not have any effect on user data in the vFiler unit involved.

Requirements for moving and removing resources

Remember the following requirements when you move or remove vFiler unit resources:

- ◆ The storage unit to be moved or removed cannot be the one containing the vFiler unit's /etc directory.
- ◆ If a storage unit that is to be moved or removed contains LUNs, you must unmap the LUNs first. See the *Block Access Management Guide* for instructions.
- ◆ If a storage unit that is to be moved or removed contains any CIFS shares, homedirs, or open files and directories, you must remove the CIFS shares, remove the homedirs from the list of homedirs, or close open files and directories.

- ◆ The following requirements are for moving an IP address between vFiler units or removing an IP address from a vFiler unit:
 - ❖ Both the source vFiler and destination vFiler units must be in the same IPspace.
 - ❖ If the address is an IP alias, the alias must be removed. If the address is not an IP alias, the network interface associated with the address must not be configured.

Considerations for moving resources between vFiler units

Remember the following considerations when you move resources between vFiler units:

- ◆ If a storage unit that is to be moved from one vFiler unit to another contains LUNs, you must unmap the LUNs first. See the *Block Access Management Guide* for instructions.
- ◆ After you move a storage unit from one vFiler unit to another, the security information associated with the files in the storage unit is retained. This might cause users to be unable to access files properly. For example, before the move, a particular user ID (UID) was allowed access to a file; after the move, the UID corresponds to a different user on the destination vFiler unit, and the user previously represented by this UID can no longer gain access the file.
- ◆ If you reassign a volume from one vFiler unit to another, Data ONTAP turns off quotas for the volume. After the volume is moved, you can turn quotas on again for the volume from the destination vFiler unit.
- ◆ If you reassign a qtree from one vFiler unit to another, Data ONTAP turns off quotas for the volume containing the qtree on both the source vFiler unit and the destination vFiler unit. After the qtree is moved, you can turn quotas on again for the volume.
- ◆ All network connections to the vFiler units are terminated when resources are being moved.

Adding resources to a vFiler unit

To add resources to a vFiler unit, complete the following step.

Step	Action
1	Enter the following command: <pre>vfiler add vfiler_name [-f] [-i ip_address [-i ip_address] ...] [path [path ...]]</pre> Use the -f option to skip warning messages. Example: The following example adds an IP address and a volume to an existing vFiler unit: <pre>vfiler add vfiler1 -i 123.123.123.125 /vol/vol3</pre>

Removing resources from a vFiler unit

To remove resources from a vFiler unit, complete the following step.

Step	Action
1	Enter the following command: <pre>vfiler remove vfiler_name [-f] [-i ip_address [-i ip_address] ...] [path [path ...]]</pre> Use the -f option to skip warning messages. Example: The following example removes an IP address and a volume from an existing vFiler unit: <pre>vfiler remove vfiler1 -i 123.123.123.125 /vol/vol3</pre>

Moving resources between vFiler units

To move resources between vFiler units, complete the following step.

Step	Action
1	Enter the following command: <pre>vfiler move <i>source_vfiler destination_vfiler</i> [-f] [-i <i>ip_address</i> [-i <i>ip_address</i>] ...] [<i>path</i> [<i>path</i> ...]]</pre> Use the -f option to skip warning messages. Example: The following example moves an IP address and a volume from one vFiler unit to another: <pre>vfiler move vfiler1 vfiler2 -i 123.123.123.125 /vol/vol3</pre>

Adjusting the vFiler unit limit

What the vFiler unit limit is

The vFiler unit limit specifies the maximum number of vFiler units that can exist on the hosting storage system. Because the limit includes the hosting storage system, vFiler0 (which always exists if the MultiStore license is enabled), the number of vFiler units you can *create* on a storage system is one less than the vFiler unit limit set on a storage system.

In a cluster, the same limit applies to each cluster node.

Default limits

The default vFiler unit limits for systems on which you have just activated a MultiStore license are as follows:

- ❖ 3 for storage systems with less than 1 GB (1,024 MB) of memory
- ❖ 5 for storage systems with 1 GB to less than 2 GB of memory
- ❖ 11 for storage systems with 2 GB or more of memory

The default is 11 for systems on which a MultiStore license has been in effect since a release of Data ONTAP earlier than 6.4.

These limits include vFiler0 (the hosting storage system, which is created when the MultiStore license is enabled and persists as long as the license is in effect), so a limit of 11 means you can create a maximum of 10 vFiler units. In a cluster, a limit of 11 means that you can create a maximum of 10 vFiler units on each cluster node.

To see the current limit, enter the following command:

vfiler limit

Increasing the limit

You can use the `vfiler limit` command to increase the limit to a maximum allowable limit. The maximum number of vFiler units you can have on a storage system depends on the memory capacity of the hosting storage system. The maximum limits are as follows:

- ❖ 11 for storage systems with less than 1 GB (1,024 MB) of memory
- ❖ 26 for storage systems with 1 GB to less than 2 GB of memory
- ❖ 33 for storage systems with 2 GB or more of memory

Use the `sysconfig -v` command to verify the memory size of your system.

Note

There is an exception to the maximum limits. Even though the memory on a FAS 270 storage system is 1,022 MB (< 1,024 MB), it can still support 26 vFiler units.

Example: To verify the memory size on your system, enter the following command on your storage system console:

```
toaster> sysconfig -v

.....
.....
.....
slot 0: System Board 650 MHz (TSANTSA C0)
      Model Name:      FAS270
      Part Number:     110-00046
      Revision:        C0
      Serial Number:   287200
      Firmware release: CFE 1.2.0
      Processors:      2
      Processor revision: B2
      Processor type:   1250
      Memory Size:     1022 MB
      NVMEM Size:      128 MB of Main Memory Used
.
```

In this example, the memory size is 1,022 MB, or less than 1 GB (1,024 MB), so the maximum number of supported vFiler units should be 11. However, the Model Name field in the above example indicates that it is a FAS 270 storage system, therefore, the maximum number of vFiler units in this case is 26.

Example: To raise the number of vFiler units you can create to 15, enter the following command on the hosting storage system:

vfiler limit 16

In a cluster, this sets a limit of 16 vFiler units on each cluster node.

Note

For the change to take effect, you must reboot the storage system (or each storage system in a cluster).

Decreasing the limit You can use the `vfiler limit` command to decrease the limit, setting the maximum number of vFiler units lower than it is now. When you decrease the limit, the change is effective immediately and does not require a reboot of the storage system.

Because the limit includes the hosting storage system vFiler0, which already exists, the number of vFiler units you can create in each case is one less than the limit.

Example: To reduce the number of vFiler units you can create from 15 to 10, enter the following command on the hosting storage system:

```
vfiler limit 11
```

Attention

Before you reboot, make sure that the number of active vFiler units on this storage system (and on any cluster partner) is less than the limit. On reboot, the number of vFiler units started will be at most one less than the limit you specified (one less than the limit on each node in a cluster), even if more vFiler units are configured.

Renaming a vFiler unit

Considerations before renaming a vFiler unit

Consider the following before renaming a vFiler unit:

- ◆ The new name should not exist on the storage system, or on the partner storage system in a cluster.
Although Data ONTAP allows the storage system and its partner to have vFiler units with same names, it is easier to administer the systems if you create a unique name for each vFiler unit.
- ◆ The `vfiler rename` command does not change the CIFS node name of the vFiler unit or affect any client connections that are active.
The `vfiler rename` command changes the name of the vfiler unit only within Data ONTAP; it does not rebroadcast the new name to the CIFS domain controllers or the NetBIOS nameservers because these protocols might be using a different name for the vFiler unit than Data ONTAP uses. If you need to change the name mapping in the CIFS domain controllers, run CIFS setup for each of these protocols.
- ◆ Do not rename a vFiler unit while it is being migrated. Doing so will cause the migrate command on the remote system to fail.

Renaming a vFiler unit

To rename a vFiler unit, complete the following step.

Step	Action
1	Enter the following command: <pre>vfiler rename <i>old_vfiler_name</i> <i>new_vfiler_name</i></pre> Example: The following example renames a vFiler unit named vfiler1 to vfiler2: <pre>vfiler rename vfiler1 vfiler2</pre>

Stopping and starting a vFiler unit

Reasons to stop a vFiler unit

You stop a vFiler unit if you need to troubleshoot vFiler unit problems or destroy a vFiler unit.

Note—————
You cannot stop or start vFiler0.

What it means to stop and start a vFiler unit

After you stop a vFiler unit, the vFiler unit can no longer receive packets from clients.

The stopped state is not persistent across reboots: after you reboot the storage system, the vFiler unit resumes automatically.

You can start a vFiler unit that is in the stopped state; after a vFiler unit starts, it is in running state and can receive packets from clients.

If iSCSI is licensed on the storage system, starting or stopping a vFiler unit starts or stops iSCSI packet processing for that vFiler unit.

Stopping a vFiler unit

To stop a vFiler unit, complete the following step.

Step	Action
1	Enter the following command: vfiler stop vfilertemplate Example: The storage system supports two vFiler units named vFiler1 and vFiler2. The following command stops all vFiler units, except vFiler 0: vfiler stop * The following messages appear after you enter the command: vfiler1 stopped vfiler2 stopped

Starting a vFiler unit

To start a vFiler unit, complete the following step.

Step	Action
1	Enter the following command: vfiler start vfilertemplate Example: The storage system supports two vFiler units named vFiler1 and vFiler2. The following command starts all vFiler units, except vFiler 0, which is already running: vfiler start * The following messages appear after you enter the command: vfiler1 running vfiler2 running

Destroying a vFiler unit

Effects of destroying a vFiler unit

Destroying a vFiler unit has the following effects:

- ◆ All resources associated with the vFiler unit are released to the hosting storage system.
- ◆ No data is destroyed.
- ◆ All the vFiler unit’s IP addresses are unconfigured and the corresponding entries in the storage system’s /etc/rc file are removed.
- ◆ If the vFiler unit was not in the same IPspace as the hosting storage system, the IP addresses previously owned by the vFiler unit are not available for use after you destroy the vFiler unit.
- ◆ The effects on quotas are the same as when you move resources from a vFiler unit to the hosting storage system. See “[Considerations for moving resources between vFiler units](#)” on page 26 for more information about how moving the resources to the hosting storage system affects quotas.
- ◆ Clients using LUNs experience an interruption of service.
You must unmap all of a vFiler unit’s LUNs before you can destroy it. The LUNs themselves are not destroyed. You can remap them from the hosting storage system, or from another vFiler unit if you reallocate the storage to that vFiler unit. For information on mapping and unmapping LUNs, see the *Block Access Management Guide*.

Prerequisite for destroying a vFiler unit

The vFiler unit must be stopped before you can destroy it.

Destroying a vFiler unit

To destroy a vFiler unit, complete the following steps.

Step	Action
1	Unmap any LUNs that are mapped to the vFiler unit’s storage. See the <i>Block Access Management Guide</i> for instructions.

Step	Action
2	Enter the following command to stop the vFiler: <code>vfiler stop <i>vfiler_name</i></code>
3	Enter the following command: <code>vfiler destroy [-f] <i>vfiler_name</i></code> Without the -f option, the command displays a confirmation prompt. With the -f option, the command destroys the vFiler immediately. Result: All resources previously owned by the vFiler are returned to the hosting storage system.

Re-creating a destroyed vFiler unit

Why re-creating a vFiler unit is easy

When you destroy a vFiler unit, its resources are released and the associated configuration information is removed, but your user data remains intact; the data is not destroyed. You can re-create the original vFiler unit that you destroyed by using the `-r` option with the `vfiler create` command. When using the `vfiler create -r` command, you must specify the path to the original vFiler unit’s primary storage unit (containing its `/etc` directory).

When re-creating a vFiler unit, you can change the original name to a new name.

To re-create a vFiler unit, complete the following step.

Step	Action
1	Enter the path to the original vFiler unit’s primary storage unit: <pre>vfiler create vfilername -r origin_vfiler_path</pre> <i>vfilername</i> is the name of the new vFiler unit you are creating. <i>origin_vfiler_path</i> is the path of the original vFiler unit’s <code>/etc</code> storage. Example: <code>vfiler create vfiler1 -r /vol/vol0/qtree1</code>

Note_____

Re-creating a vFiler unit does not restore the IP address bindings. You must manually reconfigure the IP address bindings after you re-create the vFiler unit. See [“Setting up a vFiler unit manually”](#) on page 22 for more information about how to manually reconfigure the IP address bindings for a vFiler unit.

Allowing or disallowing protocols for a vFiler unit

Protocols that a vFiler unit can use

By default, a vFiler unit can use the protocols that are licensed for the hosting storage system. For example, if the hosting storage system is licensed for CIFS and NFS, the vFiler units created can also use the CIFS and NFS protocols. You can select those protocols that you allow or disallow on the vFiler units.

The maximum number of FTP connections to a vFiler unit is determined by the `ftpd.max_connections` option set on the hosting storage system. The value set for this option is shared among the vFiler units on a storage system on a first-come, first-served basis.

Allowing and disallowing protocols

You can select the protocols that you allow on the vFiler units using the `vfiler allow` command. You can select the protocols that you disallow on the vFiler units using the `vfiler disallow` command. vFiler units support the following protocols:

- ◆ CIFS
- ◆ NFS
- ◆ rsh
- ◆ iSCSI
- ◆ FTP
- ◆ HTTP

To allow CIFS, NFS, FTP, or HTTP on a vFiler unit, each of the allowed protocols must have an active license on the hosting storage system.

Effects of disallowing CIFS, NFS, iSCSI, FTP, and HTTP

If a CIFS, NFS, iSCSI, FTP, or HTTP protocol is running and you disallow it, the protocol continues to run until the storage system reboots, but packets destined for the vFiler unit are ignored.

Effects of disallowing iSCSI: After iSCSI is disallowed on a vFiler unit, the following conditions apply on that vFiler unit:

- ◆ You cannot start iSCSI (the `iscsi start` command is rejected).
- ◆ No new iSCSI sessions are allowed.
- ◆ iSCSI commands on existing sessions are rejected.

Effects of disallowing FTP: After FTP is disallowed on a vFiler unit, no new FTP connections are allowed to the vFiler unit; however, transfers that started before FTP was disallowed will finish.

Effects of disallowing HTTP: After HTTP is disallowed on a vFiler unit, no new HTTP connections are allowed on the vFiler unit. Each new request receives a “503 HTTP server is disabled” message. Any existing connections remain alive.

Effects of disallowing rsh

Although the Data ONTAP `rsh.enable` option value (On or Off) determines whether the RSH server is enabled or disabled on a storage system, disallowing rsh on a vFiler unit is independent of the value for that option. That is, a vFiler unit can be configured to disallow rsh even when the `rsh.enable` option is set to On.

Note

You must have the `rsh.enable` option set to On and the vFiler unit configured to allow rsh on a vFiler unit.

Allowing or disallowing protocols for a vFiler unit

To allow or disallow protocols for a vFiler unit, complete the following steps.

Step	Action	
1	If you want to...	Then...
	Allow a protocol	Go to Step 2 .
	Disallow a protocol	Go to Step 3 .
2	To allow a protocol, enter the following command: <code>vfiler allow vfilertemplate proto=protocol ...</code> <i>protocol</i> is <code>nfs</code> , <code>cifs</code> , <code>iscsi</code> , <code>rsh</code> , <code>ftp</code> , or <code>http</code> . Example: The following example allows the NFS and rsh protocols on the vFiler unit named <code>vFiler1</code> : <code>vfiler allow vfiler1 proto=nfs proto=rsh</code> You are done.	

Step	Action
3	To disallow a protocol, enter the following command: <pre>vfiler disallow vfilertemplate proto=protocol ...</pre> <i>protocol</i> is nfs, cifs, iscsi, rsh, ftp, or http. Example: The following command disallows the NFS and rsh protocols on the vFiler unit named vFiler1: <pre>vfiler disallow vfiler1 proto=nfs proto=rsh</pre>

Displaying vFiler unit status

Status information you can display

You can use the `vfiler status` command to display the following types of vFiler unit status information:

- ◆ Whether the vFiler unit is stopped or started
- ◆ IPspace
- ◆ IP addresses that have been assigned and, if they are configured, which interfaces they are bound to
- ◆ Path names to the qtrees or volumes assigned
- ◆ UUID, a globally unique user ID assigned to the vFiler unit
Technical support might need this ID when diagnosing problems related to vFiler units.
- ◆ Protocols allowed
- ◆ Protocols disallowed

See the `na_vfiler` man page for more information.

AutoSupport and syslogd messages about a vFiler unit

About AutoSupport messages

A storage system licensed for a vFiler unit includes the output of the `vfiler status` command in the AutoSupport messages.

About syslogd messages

The output of the `vfiler status` command, which indicates whether the vFiler units are stopped or running, is logged in the `/etc/messages` file on the hosting storage system.

All messages generated by vFiler units are logged to the hosting storage system's `/etc/messages` file. These messages are preceded with the name of the vFiler unit. For example, if the vFiler unit named `vFiler1` exceeds a quota threshold, the following message is generated:

```
[vfiler1@quota.softlimit.exceeded:notice]: Threshold exceeded for tree 3 on volume vol1 for vfiler "vfiler1"
```

Executing commands and setting options for a vFiler unit

Methods for executing commands for a vFiler unit

You can enter commands for a vFiler unit in the following ways:

- ◆ From the context of the vFiler unit, you can enter commands directly.
When a command is run in the context of a vFiler unit, it is executed as if it is being run on the vFiler unit's console. The command is subject to the constraints of that vFiler unit.
See [“Entering commands from the vFiler unit”](#) on page 43.
- ◆ From the hosting storage system, you can use the `vfiler run` command to execute commands for the specified vFiler unit.
See [“Entering commands for a vFiler unit from the hosting storage system”](#) on page 44.
- ◆ From the client of a vFiler unit, you can establish an `rsh` connection with the vFiler unit and then enter the commands.
See [“Entering a command for a vFiler unit through rsh”](#) on page 46.

Note

Not all Data ONTAP commands can be executed for a vFiler unit, and special considerations or restrictions apply to some of those that can be executed. To see what commands are supported, enter `?` from the context of a vFiler unit (see [“Entering commands from the vFiler unit”](#) on page 43) or enter the following command:

`vfiler run vfilertemplate ?`

Special considerations and restrictions are described in the “vFiler considerations” section of each command's man pages.

Entering commands from the vFiler unit

To enter commands for a vFiler unit from the vFiler unit, complete the following steps:

Step	Action
1	To run in the context of the vFiler (as if the command is run on the vFiler unit's console), enter the following command: vfiler context <i>vfiler_name</i> Example: vfiler context vfiler1
2	Enter the command you want to run: Example: ? This shows all the commands available to you from the context of the vFiler unit.
3	To return to the context of the hosting storage system, enter the following command: vfiler context vfiler0

Entering commands for a vFiler unit from the hosting storage system

To enter commands for a vFiler unit from the hosting storage system, complete the following step.

Step	Action
1	Enter the following command: vfiler run <i>vfilertemplate command</i> Example: vfiler run vfiler1 useradmin

Note

Not all Data ONTAP commands can be executed for a vFiler unit. To see what commands are supported, enter **?** from the context of a vFiler unit (see [“Entering commands from the vFiler unit”](#) on page 43) or enter the following command:

vfiler run vfilertemplate ?

Prerequisites for entering commands through rsh from a vFiler unit client

You must meet the following prerequisites if you want to enter commands for a vFiler unit through rsh:

- ◆ The rsh protocol must be allowed for the vFiler unit. By default, rsh is allowed.
- ◆ The rsh protocol must be enabled for the vFiler unit; that is, the `rsh.enable` option for the vFiler unit must be set to On. By default, rsh is enabled.
- ◆ You must enter the command on a client that is permitted to have rsh access to the vFiler unit. In other words, the client must be one of the hosts specified by the `rsh.access` option for the vFiler unit.

Commands that you can enter through rsh

You can enter the commands listed in the following table through an rsh connection to a vFiler unit.

?	help	qtree	passwd
exportfs	quota	options	hostname
lun	igroup	iscsi	

Note_____The hostname command is only for displaying, not changing, the name of the vFiler unit.

Entering a command for a vFiler unit through rsh

To enter a command for a vFiler unit through an rsh connection, complete the following step.

Step	Action
1	Enter the following command: rsh vfiler_IP_address command Example: The following command displays all options on the vFiler unit whose IP address is 123.123.123.1: rsh 123.123.123.1 options

Options that can be set for a vFiler unit

Not all Data ONTAP options are supported on a vFiler unit. You can use the options command for the following options:

- ◆ all cifs options
- ◆ all dns options
- ◆ all nfs options
- ◆ all nis options
- ◆ rsh.access
- ◆ all pcnfsd options
- ◆ wafl.default_nt_user
- ◆ wafl.default_unix_user

- ◆ `wabl.nt_admin_priv_map_to_root`
- ◆ all `wabl.wcc_options`

Exception: You can display the `cifs.wins_servers` option only from the hosting storage system.

Path names specified by options

Path names specified by options are complete path names. For example, the path name specified by the `cifs.home_dir` option should begin with the `/vol/vol_name` prefix.

Effects of storage system reboot on a vFiler unit

Effects of storage system reboot on protocols

When you allow or disallow a protocol on a vFiler unit, this state persists across reboots. For example, if you disallow NFS for a vFiler unit and then reboot the storage system, NFS remains disallowed for the vFiler unit after the reboot.

If you disallow CIFS, NFS, or iSCSI for a vFiler unit when the protocol is enabled, rebooting the storage system disables the protocol for the vFiler unit. If you allow the protocol again after a reboot, you must reenable the protocol.

Effect of storage system reboot on the vFiler unit state

Rebooting the storage system causes all stopped vFiler units to start running. For example, if you stop a vFiler unit and then reboot the storage system, the vFiler unit starts running again after the reboot.

Understanding volumes and qtrees on a vFiler unit

Assigning volumes to a vFiler unit

A volume assigned to a vFiler unit can be a traditional volume or a flexible (FlexVol™) volume. You cannot assign aggregates to a vFiler unit.

For information about traditional volumes, FlexVol volumes, and aggregates, see the *Data ONTAP Storage Management Guide*.

Effects of taking a vFiler unit volume offline

If you take a volume offline and that volume is used for vFiler unit storage:

- ◆ All CIFS shares and NFS exports in the volume are deactivated.
- ◆ If the volume contains the /etc directory for a vFiler unit, the vFiler unit stops running.
After you put the volume back online, the vFiler unit starts running again.
- ◆ All LUNs become unavailable.

Changes required after volumes are renamed

After you rename a volume used for vFiler unit storage, the vFiler unit administrator should change the path names used in the vFiler unit's /etc/exports file accordingly. In addition, the vFiler unit administrator should verify that CIFS shares and quotas are configured properly.

Who can create qtrees

You can create qtrees only if your vFiler unit owns the volume containing the qtree. That is, as the hosting storage system administrator, you can create a qtree in a volume owned by vFiler0, but not in volumes owned by other vFiler units. A vFiler unit administrator can create a qtree in a volume that is a storage unit of the vFiler unit.

Example of creating a qtree on vFiler0: The /vol/vol0 volume is owned by vFiler0. You can use the following command to create a qtree in the /vol/vol0 volume:

```
qtree create /vol/vol0/qtree1
```

Example of creating a qtree on a vFiler unit: The /vol/vol1 volume is owned by vFiler1. The administrator for vFiler1 can use the following command to create a qtree in the /vol/vol1 volume:

```
rsh vfiler1 qtree create /vol/vol1/qtree2
```

Who can change qtree security styles and oplock settings

You can change the security style and oplock setting for a qtree or volume only if you are the owner of the qtree or volume. Therefore:

- ◆ If a vFiler unit owns a volume, you can change the security styles or oplock settings for the volume and all qtrees on the volume from the vFiler unit.
- ◆ If a vFiler unit owns qtrees on a volume owned by the hosting storage system, you can change the security styles or oplock settings from the vFiler unit only for the qtrees the vFiler unit owns.
- ◆ If the hosting storage system owns a volume that contains qtrees assigned to vFiler units, you can change the security styles or oplock settings from the hosting storage system only for the qtrees the hosting storage system owns.

Differences in qtree command output

The qtree command output changes, depending on where you enter the command:

- ◆ If you enter the qtree command from the hosting storage system, the command displays information about all qtrees on the storage system, whether the qtrees are owned by the hosting storage system or vFiler units.
- ◆ If you enter the qtree command from a vFiler unit, the command displays information about qtrees on that vFiler unit only. This applies also to vFiler0; that is, if you enter the qtree command from vFiler0, the command displays information about qtrees owned by vFiler0.
- ◆ If you enter the qtree command without arguments from a vFiler unit, a qtree that is the destination qtree for SnapMirror is shown as read_only in the Status column.

Command for displaying all qtrees and the owning vFiler units

Although you can use the qtree command from the hosting storage system to display all qtrees on the storage system, the list of qtrees in the output is not grouped by the vFiler units that own the qtrees. The following command displays all qtrees and their owning vFiler units:

```
vfiler run * qtree status
```

Backing up vFiler units

Considerations

You can back up vFiler unit data from the hosting storage system or from a vFiler unit's client. Keep the following points in mind when planning vFiler unit backups:

- ◆ From the hosting storage system, you can back up the storage units owned by vFiler units just as you would if the vFiler units did not exist—using the `dump` command, for example.

You can back up all vFiler units at once. This method does not separate the data by vFiler unit, so it is not suitable if each vFiler unit's data must be archived separately.

- ◆ From a client of a vFiler unit, you can back up all of that vFiler unit's data, but no other vFiler unit's data.
 - ❖ A CIFS client can mount “/” from a vFiler unit and see a virtual tree comprising all of that vFiler unit's storage units. (See “[Accessing a vFiler unit's file system with CIFS and NFS](#)” on page 80 for more information.)
 - ❖ A CIFS client can capture all of a vFiler unit's data, both CIFS and NFS.
 - ❖ An NFS client cannot see a virtual tree for the vFiler unit.
 - ❖ An NFS client can back up all of the vFiler unit's NFS data, but not its CIFS data.

If you want to back up an individual vFiler unit's data separately, a good way is to back up from a client (particularly a CIFS client). This backup method does not allow you to back up all vFiler units at once.

Understanding LUNs on a vFiler unit

What LUNs are

Logical Unit Numbers (LUNs) are portions of storage system storage that, when exported, look and act to the importing host like local disks. (They are sometimes referred to as “virtual disks.”) Data on a LUN can be managed at the block level (for example, by a database manager) as well as at the file level. A LUN is the basic unit of storage in a Storage Area Network (SAN).

What is supported on a vFiler unit

Data ONTAP allows you to create and manage a separate set of iSCSI LUNs and initiator groups (igroups) on each vFiler unit. For detailed information about iSCSI LUNs, see the *Block Access Management Guide*.

What is not supported on a vFiler unit

Fibre Channel Protocol (FCP) LUNs are supported only on the hosting storage system, not on a vFiler unit. For detailed information about FCP LUNs, see the *Block Access Management Guide*. This means

- ◆ You can create FCP igroups only on the hosting storage system.
- ◆ FCP-connected hosts can access only those LUNs that are owned by the hosting storage system.
- ◆ The `fc` command does not recognize vFiler units.
- ◆ You can create only iSCSI igroups on vFiler units; you cannot create FCP igroups on vFiler units.

iSCSI LUNs and igroups on a vFiler unit

From the point of view of a host importing LUNs, a vFiler unit looks and acts just like a storage system; administrators of those hosts normally do not even need to be aware that the LUN resides on a storage unit owned by a vFiler unit. But you, as the vFiler unit or hosting storage system administrator, need to be alert to some special considerations.

Be aware of the following points when you manage LUNs on a storage system on which a MultiStore license is enabled:

- ◆ You must create and manage LUNs from the vFiler unit that owns the storage unit that contains the LUNs.

For instructions for switching the console to the context of a vFiler unit, see [“Executing commands and setting options for a vFiler unit”](#) on page 43.

- ◆ A vFiler unit is aware only of those LUNs in the storage units it owns. When executed on a vFiler unit, the `lun show` command displays only that vFiler unit's LUNs.
- ◆ Ownership of LUNs changes with the ownership of the storage unit that contains the LUNs.
- ◆ LUNs must be unmapped before the storage unit containing them can be moved. This means you must unmap all affected LUNs before doing any of the following:
 - ❖ Assigning storage that contains LUNs to a vFiler unit, either when you create the vFiler unit or later
 - ❖ Destroying a vFiler unit that owns storage containing LUNs
 - ❖ Moving storage that contains LUNs from one vFiler unit to another, or between a vFiler unit and the hosting storage system

If you try to move a storage unit without unmapping the LUNs it contains, the operation fails. See the appropriate *Block Access Management Guide* for instructions for unmapping LUNs.

Note

You do not need to unmap LUNs when you migrate a vFiler unit or replace it for disaster-recovery purposes. For more information, see Chapter 6, “[Disaster Recovery and Data Migration](#),” on page 97.

- ◆ igroups are owned by the vFiler unit on which they are created.
- ◆ As with LUNs, a vFiler unit is aware only of those igroups that it owns. When executed on a vFiler unit, the `igroup show` command displays only that vFiler unit's igroups.
- ◆ LUNs must be mapped to igroups owned by the vFiler unit that owns the LUNs.
- ◆ Each vFiler unit has its own name space for LUNs and igroups.
 - ❖ igroups on different vFiler units can have the same initiator.
 - ❖ LUNs on different vFiler units can have the same LUN ID.
- ◆ When you migrate a vFiler unit or replicate it for disaster-recovery purposes, LUNs owned by the vFiler unit are also migrated or replicated, along with their maps, igroups, and iSCSI configuration (the node names and the state of the iSCSI service). However, iSCSI authentication is *not* migrated or replicated.

For more information, see Chapter 6, “[Disaster Recovery and Data Migration](#),” on page 97.

The iSCSI service on a vFiler unit

In general, the iSCSI service operates on individual vFiler units, treating them as though each were a physical storage system. But the iSCSI software adapter itself (iswt), and the commands that manage and report on it and the underlying network interface cards (NICs), operate on the hosting storage system. This is because an iSCSI adapter on a storage system has only one identity (there are no vFiler unit-specific adapter names) and so there is only one set of iSCSI sessions and statistics.

This section lists what is storage system-based and what is vFiler unit-based. For more information on the iSCSI service, see the *Block Access Management Guide*.

The following items are based on the hosting storage system:

- ◆ The `iswt` command, which you use to manage the iSCSI service on the storage system's NICs, operates on the hosting storage system, not on individual vFiler units. The `iswt` driver allows the storage system to be accessed as an iSCSI target device over the storage system's standard network interfaces.
- ◆ If iSCSI is licensed on the hosting storage system, the iSCSI service is started by default during vFiler unit setup.
- ◆ Although most iSCSI subcommands operate specifically on each vFiler unit on which they are executed, the `iscsi stats` command displays statistics by hosting storage system and iSCSI adapter.

The following group is based on the individual vFiler unit:

- ◆ The iSCSI protocol can be allowed and disallowed, and the iSCSI service can be started and stopped, for each vFiler unit.
- ◆ The adapter is online or offline for each vFiler unit, depending on whether the iSCSI service is running or stopped on that vFiler unit.
- ◆ Each vFiler unit has its own iSCSI node name, which includes the vFiler unit's UUID.
- ◆ Portal groups are defined for each vFiler unit.
- ◆ iSCSI subcommands operate specifically on each vFiler unit on which they are executed, except for the `iscsi stats` command.
- ◆ You configure iSCSI security separately for each vFiler unit.

This includes setting the default authentication mode: none, deny, or Challenge Handshake Authentication Protocol (CHAP). For more information about CHAP, see the *Block Access Management Guide*.

 - ❖ In the case of CHAP, there is a separate list of initiators and passwords for each vFiler unit.
- ◆ You can configure iSNS (iSCSI's naming service) separately for each vFiler unit.

Specifically, you can use `iscsi isns` subcommands on each vFiler unit to

- ❖ Configure which iSNS server to use
- ❖ Turn iSNS registration on or off

On creation, a vFiler unit's iSNS configuration is in the UNCONFIGURED state, no matter what the state is on the hosting storage system.

For more information, see the `na_iscsi` man page.

Networking considerations

Effects of a disabled routed daemon

Enabling the MultiStore license causes Data ONTAP to disable the routed daemon.

For more information about the route command, see the `na_route(1)` man page.

Command for changing the routing table in the default IPspace

Because vFiler units in the same IPspace share one routing table, you can manipulate the routing table by entering the route command from the hosting storage system. The route command has the following syntax:

```
route [-fn] add|delete [ host|net ] destination [ gateway metric ]
```

You can include the command in the storage system `/etc/rc` file. In this way, the routes are added automatically after each storage system reboot.

Command for changing routing tables in nondefault IPspaces

For more information about manipulating routing tables in nondefault IPspaces, see Chapter 3, “[Using vFiler Units in Different IPspaces](#),” on page 61.

About the `/etc/dgateways` file

Only the hosting storage system has the `/etc/dgateways` file. vFiler units do not maintain their own `/etc/dgateways` file.

IPSec on a vFiler unit

You can configure a vFiler unit as an Internet Protocol Security (IPSec) endpoint.

You can configure each vFiler unit separately with unique security configuration. When you migrate the vFiler unit from one hosting storage system to another (or replicate it for disaster-recovery purposes), the IPSec configuration is preserved so long as the vFiler unit’s IP address does not change.

For more information, see the *Network Administration Guide* and the `na_ipsec` man page.

Understanding SnapMirror on the hosting storage system

Where to enter the snapmirror command

You can enter the `snapmirror` command from the hosting storage system only, not from a vFiler unit. However, you can use the SnapMirror feature on any qtree or volume on the storage system, whether the qtree or volume is owned by a vFiler unit or the hosting storage system.

Considerations for using the snapmirror command

The procedure for using the `snapmirror` command for data on a vFiler unit is the same as that for data on a storage system.

When typing a storage system name in a path name in the `/etc/snapmirror.conf` file, be sure to use the name of the storage system, not the name of a vFiler unit, even though the path name is for a storage unit in a vFiler unit.

Understanding SNMP when used with a vFiler unit

About SNMP

SNMP is not supported on individual vFiler units; it is supported only on the hosting storage system. You can enable SNMP on the hosting storage system to collect data about vFiler units.

About vFiler unit data in the standard MIB and Data ONTAP custom MIB

In the standard Management Information Base (MIB), all vFiler unit data is global. It pertains to the sum of data from all vFiler units on the storage system, with the following exceptions:

- ◆ Statistics related to network interfaces are for the interfaces in the default IPspace.
- ◆ TCP statistics include data only from the connections and listen sockets in the default vFiler unit.
- ◆ UDP statistics include data only from sockets in the default vFiler unit.
- ◆ Quota information is gathered for each volume. If the hosting storage system or a vFiler unit owns a volume with quotas, quota information is provided for the hosting storage system or vFiler unit owning the volume. If a vFiler unit owns qtrees in a volume that it does not own, no quota information is provided for the vFiler unit.

In the Data ONTAP custom MIB, a group named vFiler is included. It is for per-vFiler unit information, such as the MultiStore license, IP address, protocols allowed, and so on.

Performance monitoring and statistics

Commands for displaying storage system statistics

The `sysstat` command displays storage system statistics, which are the sum of statistics generated by all vFiler units, including vFiler0. You cannot use the `sysstat` command to display statistics for a particular vFiler unit.

The `uptime` command displays the uptime for the storage system. You cannot use the `uptime` command to display the uptime for particular vFiler units.

Commands for displaying NFS and CIFS statistics

You can use the `nfsstat` command and `cifs stat` command to display NFS statistics and CIFS statistics, respectively. These commands can display statistics for the entire storage system, specified vFiler units, or the hosting storage system, depending on the command format, as explained in the following table.

Command entered from the hosting storage system		Statistics displayed
For NFS statistics	For CIFS statistics	
<code>nfsstat</code>	<code>cifs stat</code>	Statistics from all vFiler units combined
<code>vfiler run vfilertemplate nfsstat</code>	<code>vfiler run vfilertemplate cifs stat</code>	Statistics from the specified vFiler units
<code>vfiler run vfiler0 nfsstat</code>	<code>vfiler run vfiler0 cifs stat</code>	Statistics from the hosting storage system

About this chapter

This chapter explains what IPspaces are, why you might need them, and how you create and manage IPspaces and vFiler units in those IPspaces.

Topics in this chapter

This chapter discusses the following topics:

- ◆ [“Understanding IPspaces”](#) on page 62
- ◆ [“Advantages of using VLAN tagging for IPspaces”](#) on page 66
- ◆ [“Requirements for using clusters and IPspaces”](#) on page 67
- ◆ [“Case studies for IPspaces”](#) on page 69
- ◆ [“Managing IPspaces on a storage system”](#) on page 70
- ◆ [“Creating a vFiler unit in a nondefault IPspace”](#) on page 76

Understanding IPspaces

About IPspaces

An IPspace defines a distinct IP address space in which vFiler units can participate. IP addresses defined for an IPspace are meaningful only within that IPspace. A distinct routing table is maintained for each IPspace. No cross-IPspace traffic is routed.

Guidelines for vFiler unit participation in an IPspace

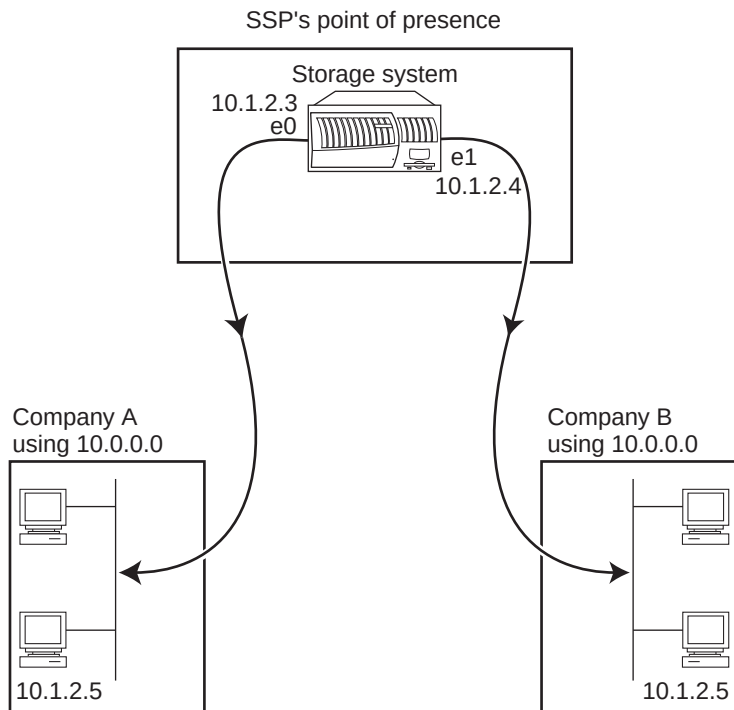
The following guidelines apply to vFiler unit participation in an IPspace:

- ◆ An IPspace can contain multiple vFiler units, but a vFiler unit can belong only to one IPspace.
- ◆ Each vFiler unit in an IPspace must have an IP address that is unique within that IPspace, but a vFiler unit in one IPspace can have the same IP address as a vFiler unit in a different IPspace.
- ◆ Be careful when assigning IPspace to a vFiler unit because you cannot change the assignment without destroying the vFiler unit.
- ◆ Each vFiler unit must have one IP address on the interface that leads to the default gateway of the assigned IPspace. This requirement ensures that the vFiler unit is reachable from within the IPspace.

Typical application of IPspaces

Suppose an SSP needs to connect customers of company A and company B to a storage system on its premises. The SSP creates two vFiler units on the physical storage system—one per customer—and provides a dedicated network path from one vFiler unit to company A's network and from the other vFiler unit to company B's network.

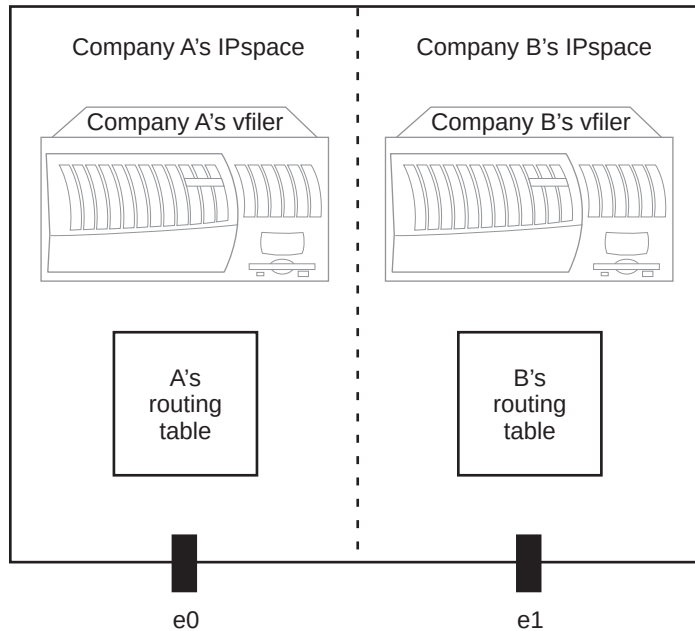
This deployment would work if both companies were using nonprivate IP address ranges; however, as shown in the following illustration, this is not the case.



Both companies use the private IP address subnet 10.0.0.0, thus causing the following problems:

- ◆ The two vFiler units on the storage system at the SSP site have conflicting IP addresses if both companies decide to use the same IP address for their respective vFiler units.
- ◆ If the two companies agree on using different IP addresses for their vFiler units, problems still arise: if any client in Company A's network has the same IP address as a client in Company B's network, packets destined for a client in A's address space might get routed to a client in B's address space, and vice versa.
- ◆ Suppose the two companies decide to use mutually exclusive address spaces (for example, Company A uses 10.0.0.0 with a network mask of 255.128.0.0 and Company B uses 10.128.0.0 with a network mask of 255.128.0.0). The SSP needs to configure static routes on the storage system to route traffic appropriately to A's and B's networks. This solution is neither scalable (because of static routes) nor secure (broadcast traffic is sent to all interfaces of the storage system).

To alleviate these problems, two IPspaces are defined on the storage system—one per vFiler unit. Because a distinct routing table is maintained for each IPspace and no cross-IPspace traffic is routed, the data for each company is securely routed to its respective network even if the two vFiler units are configured in the 10.0.0.0 address space, as shown in the following illustration.



Additionally, the IP addresses referred to by the various configuration files, such as the `/etc/hosts` file, `/etc/hosts.equiv` file, and `/etc/rc` file, are relative to that IPspace. Therefore, the IPspaces allow the SSP to configure the same IP address for the configuration and authentication data for both vFiler units, without conflict.

How interfaces participate in an IPspace

If the storage system is licensed for vFiler units, all its IP-addressable interfaces, including interfaces such as vifs and VLAN, belong to the default IPspace. The default IPspace exists automatically and cannot be destroyed.

When you create a new IPspace, you assign interfaces to the new IPspace from the default IPspace. An interface can belong only to one IPspace.

Routing in an IPspace

A distinct routing table is maintained for each IPspace. All vFiler units participating in an IPspace share its routing table.

Incoming packets: All packets coming in through an interface are tagged with the IPspace identifier (ID) of the IPspace to which the interface belongs. The IP address of the interface and the IPspace ID are used to identify the vFiler unit for which the packet is intended.

Outgoing packets: All outgoing traffic uses the IPspace ID of the vFiler unit that is generating the traffic to determine the routing table to use. Data ONTAP ensures that packets generated by the vFiler units of an IPspace are transmitted through the interfaces that belong to that IPspace.

Note

Broadcast packets are restricted to the vFiler units within the destination IPspace.

Loopback interfaces in IPspaces

Each IPspace has a unique loopback interface assigned to it. The loopback traffic of each IPspace is completely isolated from the other IPspaces.

Advantages of using VLAN tagging for IPspaces

Traffic separation

VLAN tagging for IPspaces provides traffic separation from each customer to the storage system without incurring the cost of additional network devices, such as switches.

Without VLANs, you must provide physically separate network connections to ensure that the traffic from each customer is forwarded securely to and from the storage system. This solution is neither cost-effective nor scalable.

With VLAN tagging, you can set up distinct VLANs for each customer on a single switch; thus, VLAN tagging provides an alternative to physically separate networks.

More IPspaces than interfaces allowed

Using VLAN tagging for IPspaces enables you to set up more IPspaces than there are physical interfaces on the storage system.

Dedicating at least one physical interface per IPspace limits the number of IPspaces that can be set up on a storage system to the number of physical interfaces available on the storage system. VLAN tagging overcomes this limitation. VLAN tags can be used to forward traffic to appropriate IPspaces in cases where more than one IPspace shares the same physical interface.

Secure delivery of packets to a vFiler unit in an IPspace

VLANs inherently confine the broadcast domains. As a result, only vFiler units belonging to a VLAN receive broadcasts intended for that VLAN, even if multiple vFiler units share a physical network interface.

Requirements for using clusters and IPspaces

IPspace naming requirement

The names of IPspaces to which the partner interfaces are assigned must be the same on both storage systems.

For example, in a cluster pair of StorageSystem A and StorageSystem B, if IPspaceA is created on StorageSystem A, an IPspaceA must also exist on StorageSystem B.

IPspace assignment requirement

The partner interfaces on both partners must be assigned to the same-name IPspaces on their respective storage systems.

For example, in a cluster pair of StorageSystem A and StorageSystem B, interface e4 of StorageSystem B is the takeover interface of interface e0 of StorageSystem A, and vice versa. If interface e0 belongs to IPspaceA on StorageSystem A, interface e4 must belong to IPspaceA on StorageSystem B.

Specifying partners in an asymmetric cluster setup

In an asymmetric cluster setup, vFiler unit-IPspace configuration on one cluster partner is different from the other; for example, each partner might have a different number of vFiler units configured in a specific IPspace, or one partner might have no vFiler units.

A “standby” cluster configuration is an example of an asymmetric cluster setup in which one of the hosts is connected to minimal storage of its own. This host takes over its partner’s storage and vFiler units if the partner fails. In such a configuration, the standby host might not have enough storage to support the number of vFiler units that the primary host has. To specify the partner interface when setting up such a cluster configuration, you can use the interface name of the partner instead of the IP address.

For example, two storage systems, StorageSystem1 and StorageSystem2, are configured as shown in the following table.

Storage System	vFiler unit (associated IP address)	IPspace	Interface name
Storage System1	vFiler0 (1.1.1.1)	default	e0
	vFiler1 (2.1.1.1)	ips1	e1
	vFiler2 (3.1.1.1)	ips2	e2
Storage System2	vFiler0 (1.1.1.2)	default	e0
			e4a
			e4b

To specify partner interfaces on Storage System1, complete the following step.

Step	Action
1	Create a host-partner relationship in the following way: ifconfig e0 1.1.1.1 netmask 255.255.255.0 partner e0 ifconfig e1 2.1.1.1 netmask 255.255.255.0 partner e4a ifconfig e2 3.1.1.1 netmask 255.255.255.0 partner e4b

To specify partner interfaces on Storage System2, complete the following steps.

Step	Action
1	Create two IPspaces: ips1 and ips2.
2	Assign interface e4a to ips1 and interface e4b to ips2.
3	Create a host-partner relationship on Storage System2 in the following way: ifconfig e0 1.1.1.2 netmask 255.255.255.0 partner e0 ifconfig e4a partner e1 ifconfig e4b partner e2

Case studies for IPspaces

How SSPs use IPspaces

SSPs typically provide storage services to customers of different companies who have the following specifications:

- ◆ They use the private address space, such as 10.0.0.0, for their intranet.
- ◆ They want a dedicated connection from the storage system at the SSP premises to their intranet.
- ◆ They want to ensure that the storage system is accessible only to the users they authorize.

Using the IPspace solution available on the storage systems, the SSP sets up a dedicated IPspace for each customer, thus creating independent routing domains on a single storage system. Additionally, the SSP uses either physical interfaces or VLAN tags to securely forward traffic to appropriate vFiler units. This approach

- ◆ Is cost effective
- ◆ Scales well
- ◆ Provides easy maintenance and monitoring

How an enterprise uses IPspaces

In an enterprise, the IT organization might have to maintain separate storage systems for each organization of the enterprise. As in the SSP approach, dedicating entire storage systems on an organizational basis can be not only expensive but also difficult to maintain. Setting up vFiler units on a per-organization basis solves the problem. However, distinct IPspaces for these vFiler units are usually not required. Because there is usually a level of trust between different organizations of the enterprise, network access to the various vFiler units does not need to be controlled. Therefore, all vFiler units can belong to the default IPspace.

IPspaces can be used in an enterprise setup as well if there is a need to route traffic from different vFiler units using different default routes.

Managing IPspaces on a storage system

Command for managing IPspaces

You manage IPspaces on a storage system using the `ipspace` command. This command enables you to create, assign interfaces to, destroy, and list IPspaces on a storage system.

For detailed information

For detailed information about how to perform specific tasks using the `ipspace` command, see the following topics:

- ◆ [“Creating an IPspace”](#) on page 71
- ◆ [“Listing IPspaces on a storage system”](#) on page 72
- ◆ [“Assigning an interface to an IPspace”](#) on page 73
- ◆ [“Destroying an IPspace”](#) on page 75

Creating an IPspace

Guidelines for creating an IPspace

The following guidelines apply to creating an IPspace on a storage system:

- ◆ You can have a maximum of 101 IPspaces per storage system. Out of the 101 IPspaces, one is created by default when you install the MultiStore license on your storage system. You can create the remaining 100 IPspaces on the storage system.
- ◆ You can use an alphanumeric string, 1 to 31 characters long, as the IPspace name.
- ◆ All IPspace names you create on a storage system must be unique. However, the IPspace names on cluster partners must be the same.

Creating an IPspace

To create an IPspace on a storage system, complete the following step.

Step	Action
1	Enter the following command: <pre>ipspace create <i>ipspace</i></pre> <i>ipspace</i> is the IPspace name that you want to create. Example: You can create an ipspace1 IPspace on a storage system using the following command: <pre>ipspace create ipspace1</pre>

Managing IPspaces on a storage system

Listing IPspaces on a storage system

Listing IPspaces To list all IPspaces and the interfaces that are assigned to each IPspace, complete the following step.

Step	Action
1	Enter the following command: ipspace list Example: If you enter the <code>ipspace list</code> command on a storage system that has three nondefault IPspaces, you see the following output: <pre>Number of ipspaces configured: 4 default-ipospace (e3) ipspace1 (e2d) ipspace2 (e2c) ipspace3 (e10 e2b sf_vif)</pre>

Managing IPspaces on a storage system

Assigning an interface to an IPspace

Requirement for assigning an interface

The interface you assign from one IPspace to another (including the default IPspace) must not have an IP address configured on it. If IPv6 is enabled on your storage system and an interface is brought up, an IPv6 address is autoconfigured on the interface. Before you can assign this interface to an IPspace, you must remove the autoconfigured IPv6 address using the `-alias` option of the `ifconfig` command.

Removing an IP address from an interface

To remove an IP address from an interface, complete the following steps.

Step	Action	
1	If...	Then...
	The interface is currently configured with an IP address that belongs to a vFiler unit in a nondefault IPspace	Remove the IP address configured for the interface from the vFiler unit. For more information about removing an IP address from a vFiler unit, see “ Changing resources for a vFiler unit ” on page 25. You are done.
	The interface is in the default IPspace	Go to Step 2 .
	If the interface is configured with an autoconfigured IPv6 address	Enter the following command to remove the IPv6 address: ifconfig interface inet6 -alias IPv6_address <i>interface</i> is the name of the interface. <i>IPv6_address</i> is the IPv6 address configured on the interface. You are done.

Step	Action	
	The interface is configured with an IP alias	1. Enter the following command to remove the IP alias: ifconfig <i>interface</i> -alias <i>address</i> <i>interface</i> is the name of the interface. <i>address</i> is the IP address configured for the alias. 2. Go to Step 2.
	The interface is not configured with an IP alias	Go to Step 2 .
2	Enter the following command to remove the IP address: ifconfig <i>interface</i> 0.0.0.0 <i>interface</i> is the name of the interface from which you want to remove the IP address.	

Assigning an interface to an IPspace

To assign an interface to an IPspace, complete the following step.

Step	Action
1	Enter the following command: ipSPACE assign ipSPACENAME <i>interface-name</i> <i>ipSPACENAME</i> is the IPspace name to which the interface will be assigned. <i>interface-name</i> is the name of the interface to be assigned. You can assign only one interface at a time.

Destroying an IPspace

Destroying IPspaces

To destroy an IPspace, complete the following steps.

Step	Action
1	Make sure that there are no network interfaces or vFiler units associated with the IPspace you want to destroy. Read “Listing IPspaces on a storage system” on page 72 to identify any interfaces associated with the IPspace you want to destroy. Read “Managing IPspaces on a storage system” on page 73 to assign network interfaces to another IPspace. Read “Displaying vFiler unit status” on page 41 and “Destroying a vFiler unit” on page 35 for procedures to identify any vFiler units associated with the IPspace and to destroy those vFiler units.
2	Enter the following command: <pre>ipspace destroy <i>ipspace</i>name</pre> <i>ipspace</i>name is the IPspace name to be destroyed.

Creating a vFiler unit in a nondefault IPspace

Guidelines for creating a vFiler unit in a nondefault IPspace

When creating a vFiler unit in a nondefault IPspace, you need to meet the same prerequisites and follow the same guidelines as you would when creating a vFiler unit in the default IPspace. For more information about the prerequisites and guidelines, see “[Planning for vFiler unit creation](#)” on page 13.

Creating a vFiler unit in a nondefault IPspace

To create a vFiler unit in a nondefault IPspace, complete the following steps.

Note— You must follow these steps to configure IP addresses to be used by vFiler units in nondefault IPspaces. Make sure you use the `-n` option of the `vfiler create` command, and do not use the `setup` command to specify IP addresses for interfaces assigned to different IPspaces. The `setup` command (which runs automatically after `vfiler create` unless you use the `-n` option) does not allow duplicate IP addresses even if they are for interfaces in different IPspaces.

Step	Action
1	To create an IPspace, complete the steps in “ Creating an IPspace ” on page 71.
2	To assign an interface to be used by the vFiler unit in the newly created IPspace, complete the steps in “ Assigning an interface to an IPspace ” on page 73.
3	To verify that each interface to be used by the vFiler unit is ready to be configured, complete the steps in “ Ensuring that the network interface is ready ” on page 17.

Step	Action	
4	To create the vFiler unit, enter the following command: <pre>vfiler create vfiler_name -n -s ipspace -i ip_address [-i ip_address] ... path [path] ...</pre> <i>ipspace</i> is the IPspace in which the vFiler unit IP addresses reside. <i>ip_address</i> is an IP address. <i>path</i> is the complete path name to an existing volume or qtree. The first path name is the storage unit that contains the /etc directory, which contains the configuration information about the vFiler unit. Example: The following command creates a vFiler unit named vFiler1 in the IPspace named ipspace1, using the IP address 123.123.123.123 and the /vol/vol1 volume as resources: <pre>vfiler create vfiler1 -n -s ipspace1 -i 123.123.123.123 /vol/vol1</pre>	
5	For each IP address used by the vFiler that is...	Go to...
	An IP alias	Step 6.
	A base IP address	Step 7.
6	Enter the following command to create the IP alias: <pre>ifconfig interface alias address</pre> If the interface to which you are assigning the alias is currently down, go to Step 7. Otherwise go to Step 8.	
7	Use the ifconfig command to configure the interface as Up with the IP address you specified in Step 4. Continue with Step 8.	

Step	Action
8	To modify the routing table for the vFiler unit, enter the following command: <pre>vfiler run vfiler_name route add [host net] [prefixlen prefixlen] destination gateway metric</pre> Example: The following example adds a route to the routing table used by the vFiler unit named vFiler1: <pre>vfiler run vfiler1 route add 1.2.3/24 1.2.3.1 1</pre>
9	For each interface used by the vFiler unit, add the following to the /etc/rc file on the hosting storage system: ◆ An ifconfig command to configure the interface as Up, with the address specified in Step 4 ◆ The route command you entered in Step 8 This configures the interfaces as Up and enforces the route commands across reboots.
10	If the hosting storage system is part of a cluster, edit the /etc/rc file on each cluster partner to ensure that each interface the vFiler unit uses has a partner interface defined for it.
11	The IPspace that the vFiler unit is assigned to must have a default gateway. If the IPspace already has a default gateway, skip this step. Enter the following command to establish the route to a default gateway in the IPspace that the vFiler unit can use: <pre>vfiler run vfiler_name route add default gateway metric</pre> Example: The following example adds a route to the default gateway for the IPspace used by the vFiler unit named vFiler1: <pre>vfiler run vfiler1 route add default 1.2.3.1 1</pre>

About this chapter

This chapter provides information you need for preparing the vFiler unit for NFS and CIFS.

Topics in this chapter

This chapter discusses the following topics:

- ◆ [“Accessing a vFiler unit’s file system with CIFS and NFS”](#) on page 80
- ◆ [“How to specify path names for NFS exports or CIFS shares”](#) on page 81
- ◆ [“Preparing the vFiler unit for NFS”](#) on page 82
- ◆ [“Preparing the vFiler unit for CIFS”](#) on page 83

Accessing a vFiler unit's file system with CIFS and NFS

Accessing a file system with CIFS

For CIFS clients, the root of the primary storage qtree is the root ("/") of a vFiler unit's file system. If "/" is shared, a CIFS client mapping to it can browse all of the vFiler unit's storage in a single tree. This mechanism is called the vFiler unit's "pseudo-root."

In the example "[Sample vFiler unit](#)" on page 5, the root of vFiler1's file system is /vol/vol0/qtree1. If "/" is shared, CIFS clients can browse all of /vol/vol0/qtree1 and /vol/vol1.

Accessing a file system with NFS

Pseudo-root directories are not available to NFS clients. NFS clients must import discrete storage units as they are defined on the hosting storage system.

In the example "[Sample vFiler unit](#)" on page 5, NFS clients needing access to all of vFiler1 storage must import /vol/vol0/qtree1 and /vol/vol1 separately, and will not see a tree connecting them. But administrators of NFS clients can see a list of all of the storage units owned by a vFiler unit in the vFiler unit's /etc/vfilerstorage file.

How to specify path names for NFS exports or CIFS shares

Path names for NFS exports and CIFS shares

When you specify a path to export to NFS clients or to share with CIFS clients, use the complete path name.

Example of a path for an NFS export

Suppose a vFiler unit named vFiler1 uses the /vol/vol1 volume for storage. To export the home directory at the root of this volume to the clients of vFiler1, use /vol/vol1/home in the /etc/exports file or in the `exportfs` command.

Example of a path for a CIFS share

Suppose a vFiler unit named vFiler1 uses the hosting storage system's /vol/vol1 volume as its primary storage. To share the entire volume, and all other storage owned by the vFiler unit, in a single tree, specify "/" as the share path. To offer the home directory at the root of this volume as the home share, specify /home as the path name for the home share. The vFiler unit mechanism that makes this possible is known as "pseudo-root." See "[Accessing a vFiler unit's file system with CIFS and NFS](#)" on page 80 for more information.

Preparing the vFiler unit for NFS

When to use this section

If you use the default form of the `vfiler create` command, you are prompted for NFS setup information as soon as the vFiler unit has been created. At the end of the setup, NFS is running (if NFS is licensed on the physical storage system). Use this section only if you need to start NFS manually.

Starting the NFS protocol

To start NFS, complete the following step.

Step	Action
1	Do one of the following: ◆ From the vFiler unit, enter the following command: nfs on ◆ From the hosting storage system, enter the following command: vfiler run vfiler_name nfs on Result: The NFS protocol server starts running on the vFiler unit.

Exporting all file systems in /etc/exports

To export all paths in the `/etc/exports` file, complete the following step.

Step	Action
1	Do one of the following: ◆ From the vFiler unit, enter the following command: exportfs -a ◆ From the hosting storage system, enter the following command: vfiler run vfiler_name exportfs -a ◆ From a vFiler unit client that is allowed to connect to the vFiler unit through rsh, enter the following command: rsh vfiler_name exportfs -a

Preparing the vFiler unit for CIFS

Different methods of CIFS management

From the hosting storage system, you can use the `vfiler run` command format to issue `cifs` commands for vFiler units. From the vFiler units, you use User Manager or Server Manager to manage user accounts and shares. The `cifs` command is not available through `rsh`.

Tasks that can be performed only on the hosting storage system

Server Manager does not perform all the functions of the following `cifs shares` `-add` and `cifs shares -change` commands. You can execute these commands only from the vFiler unit command line (by means of the `vfiler context` command; see [“Executing commands and setting options for a vFiler unit”](#) on page 43 for more information) or from the hosting storage system (by means of the `vfiler run` command):

```
cifs shares -add -forcegroup group_name
cifs shares -add share_name pathname -nosymlink_strict_security
cifs shares -add -widelink
cifs shares -add -novscan
cifs shares -add -novscanread
cifs shares -change share_name { -forcegroup group_name | -
noforcegroup }
cifs shares -change share_name { -symlink_strict_security | -
nosymlink_strict_security }
cifs shares -change share_name { -widelink | -nowidelink }
cifs shares -change share_name { -vscan | -novscan }
cifs shares -change share_name { -vscanread | -novscanread }
```

No per-vFiler unit limit with CIFS

Data ONTAP does not limit the number of users, shares, open files, and locked files on a per-vFiler unit basis.

About local user accounts for vFiler units

From the hosting storage system, you can use the `useradmin` command to create local accounts for CIFS users of each vFiler unit. Each vFiler unit supports up to 96 local user accounts. The maximum number of vFiler unit user accounts per storage system is 96 times the maximum number of vFiler units for that storage system. (For information on the maximum number of vFiler units for a particular storage system, see [“Adjusting the vFiler unit limit”](#) on page 29.)

About this chapter

This chapter describes how to restrict and track the disk space and number of files used by a user, group, or qtree on a vFiler unit.

Topics in this chapter

This chapter discusses the following topics:

- ◆ [“Understanding quotas”](#) on page 86
- ◆ [“Turning quotas on and off”](#) on page 87
- ◆ [“How to resize quotas”](#) on page 90
- ◆ [“Understanding the /etc/quotas file”](#) on page 91
- ◆ [“Displaying quota status”](#) on page 92
- ◆ [“Understanding quota reports”](#) on page 93
- ◆ [“Messages from exceeded quota thresholds and soft quotas”](#) on page 95

Understanding quotas

Types of quotas supported on a vFiler unit

You can apply the same types of quotas—user, group, and tree quotas—on a vFiler unit as you can on a storage system.

Who manages quota specifications?

The vFiler unit administrator specifies the size of each quota in the vFiler unit's `/etc/quotas` file. That is, the vFiler unit administrator tracks and limits the amount of disk space and the number of files each user, group, or qtree uses.

Exception: If a qtree owned by the vFiler unit resides on a volume owned by the hosting storage system, you, the hosting storage system administrator, can also specify a quota for the qtree in the hosting storage system's `/etc/quotas` file. The following example shows how the tree quota on the hosting storage system affects a vFiler unit qtree.

Suppose the `/vol/vol1/mtree1` qtree is a storage unit of the vFiler unit, and the `/vol/vol1` volume is owned by the hosting storage system.

In the `/etc/quotas` file of the vFiler unit, the vFiler unit administrator specifies that this qtree is limited to 20 GB of disk space. In the `/etc/quotas` file of the hosting storage system, you can specify the disk space limit for the qtree to be 10 GB. This means that if quotas are turned on from the hosting storage system for the `/vol/vol1` volume, the qtree cannot exceed the limit in either `/etc/quotas` file, whichever is lower. In this example, the qtree cannot exceed 10 GB.

The hosting storage system administrator controls the usage of quotas on each volume that the storage system owns using the `quota allow volume` and `quota disallow volume` commands. Once the hosting storage system administrator allows quotas, you can turn quotas on or off on the vFiler units using the `quota on volume` and `quota off volume` commands, respectively.

For more information

For detailed information about quotas, see the *Data ONTAP Storage Management Guide*.

Turning quotas on and off

What happens to quotas when you create a vFiler unit

When you create a vFiler unit, quotas are automatically turned off, on both the hosting storage system and the new vFiler unit, for all the volumes you assign to the new vFiler unit, and for all volumes from which you assign qtrees to the new vFiler unit. (Quotas are also turned off for volumes that you move from one vFiler unit to another.)

Example of assigning a qtree to a new vFiler unit: If quotas have been turned on for the volume /vol/vol1 on the hosting storage system vFiler0, and you then assign a qtree in that volume, /vol/vol1/qtree1, to a different vFiler unit, vFiler1, then the following occurs:

- ◆ Quotas are turned off *for the entire volume* /vol/vol1 on vFiler0.
- ◆ Turning quotas on again for /vol/vol1 on vFiler0 reapplies quotas for /vol/vol1 as specified in vFiler0's /etc/quotas file, except that user and group quotas no longer apply to qtree1. But any specified qtree quota *does* apply.
- ◆ If you now apply user, group, and qtree quotas to /vol/vol1/qtree1 on vFiler1, these quotas will be enforced along with any qtree quota set from vFiler0 (the more restrictive qtree quota takes precedence).

Prerequisite for turning quotas on and off

On a hosting storage system licensed for vFiler units, the hosting storage system administrator must allow quotas for a volume before you can turn quotas on and off for the volume. By default, quotas are allowed on all volumes.

Only hosting storage system administrators can allow or disallow quotas for a volume.

Effects of disallowing quotas

Disallowing quotas on a volume has these effects on all vFiler units that have storage units in the volume:

- ◆ If quotas are currently turned off, you or the vFiler unit administrator are prevented from turning quotas on for that volume.
- ◆ If quotas are currently turned on, they are turned off immediately and are prevented from being turned back on.

Allowing or disallowing quotas for a volume

To allow or disallow quotas for a volume, complete the following step.

Step	Action
1	Enter the following command: quota allow disallow volume Result: After you enter the <code>quota allow</code> command, you can turn on quotas for the specified volume from a vFiler unit. After you enter the <code>quota disallow</code> command, vFiler units are prevented from turning quotas on for the specified volume. If any vFiler units currently have quotas turned on for the volume, quotas are turned off immediately for those vFiler units.

How turning quotas on and off affects a vFiler unit

Turning quotas on using the `quota on volume` command activates quotas on the specified volume based on the contents of `/etc/quotas`. Changes made to `/etc/quotas` do not take effect the next time the `quota on` or `quota resize` command is executed.

Turning quotas off using the `quota off volume` command deactivates the specified volume.

You can turn quotas on and off on a per-volume basis for a vFiler unit. After you turn quotas on for a particular volume, Data ONTAP initializes quotas for the storage units residing on the volume, which are owned by the vFiler unit. Quota on or off states are persistent and stay set after reboots.

Who can turn quotas on and off?

Hosting storage system administrators and vFiler unit administrators can turn quotas on and off for a volume as long as they own some storage space in the volume. For example, from the hosting storage system, you can turn quotas on for the `/vol/vol1` volume if vFiler0, which is the hosting storage system, owns some storage space in the volume.

You can turn quotas on and off for a volume from a vFiler unit regardless of how the quota status for that volume is set from other vFiler units, including vFiler0. For example, if vFiler1 owns the `/vol/vol1/qtree1` qtree and the hosting storage system owns the rest of the `/vol/vol1` volume, you can turn quotas on and off from vFiler1 for the volume regardless of whether the hosting storage system has quotas on or off for the same volume.

Turning quotas on or off from a vFiler unit

To turn quotas on or off for a volume from a vFiler unit, complete the following step.

Step	Action	
1	If you manage the vFiler unit from...	Then...
	The hosting storage system	Enter the following command: vfiler run vfilertemplate quota on off volume
	The vFiler unit	Enter the following command through an rsh connection to the vFiler unit: quota on off volume

For more information

For more information about modifying and turning quotas on and off, see the *Data ONTAP Storage Management Guide*.

How to resize quotas

Method to resize quotas

You resize quotas on the vFiler unit in the same way that you resize quotas on a storage system.

Which quota records are adjusted?

When you enter the `quota resize` command from a vFiler unit to resize quotas for a volume, only the quota records for the vFiler unit are adjusted.

If the vFiler unit consists of qtrees, you can resize the quotas for this volume without turning quotas off and on for the entire volume.

For more information about resizing quotas, see the *Data ONTAP Storage Management Guide*.

Understanding the /etc/quotas file

Format of the /etc/quotas file

The format of the /etc/quotas file for a vFiler unit is the same format as for a storage system. All path names in the file are complete path names.

Where user and group quotas take effect

For each user or group quota, you can specify in the Type field the qtree or volume in which the quota takes effect. You can specify only qtrees and volumes that are owned by the vFiler unit.

If you omit the path name in the Type field, the quota takes effect in the storage system's root volume when quotas are turned on for the root volume.

Recommendation: A vFiler unit administrator should specify a path name in the Type field even though the quota is for a user or group in the storage system's root volume. This ensures that the quota remains applicable even after you change the root volume to a different volume.

Displaying quota status

Volumes for which you can display quota status

You can display quota status for any volume on which your vFiler unit owns storage space.

Displaying quota status

To display quota status, complete the following step.

Step	Action	
1	If you manage the vFiler unit from...	Then...
	The hosting storage system	Enter the following command: vfiler run vfilertemplate quota
	The vFiler unit	Enter the following command through an rsh connection to the vFiler: quota
	Result: The command displays quota status information about all volumes in which the vFiler unit owns storage space. The following messages are sample messages in the command output: vol0: quotas are on. vol1: quotas are off. vol2: quotas are disabled.	

Understanding quota reports

About quota reports for a vFiler unit

A quota report contains information for the vFiler unit from which you enter the `quota report` command.

Example: The `/vol/vol1` volume contains two qtrees, one owned by vFiler1 and the other owned by vFiler2. When you display the quota report for vFiler1, only the quota information pertaining to vFiler1 is shown. No information about the qtrees owned by vFiler2 or the hosting storage system is included in the quota report.

About quota reports for the hosting storage system

When you display a quota report for a hosting storage system, it shows

- ◆ The disk space and the number of files owned by the quota targets in volumes and qtrees that do not belong to vFiler units other than vFiler0.
- ◆ The disk space and the number of files in qtrees owned by other vFiler units but that are restricted by tree quotas specified in the storage system's `/etc/quotas` file. Information about users and groups in these qtrees, however, is not shown.

How the storage system's tree quota affects the vFiler unit's quota report

The K-Bytes and Files fields include the limits for disk space and number of files in a qtree. However, if a qtree owned by a vFiler unit is restricted by the tree quota specified in the storage system's `/etc/quotas` file, the K-Bytes and Files fields in the vFiler unit's quota report might not reflect the limits that are in effect, for these reasons:

- ◆ If the vFiler unit does not impose a tree quota on the qtree, the quota report does not include an entry for the qtree, although the qtree is restricted by the tree quota imposed by the storage system.
- ◆ If the vFiler unit imposes a tree quota on the qtree, the K-Bytes and Files fields in the quota report reflect the limits specified in the vFiler unit's `/etc/quotas` file, although the qtree is restricted by the tree quota imposed by the storage system.

Example: The `/vol/vol1/mtree1` qtree is owned by vFiler1. Suppose the storage system's `/etc/quotas` file specifies that this qtree can use up to 100 GB of disk space, and the vFiler unit's `/etc/quotas` file specifies that this qtree can use up to

150 GB of disk space. When you display a quota report for vFiler1, the entry for qtree1 shows that the maximum amount of disk space allowed is 150 GB, even though qtree1 is actually limited by the more restrictive quota, which is 100 GB.

Command for displaying vFiler unit names in the quota report

The `quota report` command syntax for a vFiler unit is the same as the syntax for a storage system, except that it also takes the `-v` argument. If you use the `quota report -v` command, the report displays the vFiler unit name before the Quota Specifier field.

Messages from exceeded quota thresholds and soft quotas

Where quota warning messages are sent

If a quota threshold or soft quota defined on a vFiler unit is exceeded, a warning message is logged to the storage system console.

Example of the warning message

The following example is a warning message generated when a quota threshold is exceeded:

```
[vfiler1@quota.softlimit.exceeded:notice]: Threshold exceeded for tree 3 on volume vol1 for vfiler "vfiler1"
```


About this chapter

This chapter covers two distinct but similar sets of tasks:

- ◆ Preparing a disaster-recovery or a backup vFiler unit and activating it if a disaster occurs.
- ◆ Moving a vFiler unit (for example, if you are replacing an old storage system with a new one) either by copying data from one physical storage system to another or by using SnapMover® vFiler unit migration between clustered nodes.

Moving a vFiler unit is called “migration.”

Note

The procedures in this chapter do not cover migrating or replicating the hosting storage system, vFiler0. The underlying `vfiler migrate` and `vfiler dr` commands do not operate on vFiler0.

Topics in this chapter

This chapter covers the following topics:

- ◆ [“Understanding disaster recovery and data migration”](#) on page 98
- ◆ [“Preparing the destination storage system”](#) on page 99
- ◆ [“Storage and network checklists”](#) on page 107
- ◆ [“How to create, activate, or delete a disaster-recovery vFiler unit”](#) on page 110
- ◆ [“Reactivating the original vFiler unit”](#) on page 119
- ◆ [“How to move \(migrate\) a vFiler unit”](#) on page 133

Understanding disaster recovery and data migration

Similarities between disaster recovery and data migration

Although the reasons for preparing a disaster-recovery vFiler unit are quite different from the reasons for moving (migrating) a vFiler unit, many of the tasks involved are similar. In both cases, you need to take inventory of the source vFiler unit and make sure the destination storage system has the storage capacity and characteristics, and the network connectivity, to host an identical copy of the original vFiler unit. Then, in both cases, you use a small set of vFiler unit commands to actually create the new vFiler unit, populate it with the original vFiler unit's data, and prepare its connections to the network.

What the procedures do and do not describe: In both cases, the procedures that follow help you not only to move or copy the data (using the SnapMirror technology), but also to configure all of the networking needed to serve the data. But they do not describe adjustments on the client side (for example, mounts onto a Solaris or Windows host). You will need to work closely with the administrators of those systems and with your network administrator to make those adjustments.

Note

A SnapMirror license is required on both the source and destination storage systems.

Differences between disaster-recovery and data migration

After the new vFiler unit has been created, the states of the source and destination vFiler units depend on how you will use the new vFiler unit.

- ◆ If you are preparing a disaster-recovery vFiler unit:
 - ❖ The original (source) vFiler unit remains intact and continues to serve data to its clients.
 - ❖ The new (destination) vFiler unit is inactive, and its IP addresses remain unconfigured, until you activate it in the case of a disaster that destroys or incapacitates the original vFiler unit.
- ◆ If you are moving (migrating) a vFiler unit:
 - ❖ The original (source) vFiler unit is destroyed after the migration is complete.
 - ❖ The new (destination) vFiler unit becomes active and begins serving data as soon as the migration is complete.

Preparing the destination storage system

When to use this section Before moving (migrating) a vFiler unit or setting up a disaster-recovery vFiler unit, you must perform the checks, and the accompanying actions if necessary, listed in “[Storage checks and actions](#)” and “[Network checks and actions](#)” in this section.

Storage checks and actions Photocopy the worksheet “[STORAGE CHECKLIST](#)” on page 107 and fill it out as you perform the following steps.

Note_____

The storage checks are not necessary if you are going to use SnapMover vFiler unit migration to migrate the vFiler unit.

Step	Action	
1	Check that the destination storage system has enough storage space to hold the source vFiler unit’s volumes. On the source storage system, use the <code>vfiler status -r</code> command to see what volumes the vFiler unit is using. Then use the <code>df</code> command on each of those volumes to check how much space is being used. The destination volumes must have at least that much free space; run <code>df</code> on the destination storage system to check. Fill in the <code>df</code> results on the worksheet. Note_____ If the source and destination storage systems use different-sized disks and have different block sizes, adjust the <code>df</code> numbers accordingly.	
2	If...	Then...
	There is enough space on the destination volumes	Continue with Step 3 .
	There is not enough space	a. If necessary, install new disk shelves. b. Use the <code>aggr add</code> command to add new disks to the destination volumes.

Step	Action	
3	Make sure that the destination storage system has the same volume structure as the source and that the volumes to be used by the destination vFiler unit are not used by any other vFiler unit. The volumes to be used by the destination vFiler unit must have the same path names as those used by the source vFiler unit.	
4	If...	Then...
	The destination storage system has volumes whose path names match those used by the source vFiler unit and the volumes to be used by the destination vFiler unit <i>are not used</i> by any other vFiler unit	Continue with Step 5 .
	The destination storage system has a volume whose path name matches that of the source vFiler unit but the volume <i>is used</i> by another vFiler unit	Do one of the following: ◆ If the volume cannot be destroyed, use the <code>vol rename</code> command to rename the volume. Then, create a new volume with the old name of the volume you just renamed. ◆ If the volume can be removed, use the <code>vfiler remove</code> command to disassociate the volume from that vFiler unit. ◆ If the volume is the root volume of the vFiler unit, use the <code>vfiler destroy</code> command to destroy the vFiler unit.
	The destination storage system does not have matching volume path names	Use the <code>aggr create</code> command on the destination storage system to create volumes whose names match those being used by the source vFiler unit, or use <code>aggr rename</code> to rename a volume. For FlexVol volumes, use the <code>vol create</code> command. For more information, see the <i>Data ONTAP Storage Management Guide</i>.
5	Make sure there are no qtrees in the destination volumes whose names match those of qtrees in the source volumes. The migration process uses SnapMirror, which will replicate qtree names from the source to the destination volume, so these names should not already exist on the destination.	

Step	Action	
6	If...	Then...
	There are qtrees in the destination volumes whose names match those of qtrees in the source volumes	Rename the matching qtrees in the destination volumes. To rename a qtree, move it from a client as you would a directory or folder. For more information, see the <i>Data ONTAP Storage Management Guide</i>.
	There are no matching qtree names in the destination volumes	Continue with Step 7 .
7	Check whether quotas are being enforced from the hosting storage system. Quotas enforced from the vFiler unit will be copied to the new vFiler unit, but quotas enforced from the hosting storage system will not. To see where quotas are being enforced from, use the hosting storage system to enter the following command: <pre>quota report</pre>	
8	If...	Then...
	The output of the <code>quota report</code> command shows that quotas for qtrees used by the vFiler unit are being enforced from the hosting storage system	a. Print out a copy of the storage system's quota file: <code>/vol/vol0/etc/quotas</code>. b. Copy the relevant entries into the destination storage system's <code>/vol/vol0/etc/quotas</code> file.
	No quotas for qtrees used by the vFiler unit are being enforced from the hosting storage system	Continue with Step 9 .
9	Continue with “ Network checks and actions ” on page 101.	

Network checks and actions

Photocopy the worksheet “[NETWORK CHECKLIST](#)” on page 108 and fill it out as you perform the following steps.

Step	Action	
1	Check to see if the destination vFiler unit can take over the source vFiler unit's IP addresses. Use the command <code>ifconfig -a</code> on the source vFiler unit and on the destination storage system to display information about all their network interfaces. You can reuse the source IP addresses and aliases on the destination vFiler unit if the destination vFiler unit will be on the same subnet as the source vFiler unit.	
2	If...	Then...
	The source and destination storage systems are on the same subnet Note_____ This is the default case assumed by the <code>vfiler migrate</code> command.	Continue with Step 3.
	They are on different subnets	a. Obtain as many new IP addresses for the destination vFiler unit as are in use on the source vFiler unit. Note_____ You might need to replicate subnet-separation arrangements that exist on the source vFiler unit. For example, the source vFiler unit might use one IP address for a service network and another for an administration network. b. Make a note of the new IP addresses on the worksheet. The <code>vfiler dr configure</code> command will prompt you for these addresses; see “Creating the disaster-recovery vFiler unit” on page 112. The <code>vfiler migrate</code> command will also prompt you for these addresses, but you might then need to run setup on the destination vFiler unit. See “If you have moved the vFiler unit to a different subnet or Windows domain” on page 138.

Step	Action	
3	Check whether the source vFiler unit uses the default IPspace. Use the command <code>ipspace list</code> on the source vFiler unit to display information about IPspaces and the interfaces assigned to them. For more information about IPspaces, see Chapter 3, “Using vFiler Units in Different IPspaces,” on page 61.	
4	If...	Then...
	<code>ipspace list</code> reports default- ipspace	Continue with Step 5 .
	<code>ipspace list</code> reports something other than default- <code>ipspace</code>	a. Use the command <code>ipspace create <i>ipspace</i>name</code> to create a corresponding IPspace with the same name on the destination storage system. b. Use the command <code>ipspace assign <i>ipspace</i>name <i>interfacename</i></code> to assign physical interfaces to the IPspace. These interfaces must all be attached to the same physical network.
5	Check to see if the destination vFiler unit will have access to the same NIS servers as the source. Note_____ You can skip this check if the source and destination vFiler units are on the same subnet. _____ Use the command <code>nis info</code> on the source vFiler unit to see what NIS servers are available to it. (Tip: The <code>ypwhich</code> command shows which server the storage system is currently bound to.)	

Step	Action	
6	If...	Then...
	The destination vFiler unit can use the same NIS servers as the source vFiler unit Note_____ This is the default case assumed by the <code>vfiler migrate</code> command.	Continue with Step 7.
	The destination vFiler unit cannot use same NIS servers	a. Find out which NIS servers are available to the destination storage system. b. Make a note of the IP addresses of those servers on the worksheet. The <code>vfiler dr configure</code> command will prompt you for these addresses; see “Creating the disaster-recovery vFiler unit” on page 112. The <code>vfiler migrate</code> command will <i>not</i> prompt you for these addresses. If you move a vFiler unit to a different subnet, you might need to run <code>setup</code> on the destination vFiler unit. See “If you have moved the vFiler unit to a different subnet or Windows domain” on page 138.
7	Check to see if the destination vFiler unit will have access to the same DNS servers as the source. Note_____ You can skip this check if the source and destination vFiler units are on the same subnet. Use the command <code>dns info</code> on the source vFiler unit to see what DNS servers are available to it.	

Step	Action	
8	If...	Then...
	The destination vFiler unit can use the same DNS servers as the source vFiler unit Note This is the default case assumed by the <code>vfiler migrate</code> command.	Continue with Step 9 .
	The destination vFiler unit cannot use the same DNS servers	a. Find out which DNS servers are available to the destination storage system. b. Make a note of the IP addresses of those servers on the worksheet. The <code>vfiler dr configure</code> command will prompt you for these addresses; see “Creating the disaster-recovery vFiler unit” on page 112. The <code>vfiler migrate</code> command will <i>not</i> prompt you for these addresses. If you move a vFiler unit to a different subnet, you might need to run <code>setup</code> on the destination vFiler unit. See “If you have moved the vFiler unit to a different subnet or Windows domain” on page 138.
9	Check to see if the destination vFiler unit will have access to the same WINS servers and the same Windows security network as the source.	

Step	Action	
10	If...	Then...
	The destination vFiler unit can use the same WINS servers and Windows security network as the source vFiler unit	Continue with Step 11 .
	The destination vFiler unit cannot use the same WINS servers and Windows security network	a. Find out the name and type (Windows NT 4 or Windows 2000) of the domain the destination vFiler unit will be in. b. Make a note of this information on the worksheet. When you activate the disaster-recovery vFiler unit, you will need to configure it into the new domain. See “Activating the disaster-recovery vFiler unit” on page 116. If you move a vFiler unit into a different domain, you will need to configure it into the new domain. See “If you have moved the vFiler unit to a different subnet or Windows domain” on page 138.
11	Check to see if the destination vFiler unit can use the same trusted host for vFiler unit administration as the source vFiler unit.	
12	If...	Then...
	The destination vFiler unit can use the same trusted host as the source vFiler unit	You are done.
	The destination vFiler unit cannot use the same trusted host	a. Find out the name of the new trusted host. b. Make a note of this information on the worksheet. You will need to configure the new trusted-host information after configuring the disaster-recovery vFiler unit, or after moving the vFiler unit. See “Creating the disaster-recovery vFiler unit” on page 112 or “If you have moved the vFiler unit to a different subnet or Windows domain” on page 138.

Storage and network checklists

How to use these checklists

It's a good idea to photocopy the checklists on this page and the next page and fill them out as you go through the checks described earlier in this section.

STORAGE CHECKLIST:

How much disk space is *used* on the source storage system's volumes? (See [Step 1](#) of the “[Storage checks and actions](#)” on page 99.)

df of source storage system's volumes:

How much disk space is *free* on the destination storage system's volumes? (See [Step 1](#) of the “[Storage checks and actions](#)” on page 99.)

df of destination storage system's volumes:

Have you added enough disks to the destination volumes, if necessary? (See [Step 2](#) of the “[Storage checks and actions](#)”.)_____

Do the path names of the source and destination volumes match? (See [Step 4](#) of the “[Storage checks and actions](#)”.)_____

If you are managing qtree-based vFiler units, do any destination volume qtree names match those on the source volume? (See [Step 6](#) of the “[Storage checks and actions](#)”.)_____

Have you copied storage system-based quota information from the source to the destination storage system's /etc/quotas file, if necessary? (See [Step 8](#) of the “[Storage checks and actions](#)”.)_____

NETWORK CHECKLIST:

Are there enough IP addresses available for the vFiler unit on the destination network? (See [Step 2](#) of the “[Network checks and actions](#)” on page 101.)

old interface:_____ new interface:_____

old interface:_____ new interface:_____

old interface:_____ new interface:_____

old interface:_____ new interface:_____

Note

Check syntax carefully; interface names are case-sensitive.

Have you created the number of non-default IPspaces, if any are required? (See [Step 4](#) of the “[Network checks and actions](#)” on page 101.)

Have you gathered all the authority servers? (See [Step 5](#) through [Step 10](#) of the “[Network checks and actions](#)”.)_____

old NIS domain:_____ new NIS domain:_____

old NIS IP addr:_____ new NIS IP addr:_____

old NIS IP addr:_____ new NIS IP addr:_____

old DNS domain:_____ new DNS domain:_____

old DNS IP addr:_____ new DNS IP addr:_____

old DNS IP addr:_____ new DNS IP addr:_____

old DNS IP addr:_____ new DNS IP addr:_____

old WINS IP addr:_____ new WINS IP addr:_____

old WINS IP addr:_____ new WINS IP addr:_____

old NT domain type: NT4 W2K

old domain name (FQDN and NetBIOS):_____

new NT domain type: NT4 W2K

new domain name (FQDN and NetBIOS):_____

Can you use the same trusted host for vFiler unit administration? (See [Step 12](#) of the “[Network checks and actions](#)”)._____

old trusted host name:_____new trusted host name:_____

What to do next

When you have finished the preparation by following the checklists listed in the previous sections, you are ready to create the destination vFiler unit.

- ◆ If you are preparing a disaster-recovery vFiler unit, follow the directions under “[How to create, activate, or delete a disaster-recovery vFiler unit](#)” on page 110.

In this case, the original vFiler unit remains intact and running.

- ◆ If you are moving a vFiler unit from one physical storage system to another, follow the directions under “[How to move \(migrate\) a vFiler unit](#)” on page 133.

In this case, the original vFiler unit is destroyed after it has been replicated on the destination storage system.

How to create, activate, or delete a disaster-recovery vFiler unit

When to use this section

Use the procedures listed in this section if you want to create a backup copy of a vFiler unit (both the data and the vFiler unit configuration) for disaster-recovery purposes.

The procedures assume that the original vFiler unit will remain intact and running unless a disaster puts it out of action. If you want to destroy the original vFiler unit after the new vFiler unit is up and running, follow the procedures listed under “[How to move \(migrate\) a vFiler unit](#)” on page 134 instead.

What to do before or after a disaster

Before a disaster: To create a disaster-recovery vFiler unit, follow this process.

Stage	Description
1	Qualify and prepare the destination storage system. Perform the checks listed in “ Preparing the destination storage system ” on page 99.
2	Create the disaster-recovery vFiler unit. See “ Creating the disaster-recovery vFiler unit ” on page 112.

After a disaster: After a disaster has occurred, and the original vFiler unit is destroyed, incapacitated, or unreachable, follow this process.

Stage	Description
1	Activate the disaster-recovery vFiler unit. See “ Activating the disaster-recovery vFiler unit ” on page 116.
2	To reactivate the original vFiler unit after the damage caused by the disaster has been repaired, see “ Reactivating the original vFiler unit ” on page 119.

Deleting a disaster-recovery vFiler unit: To delete a disaster-recovery vFiler unit, follow the process described in “[Deleting the disaster-recovery vFiler unit](#)” on page 118.

Creating the disaster-recovery vFiler unit

Before you start

Before you begin creating the disaster-recovery vFiler unit, make sure the following are true:

- ◆ You have performed all the preparation steps listed under “[Storage checks and actions](#)” on page 99 and “[Network checks and actions](#)” on page 101.
- ◆ SnapMirror is licensed and enabled on both the source and the destination storage system.
- ◆ The source and destination storage systems can communicate with each other over the network (for example, by means of DNS lookup or entries in /etc/hosts).
- ◆ The destination volumes are online.
- ◆ You know the source storage system’s administrative ID and password.

Creating the vFiler unit

To create the disaster-recovery vFiler unit, perform the following steps.

Attention

In the disaster-recovery storage system, protect any volumes that have the same names as volumes on the original vFiler unit. Otherwise, data in those volumes will be lost.

Step	Action
1	On the destination storage system, enter the following command: <pre>vfiler dr configure source_vfiler@source_filer</pre> Notes: ◆ If you want to set up synchronous SnapMirror between the source and destination storage systems, use the -s option of the <code>vfiler dr configure</code> command. For more information about the -s option, see the <code>na_vfiler</code> man page. ◆ The <code>vfiler dr configure</code> command uses the Data ONTAP SnapMirror feature as its underlying technology. You can set multiple paths from the source to the destination storage systems in SnapMirror. The -a option of the <code>vfiler dr configure</code> command enables you to set multiple paths for the configure operation. For more information, see the section on using SnapMirror over multiple paths in the Data ONTAP <i>Data Protection, Online Backup and Recovery Guide</i>.
2	Respond to the login prompt with a valid administrative ID and password for the source storage system.
3	Respond to the IP address and binding prompts, using the addresses you entered on the “new interface” lines on the worksheet under “ NETWORK CHECKLIST ” on page 108.
4	Respond to the NIS and DNS server prompts, using the addresses you entered under “New NIS addr” and “New DNS addr” on the worksheet under “ NETWORK CHECKLIST ” on page 108.
5	(Optional) Monitor progress using the following command: <pre>vfiler dr status source_vfiler@source_filer</pre> Note_____ The <code>vfiler dr configure</code> command might take some time to complete, especially if a source qtree has many millions of inodes. _____ When the <code>vfiler dr status</code> command reports all the storage units of the source vFiler unit as snapmirrored, proceed to the next step. Note_____ The disaster-recovery vFiler unit now exists, but has not been started. See “What the vfiler dr configure command does” on page 114 for more information. _____

Step	Action
6	If you copied quota information into the destination storage system's /etc/quotas file (see Step 8 of “Preparing the destination storage system” earlier in this chapter), activate the quotas on that storage system. For each volume you are activating quotas on, use the following command: quota on volume_name
7	Edit the disaster-recovery vFiler unit's /etc/hosts.equiv file, adding the name of the trusted host for administering the disaster-recovery vFiler unit. (This is the host name you entered on the worksheet under “NETWORK CHECKLIST” on page 108 as “new trusted host.”) Note_____ If the trusted host is a Windows system, or if it is a UNIX system and the trusted user is not the root user, you need to add the user name as well: for example, adminhost joe_smith. _____
8	Add the path to the root volume and the name of the trusted host to the disaster-recovery vFiler unit's /etc/exports file. Example: /vol/vf1_root access=adminhost, root=adminhost
9	If the vFiler unit's storage units contain iSCSI LUNs, reconfigure iSCSI authentication. See the Data ONTAP <i>Block Access Management Guide</i> for instructions.

What the vfiler dr configure command does

The vfiler dr configure command does the following:

- ◆ Checks that the destination storage system is capable of receiving the source data.
- ◆ Configures and runs SnapMirror to copy the data from the source to the destination vFiler unit.
 - ❖ iSCSI LUNs (including the LUN maps) are copied from the source vFiler unit to the destination vFiler unit.
 - ❖ Initiator groups (igroups) and the iSCSI configuration, including node names and the iSCSI service state, are also copied to the destination vFiler unit.
 - ❖ iSCSI authentication is *not* copied to the destination vFiler unit.
- ◆ Saves the IP configuration information you supply (see [Step 3](#) under “[Creating the vFiler unit](#)” on page 112).
- ◆ Saves the NIS and DNS server information you supply (see [Step 4](#) under “[Creating the vFiler unit](#)” on page 112).

- ◆ Saves the quota information from the source vFiler unit's `/etc/quotas` file.
- ◆ Causes a baseline transfer to occur from the source to the destination.
- ◆ Sets the incremental update interval from the source to the destination to be once every three minutes.
 - ❖ Three minutes is the default setting. If you want to change the default setting, edit the `etc/snapmirror.conf` file as described in the “Data Protection Using SnapMirror” chapter of the *Online Backup and Recovery Guide*.
 - ❖ The `vfiler dr configure` command automatically configures everything that SnapMirror requires for regular updates. No other SnapMirror configuration is necessary. If any SnapMirror configuration requirements are missing from your system (for example, a volume or license is missing), the `vfiler dr configure` command returns errors.
- ◆ Overwrites all data on the volumes of the destination vFiler unit.

You must protect any volumes on the destination storage system that have the same names as the volumes on the source vFiler unit. Otherwise, data in those volumes will be lost.
- ◆ Creates a special type of vFiler unit, known as a DR backup vFiler unit, on the destination storage system.

This vFiler unit is stopped and cannot be started except as described under “[Activating the disaster-recovery vFiler unit](#)” on page 116. Before activation, the vFiler unit will respond only to the `vfiler dr delete`, `vfiler dr status`, and `vfiler dr resync` commands. You should not use `ifconfig` to configure its addresses.

How to create, activate, or delete a disaster-recovery vFiler unit

Activating the disaster-recovery vFiler unit

Activating the vFiler unit To activate the disaster-recovery vFiler unit on the destination storage system, perform the following steps.

Step	Action
1	On the destination storage system, enter the following command: vfiler dr activate source_vfiler@source_filer
2	To change the name of the Windows domain controller, use the cifs prefdc command.
3	To change the Windows WINS server, run cifs setup . Note _____If the Windows domain has changed, you might need to change the permissions on the Windows data files to allow your users the same access they had in the old domain. _____
4	Make adjustments on the clients, such as remounting volumes and qtrees.

Note_____Activating the vFiler unit in the event of a disaster can cause the /etc/hosts.equiv file to be overwritten. If you specified a different set of DNS or NIS servers for the DR location as part of **vfiler dr configure** command, the existing /etc/hosts.equiv file is overwritten, and the old file is copied to /etc/hosts.equiv.bak. Copy the /etc/hosts.equiv.bak file to /etc/hosts.equiv after entering the **vfiler dr activate** command.

What activation does The **vfiler dr activate** command does the following:

- ◆ Breaks the SnapMirror relationships between the source and destination storage systems.
- ◆ Converts the vFiler unit into an ordinary vFiler unit that can be started and that responds to all the commands vFiler units support.

- ◆ Brings LUNs online.
- ◆ Configures IP bindings on the destination vFiler unit according to the information you provided to the `vfiler dr configure` command, adding the destination IP information to the destination `/etc/rc` file. (Any IP information that pertains only to the source vFiler unit is removed from the destination `/etc/rc` file.)
- ◆ Configures the NIS and DNS servers according to the information you provided to the `vfiler dr configure` command.
- ◆ Configures any quota information saved by the `vfiler dr configure` command.

How to create, activate, or delete a disaster-recovery vFiler unit

Deleting the disaster-recovery vFiler unit

Deleting the vFiler unit

To delete a disaster-recovery vFiler unit at any time after a disaster-recovery vFiler unit has been set up, complete the following step.

Step	Action
1	On the destination storage system, enter the following command: <code>vfiler dr delete source_vfiler@source_filer</code>

Before removing the disaster-recovery vFiler unit, the `vfiler dr delete` command also removes all SnapMirror relationships, and any other configuration information related to the disaster-recovery vFiler unit, from the source vFiler unit.

If any errors are detected in SnapMirror relationships, the deletion of the vFiler unit is aborted. To ignore SnapMirror errors and remove the disaster-recovery vFiler unit, you can use the `-f` option available in the `vfiler dr delete` command.

For example, to remove a disaster-recovery vFiler unit on the destination storage system *StorageSystem2* created for a vFiler unit *vfiler1* on source storage system *StorageSystem1* even if SnapMirror errors exist, enter the following command:

```
StorageSystem2> vfiler dr delete -f vfiler1@StorageSystem1
```

Reactivating the original vFiler unit

When to use this section

Use this section when you want to bring the original vFiler unit back online after repairing or replacing the original storage system following a disaster.

How you do this depends on whether you have recovered the original storage system, or are bringing up a new storage system as a replacement.

- ◆ If you have recovered the original storage system, follow the instructions under [“Ways to reactivate the original vFiler unit”](#) on page 119.
- ◆ If you are bringing up a replacement storage system, follow directions under [“Recreating the vFiler unit on a replacement storage system”](#) on page 132.

Ways to reactivate the original vFiler unit

You can reactivate the original vFiler unit in the following ways:

- ◆ If the storage system was not damaged but suffered only a temporary failure, you can select one of the two following methods:
 - ❖ If the storage element associated with the vFiler unit is a volume (traditional or FlexVol), you can reactivate the vFiler unit by resynchronizing the original vFiler unit with the disaster-recovery vFiler unit and then bringing up the original vFiler unit by following the procedure [“Resynchronizing the vFiler unit”](#) on page 120.
You cannot use this method if the storage elements associated with the vFiler units are qtrees.
 - ❖ If the storage element associated with the vFiler unit is a qtree, you can reactivate the vFiler unit using SnapMirror commands, as described in the procedure [“Reactivating the original vFiler unit using SnapMirror commands”](#) on page 126.
If you are not comfortable using SnapMirror commands, follow the procedure described in the next bullet to reactivate the vFiler unit. That procedure is less complicated, but requires more data movement.
- ◆ If the storage system was severely damaged, or you are not comfortable using SnapMirror commands, follow the procedure [“Reactivating the original vFiler unit using vFiler dr commands”](#) on page 129.

Resynchronizing the vFiler unit

About resynchronizing the vFiler unit

Data ONTAP 7.1 introduces a `vfiler dr resync` command that enables you to resynchronize your original vFiler unit with the currently activated disaster-recovery vFiler unit before bringing up the original vFiler unit. This method not only saves you from deleting the original vFiler unit and creating a new vFiler unit, but also does not require a baseline transfer (involving a large amount of data), which would otherwise occur between the new vFiler unit and the disaster-recovery vFiler unit.

How the `vfiler dr resync` command works

If the original vFiler unit needs to be resynchronized, the `vfiler dr resync` command is executed on the storage system on which that vFiler unit exists. If the disaster-recovery vFiler unit has been activated, the `resync` operation proceeds to update the original vFiler unit. After the update, the `resync` operation does not activate the original vFiler unit. The original vFiler unit becomes the disaster-recovery vFiler unit and the currently active disaster-recovery vFiler unit continues to serve data.

If you want to switch the roles of the two vFiler units, you must do that by using the `vfiler dr activate` command on the original vFiler unit.

After the roles have been switched and the original vFiler unit has been reactivated, you can use the `vfiler dr resync` command on the now deactivated disaster-recovery vFiler unit to synchronize it with the original vFiler unit. The disaster-recovery vFiler unit goes back to being the backup vFiler unit, ready for use if the original vFiler unit goes down.

When the `vfiler dr resync` command is executed on the storage system on which the disaster-recovery vFiler unit exists, it can issue an update for the existing storage elements and add and remove storage elements from the disaster-recovery vFiler unit.

Note

The `vfiler dr resync` command does not retrieve any configuration changes that might have been done on the disaster-recovery vFiler unit and preserves the IP information (including DNS and NIS) of the storage system on which the command is run.

Prerequisites for using the vfiler dr resync command

- You can use the `vfiler dr resync` command if the following prerequisites are met:
- ◆ The storage elements for the vFiler unit are only volumes (traditional or FlexVol) and not qtrees.
 - ◆ The source and destination vFiler units contain identical volumes.
 - ◆ The size of volumes on the source and the destination vFiler units is the same.
 - ◆ The vFiler unit from which you are updating is activated.
 - ◆ The original vFiler unit is not in the process of migration.

Resynchronizing the original vFiler unit

To resynchronize the original vFiler unit on the original storage system with the currently activated disaster-recovery vFiler unit, perform the following steps.

Attention —

On the storage system on which you are resynchronizing the original vFiler unit, protect any volumes that have the same names as volumes on the disaster-recovery vFiler unit.

If a volume with the same name exists, the volume is automatically added and initialized for SnapMirror transfers from the disaster-recovery vFiler unit. Any existing data on the newly added volume is lost.

Step	Action
1	Ensure that the original vFiler unit (the one you are trying to resync) is in a stopped state. If not, enter the following command to stop the vFiler unit: <code>vfiler stop vfilername</code>

Step	Action
2	On the original storage system, enter the following command: <pre data-bbox="205 282 1008 340">vfiler dr resync [-l <i>authinfo</i>] [-a <i>alt-remote</i>, <i>alt-local</i>] [-s] dr_vfilername@disaster_recovery_filer</pre> <i>authinfo</i> is the authentication information specified in the <i>username:password</i> format. The <i>username</i> is the login name of the administration host on the disaster recovery storage system and the <i>password</i> is the password for that user name. If you do not specify the authentication information in the <code>vfiler dr resync</code> command, you are prompted for it when the command is run. <i>alt-remote</i> is the alternate hostname or IP address of the source (the disaster recovery storage system, in this case). <i>alt-local</i> is the alternate hostname or IP address of the destination (the original storage system, in this case). -s enables you to set up a synchronous SnapMirror relationship between the source and destination storage systems. <i>dr_vfilername</i> is the name of the disaster-recovery vFiler unit that is currently in the activated state. <i>disaster_recovery_filer</i> is the name of the storage system on which the currently activated disaster-recovery vFiler unit exists. Notes: ◆ The <code>vfiler dr resync</code> command uses the Data ONTAP SnapMirror feature as its underlying technology. You can set multiple paths from the source to the destination storage systems in SnapMirror. The -a option of the <code>vfiler dr resync</code> command enables you to set multiple paths for the resync operation. For more information, see the section on using SnapMirror over multiple paths in the <i>Data ONTAP Data Protection, Online Backup and Recovery Guide</i>. ◆ The <code>vfiler dr resync</code> command resynchronizes all storage elements belonging to the disaster-recovery vFiler unit, including the volumes that were added to the disaster-recovery vFiler unit after it was activated. ◆ When you run the <code>vfiler dr resync</code> command on the disaster-recovery vFiler unit to resync it with the original vFiler unit, you do not need to specify the -l, -a, and -s options. The values stored when <code>vfiler dr configure</code> was run (for a baseline transfer) to back up the original vFiler unit are used.

Step	Action
3	After the resync operation is complete, enter the following command on the storage system on which the original vFiler unit exists to check the status of the vFiler unit that was resynchronized: <pre>vfiler status -r <i>original_vfilername</i></pre> The original vFiler unit will be in a stopped, DR backup state. This is because the <code>vfiler dr resync</code> command does not activate the vFiler unit upon resynchronizing. The vFiler unit continues to behave like a backup disaster-recovery vFiler unit until you use the <code>vfiler dr activate</code> command to reactivate it. For information on how to activate the vFiler unit, see procedure “Activating the disaster-recovery vFiler unit” on page 116.

Step	Action	
4	Reinstate the disaster-recovery vFiler unit on the disaster-recovery storage system.	
	If...	Then...
	The storage elements for the vFiler unit are only volumes	<i>On the disaster-recovery storage system, enter the following command:</i> <pre>vfiler dr resync [-l authinfo] [-a alt-remote, alt-local] [-s] original_vfilename@original_filer</pre> For details of the <code>vfiler dr resync</code> command, see Step 2 of “Resynchronizing the original vFiler unit” on page 121. You are done.
	The storage elements for the vFiler unit are qtrees	<i>1. On the disaster-recovery storage system, destroy the disaster-recovery vFiler unit by entering the following command on the disaster-recovery hosting storage system:</i> <pre>vfiler destroy vfilename</pre> <i>vfilename</i> is the name of the original vFiler unit. <i>2. On the disaster-recovery storage system, re-create the disaster-recovery vFiler unit. From the original storage system or its replacement, perform the procedure “Creating the disaster-recovery vFiler unit” on page 112.</i>

Handling resyncing failures

If the `vfiler dr resync` operation is interrupted or does not complete, you must perform the following steps.

Step	Action	
1	If...	Then...
	There was a “volume offline or does not exist” error	1. Make the volume online or create it. 2. Continue with Step 4.
	There were volume resync errors	Perform the steps listed in “ Reactivating the original vFiler unit using SnapMirror commands ” on page 126. You are done.
	There were no volume resync errors	Continue with Step 2 .
2	Enter the following command to check if the vFiler unit you were resyncing exists on the storage system on which you were running the <code>vfiler dr resync</code> command: <code>vfiler status</code>	
3	If the vFiler unit exists, enter the following command to destroy it: <code>vfiler destroy vfilename</code>	
4	Enter the following command to re-create the vFiler unit you destroyed earlier: <code>vfiler create -r vfilename pathname</code>	
5	Enter the following command to resync the vFiler unit: <code>vfiler dr resync [-l authinfo] [-a alt-remote, alt-local] [-s] dr_vfilename@disaster_recovery_filer</code> For details of the <code>vfiler dr resync</code> command, see Step 2 of “ Resynchronizing the original vFiler unit ” on page 121.	

Reactivating the original vFiler unit using SnapMirror commands

Reactivating the original vFiler unit using SnapMirror commands

To reactivate the original vFiler unit on the original storage system using SnapMirror commands, perform the following steps.

Step	Action
1	Boot the original storage system and interrupt the boot process by pressing the Del or Esc key while the memory self-test is in progress. Note_____ If you don't press the Del or Esc key in time, you can press Ctrl-C when prompted later during the boot; then choose option 5 (maintenance mode); then enter halt. _____
2	At the OK prompt, set the no-vfiler-ips? variable as follows: <pre>setenv no-vfiler-ips? true</pre> This ensures that the storage system does not try to bind IP addresses already being used by the disaster-recovery vFiler unit. When the storage system boots, the original vFiler unit starts running, but it does not accept any read or write requests because its interfaces are unconfigured.
3	Resynchronize the mirrored volumes and qtrees. For each volume and qtree owned by the original vFiler unit, enter the following command on the <i>original storage system</i> (the one you are trying to activate): <pre>snapmirror resync -S disaster_recovery_filer:/pathname original_filer:/pathname</pre> Example: <pre>snapmirror resync -S drfiler:/vol/vfiler1/qtree1 prfiler:/vol/vfiler1/qtree</pre> Troubleshooting: If the snapmirror resync command fails, telling you that there are no matching Snapshot copies, you might have accidentally deleted the Snapshot copies that SnapMirror depends on. In this case, you need to initialize SnapMirror using the snapmirror initialize command (see the na_snapmirror man page for details). Then continue with the next step of this procedure, but skip Step 7 when you get there.

Step	Action	
4	Find out if the <code>snapmirror.access</code> option on the <i>disaster-recovery storage system</i> is set to legacy. Enter the following command on the disaster-recovery storage system: options snapmirror	
5	If...	Then...
	The options <code>snapmirror</code> command returns <code>snapmirror.access legacy</code>	Edit the file <code>/etc/snapmirror.allow</code> and add the host name of the <i>original storage system</i> if it is not already there.
	The options <code>snapmirror</code> command returns a list of host names, and the name of the original storage system is not in the list	Use the options <code>snapmirror</code> command to add the host name of the <i>original storage system</i> . Example: <code>options snapmirror.access host=fridge,toaster,prfiler</code>
6	Run setup <i>on the disaster-recovery vFiler unit</i> and unconfigure its IP addresses.	
7	Update the data on the original vFiler unit. For each volume and qtree owned by the original vFiler unit, enter the following command <i>on the original storage system</i> : snapmirror update -S disaster_recovery_filer:/pathname original_filer:/pathname Example: <code>snapmirror update -S drfiler:/vol/vfiler1/mtree1 prfiler:/vol/vfiler1/mtree1</code>	
8	Stop SnapMirror transfers to the disaster-recovery vFiler unit. For each volume and qtree owned by the original vFiler unit, enter the following command <i>on the original storage system</i> : snapmirror quiesce pathname Example: <code>snapmirror quiesce /vol/vfiler1/mtree1</code> Note _____	
	This operation can take a long time. Use Ctrl-C to interrupt it if you need to. _____	

Step	Action
9	Check that all the paths are quiesced by entering the following command: snapmirror status The status column in the output should show each path as Quiesced.
10	Break the SnapMirror relationship. For each volume and qtree owned by the original vFiler unit, enter the following command <i>on the original storage system</i>: snapmirror break pathname Example: snapmirror break /vol/vfiler1/qtrees1
11	<i>On the original vFiler</i>, run setup to configure the vFiler unit's IP addresses and configure the NIS and DNS servers.
12	If the storage units that have been copied contain iSCSI LUNs, check that the iSCSI configuration <i>on the original vFiler unit</i> is intact and still makes sense. You might need to remap the LUNs and re-create initiator groups (igroups).
13	If the storage units that have been copied contain iSCSI LUNs, bring the LUNs back online <i>on the original vFiler unit</i>.
14	Stop the disaster-recovery vFiler unit. <i>On the disaster-recovery storage system</i>, enter the following command: vfiler stop vfilername <i>vfilername</i> is the name of the disaster-recovery vFiler unit. You are done reactivating the original storage system. To ensure that you have a disaster recovery vFiler unit available for the vFiler unit on the reactivated original storage system, go to the next step.
15	Resynchronize the disaster-recovery vFiler unit. From the <i>disaster-recovery storage system</i>, perform the procedure “Resynchronizing the original vFiler unit” on page 121. You are done.

Reactivating the original vFiler unit using vFiler dr commands

Reactivating the vFiler unit using the vFiler dr commands

To reactivate the original vFiler unit on the original storage system, perform the following steps.

Step	Action
1	Boot the original storage system and interrupt the boot process by pressing the Del or Esc key while the memory self-test is in progress. Note——— If you don't press the Del or Esc key in time, you can press Ctrl-C when prompted later during the boot; then choose option 5 (maintenance mode); then enter halt.
2	At the OK prompt, set the <code>no-vfiler-ips?</code> variable as follows: <pre>setenv no-vfiler-ips? true</pre> This ensures that the storage system does not try to bind IP addresses already being used by the disaster-recovery vFiler unit.
3	At the OK prompt, enter boot.
4	Destroy the original vFiler unit. Enter the following command <i>on the original storage system</i>: <pre>vfiler destroy vfilername</pre> <i>vfilername</i> is the name of the original vFiler unit.
5	Follow the procedure for “Creating a replacement vFiler unit on the original storage system or its replacement” on page 130.

Creating a replacement vFiler unit on the original storage system or its replacement

Follow these steps to create a replacement vFiler unit on the original (production) storage system or its replacement. If you are reactivating the vFiler unit on the original storage system, make sure you perform this procedure only after completing the preparatory steps in [“Reactivating the vFiler unit using the vFiler dr commands”](#) on page 129.

Step	Action
1	Stop the <i>disaster-recovery</i> vFiler unit by using the <code>vfiler stop</code> command.
2	Prepare for the new vFiler unit <i>on the original (or its replacement) storage system</i> : Perform the preparation steps in “Preparing the destination storage system” on page 99.
3	Update the data on the original storage system: For each volume and qtree owned by the new vFiler unit, enter the following commands <i>on the original (or its replacement) storage system</i>: <pre>snapmirror break name snapmirror resync -S disaster_recovery_filer:name production_filer:name</pre> <i>disaster_recovery_filer</i> is the name of the disaster recovery storage system <i>production_filer</i> is the name of the original storage system <i>name</i> is the volume name or path name of the qtree Example 1: For volume vol1, enter the following commands: <pre>snapmirror break drfiler:vol1 snapmirror resync -S drfiler:vol1 prfiler:vol1</pre> Example 2: For qtree qtree1, enter the following commands: <pre>snapmirror break drfiler:/vol/vol2/qtree1 snapmirror resync -S drfiler:/vol/vol2/qtree1 prfiler:/vol/vol2/qtree1</pre>
4	Create the new vFiler unit <i>on the original (or its replacement) storage system</i>, following directions under “Creating the vFiler unit” on page 112. The original vFiler unit is now reactivated on the original storage system or its replacement.

Step	Action	
5	Reinstate the disaster-recovery vFiler unit on the disaster-recovery storage system.	
	If...	Then...
	The storage elements for the vFiler unit are only volumes	<i>On the disaster-recovery storage system, enter the following command:</i> <pre>vfiler dr resync [-l authinfo] [-a alt-remote, alt-local] [-s] original_vfilername@original_filer</pre> For details of the <code>vfiler dr resync</code> command, see Step 2 of “Resynchronizing the original vFiler unit” on page 121. You are done.
	The storage elements for the vFiler unit are qtrees	<i>1. On the disaster-recovery storage system, destroy the disaster-recovery vFiler unit by entering the following command on the disaster-recovery hosting storage system:</i> <pre>vfiler destroy vfilername</pre> <i>vfilername</i> is the name of the original vFiler unit. <i>2. On the disaster-recovery storage system, re-create the disaster-recovery vFiler unit.</i> From the <i>original storage system or its replacement</i>, perform the procedure “Creating the disaster-recovery vFiler unit” on page 112.

Recreating the vFiler unit on a replacement storage system

Recreating the vFiler unit on a replacement storage system

To re-create the original vFiler unit on a replacement storage system, perform the following steps.

Step	Action
1	Boot the replacement storage system.
2	Follow the procedure for “ Creating a replacement vFiler unit on the original storage system or its replacement ” on page 130.

How to move (migrate) a vFiler unit

When to use this section

Use this section if you want to move a vFiler unit (that is, the data and the vFiler unit configuration) from one storage system to another—for example, from a storage system that is overloaded to one that has spare capacity, or from an old storage system to a new one. This is sometimes called “migrating” a vFiler unit.

The procedures in this section assume that you want to destroy the original vFiler unit once the new vFiler unit is up and running. If you want to leave the original vFiler unit intact and create a backup copy of it for disaster-recovery purposes, use the procedures listed under “[How to create, activate, or delete a disaster-recovery vFiler unit](#)” on page 110 instead.

Ways to migrate a vfiler unit

You can migrate a vFiler unit in two ways:

- ◆ By copying data from a source vFiler unit to a destination vFiler unit.
For more information, see “[Migrating the vFiler unit by copying data](#)” on page 134.
- ◆ By changing the software-based disk ownership of the disks between clustered nodes using the SnapMover data migration technology, thus avoiding copying of data from a source vFiler unit to the destination vFiler unit.

For more information about software-based disk ownership, see the *Data ONTAP Storage Management Guide*.

For information about migrating data using SnapMover, see “[Migrating a vFiler unit between clustered nodes with SnapMover](#)” on page 140.

Migrating the vFiler unit by copying data

When to use this section

You should read this section and perform the instructions provided in this section if you want to migrate a vFiler unit by copying data from a physical (source) storage system to another physical (destination) storage system. You might want to copy data from one storage system to another to maintain a backup copy of the data so that if the primary disks fail or the data on them becomes inaccessible, you can use the backup copy to restore services to clients.

Before you start

Before you begin the migration, make sure the following are true:

- ◆ You have performed all the preparation steps listed under “[Storage checks and actions](#)” on page 99 and “[Network checks and actions](#)” on page 101.
- ◆ SnapMirror is licensed and enabled on both the source and the destination storage systems.
- ◆ The source and destination storage systems are on the same subnet.

Note

If the storage systems are not on the same subnet, you need to run setup (and cifs setup if necessary) on the new vFiler unit after the migration to configure authority servers as specified on your worksheet. See “[If you have moved the vFiler unit to a different subnet or Windows domain](#)” on page 138 for details.

- ◆ The source and destination storage systems can communicate with each other over the network (for example, by means of DNS lookup or entries in /etc/hosts).
- ◆ If any of the source qtrees or volumes contain LUNs (virtual disks), the destination storage system is running a version of Data ONTAP that supports virtual disks.

If the source vFiler unit owns iSCSI LUNs, the destination vFiler unit must be running a version of Data ONTAP that supports iSCSI LUNs on vFiler units

- ◆ The destination volumes are online and writable.
- ◆ You know the source storage system’s administrative ID and password.

How migrating a vFiler unit affects clients

If you migrate a vFiler unit to another storage system on the same subnet, you see the following effects on clients:

- ◆ NFS volume-level mounts move transparently.
- ◆ NFS qtree-level mounts need to be remounted.
- ◆ CIFS clients experience the equivalent of a reboot.
- ◆ LUN access continues uninterrupted.

Although an iSCSI host is briefly disconnected from the source vFiler unit, an initiator hides this brief disruption from applications accessing the LUNs. To the applications, access continues without interruption.

For information on moving a vFiler unit to a storage system on a different subnet, see [“If you have moved the vFiler unit to a different subnet or Windows domain”](#) on page 138.

Attention

The procedure that follows destroys the original vFiler unit after replicating it on the destination storage system. If you want to leave the original vFiler unit intact and create a backup copy of it for disaster-recovery purposes, use the procedures listed under [“How to create, activate, or delete a disaster-recovery vFiler unit”](#) on page 110 instead.

Migrating the vFiler unit by copying data

To migrate the storage system by copying data from one vfiler unit to another, perform the following steps.

Note

The following procedure uses the `vfiler migrate` command to migrate the vFiler unit by copying data. For more information about the `vfiler migrate` command and various options available for the command, see the `na_vfiler man` page.

Step	Action
1	Qualify and prepare the destination storage system. Perform the checks listed in “Preparing the destination storage system” on page 99.

Step	Action
2	On the destination storage system, enter the following command: <pre>vfiler migrate start source_vfiler@source_filer</pre> Note_____ This procedure locks up the console for the minimum amount of time. If this is not a concern for you (if you want to let the job run overnight, for example) you can perform the migration with a single command: <pre>vfiler migrate source_vfiler@source_filer</pre> If you do this, skip Step 5 and Step 6. You can also specify the administrative ID, password, and IP address information in this command. For more information, see the na_vfiler man page.
3	Respond to the login prompt with a valid administrative ID and password for the source storage system.
4	Respond to the IP address and binding prompts, using the addresses you entered on the “new interface” lines on the worksheet.
5	Monitor the progress of the migration with the following command: <pre>vfiler migrate status source_vfiler@source_filer</pre> Note_____ The <code>vfiler migrate</code> command might take some time to complete, especially if a source qtree has many millions of inodes.

Step	Action
6	When the status command reports that SnapMirror has replicated all the storage units of the source vFiler unit, complete the migration by entering the following command: <pre>vfiler migrate complete source_vfiler@source_filer</pre> Note_____ If you want to cancel the migration, you can do so now by issuing the following command on the destination storage system: <pre>vfiler migrate cancel source_vfiler@source_filer</pre> This destroys the destination vFiler unit and removes the SnapMirror and other migration-related configuration information from the source vFiler unit. Note_____ If the <code>vfiler migrate complete</code> command reports that the process cannot complete, wait a few minutes and then enter the same <code>vfiler migrate complete</code> command again.
7	If you copied quota information into the destination storage system's <code>/etc/quotas</code> file (see Step 8 of “Preparing the destination storage system” earlier in this chapter) activate the quotas on that storage system. For each volume you are activating quotas on, use the following command: <pre>quota on volume_name</pre>
8	Make adjustments on the clients. You will need to remount qtrees, but volume-level mounts should be intact.
9	If you have moved the vFiler unit to a different subnet, follow the steps in “If you have moved the vFiler unit to a different subnet or Windows domain” on page 138.
10	If the vFiler unit's storage units contain iSCSI LUNS, reconfigure iSCSI authentication. See the <i>Block Access Management Guide</i> for instructions.

What the vFiler unit migrate commands do

The `vfiler migrate start` command does the following:

- ◆ Checks that the destination storage system is capable of receiving the source data.
- ◆ Configures and runs SnapMirror to copy the data from the source to the destination vFiler unit.
- ◆ Saves the IP configuration and binding information you supply (see [Step 4](#) in “Creating the vFiler unit” on page 112).
- ◆ Saves the quota information from the source vFiler unit’s `/etc/quotas` file.

The `vfiler migrate complete` command does the following:

- ◆ Stops the source vFiler unit.
- ◆ Updates the data on the destination vFiler unit.
- ◆ Breaks the SnapMirror relationships.
- ◆ Configures IP bindings on the destination vFiler unit according to the information you provided to the `vfiler migrate start` command, adding the destination IP information to the destination `/etc/rc` file. (Any IP information that pertains only to the source vFiler unit is removed from the destination `/etc/rc`.)
- ◆ Configures any quota information saved by the `vfiler migrate start` command.
- ◆ Destroys the source vFiler unit.
- ◆ Brings LUNs online using migrated LUN maps and igroups.

If you have moved the vFiler unit to a different subnet or Windows domain

If you have moved the vFiler unit to a different subnet, you might need to configure the network servers and make adjustments on the clients. If you have moved the vFiler unit to a different Windows domain, you might also need to modify data-file security attributes.

Perform the following steps *on the destination vFiler unit*.

Step	Action
1	To configure NIS and DNS servers, run <code>setup</code> .
2	To change the name of the Windows domain controller, use the <code>cifs prefdc</code> command.

Step	Action
3	To change the Windows WINS server, run <code>cifs setup</code>. Note_____ If the Windows domain has changed, you might need to change the permissions on the Windows data files to allow your users the same access they had in the old domain.
4	To change the trusted host, do the following: a. Edit the vFiler unit's <code>/etc/hosts.equiv</code> file, adding the name of the trusted host for administering the vFiler unit. (This is the hostname you entered on the worksheet as "new trusted host.") Note_____ If the trusted host is a Windows system, or if it is a UNIX system and the trusted user is not the root user, you need to add the user name as well: for example, <code>adminhost joe_smith</code>. b. Add the path to the root volume and the name of the trusted host to the vFiler unit's <code>/etc/exports</code> file. Example: <code>/vol/vol0 access=adminhost, root=adminhost</code>
5	Make adjustments on the clients. These include remounting volumes and qtrees.

Migrating a vFiler unit between clustered nodes with SnapMover

When to use this section

You should read this section and perform the instructions provided in this section if you want to migrate a vFiler unit without copying its data from a physical storage system (source) to another physical storage system (destination).

What SnapMover vFiler unit migration on clustered nodes is

SnapMover vFiler unit migration on clustered nodes is the no-copy transfer of a traditional volume-level vFiler unit from one host node to the other. For example, if heavy traffic to one vFiler unit is affecting the performance of its host node, and the other cluster node is lightly loaded, you can transfer ownership of that vFiler unit to the host node's cluster partner to balance the load processing on the two nodes.

The SnapMover feature uses software-based disk ownership to transfer ownership of the physical volume that contains the vFiler unit from the original host node to its partner. Because the SnapMover feature carries out the vFiler unit migration through transfer of disk ownership rather than by copying data from one set of disks to another, the migration operation is quickly completed.

Requirements for vFiler unit migration on clustered nodes

Before you use SnapMover to migrate a vFiler unit between clustered nodes, make sure the following are true:

- ◆ The following licenses must be installed on each clustered node:
 - ❖ MultiStore
 - ❖ SnapMover
- ◆ Other licenses used by the vFiler unit need to match. For example, if CIFS is licensed on the source cluster node, it must also be licensed on the destination cluster node. Otherwise, moving the vFiler unit causes CIFS to be unavailable for that vFiler unit after it is moved.
- ◆ Both the source cluster node and the destination cluster node must be connected to the same storage subsystem. The disks must be visible to both the source and destination cluster nodes.
- ◆ To ensure that software-based disk ownership changes are transparent to NFS users, the destination cluster node must have an Ethernet connection to the same subnet that the source cluster node uses.

- ◆ The volume assigned to the vFiler unit must be a traditional volume. You cannot move FlexVol volumes or the aggregates that contain those FlexVol volumes using SnapMirror. For information about traditional and FlexVol volumes, see the *Data ONTAP Storage Management Guide*.
- ◆ The vFiler unit's storage units must all be composed of complete volumes—that is, the vFiler unit's paths must use the form `/vol/volname`. SnapMover migration of storage units naming specific volume subdirectories—for example, `/vol/volname/directory` or `/vol/volname/qtree`, is not supported.
- ◆ The volume containing the configuration information for the vFiler unit (`/vol/volname`) must be writable.

Guidelines for setting up volumes to support vFiler unit migration

When a vFiler unit is migrated, all volumes associated with that vFiler unit are moved. Therefore, when you create the vFiler units, consider the following:

- ◆ SnapMover cannot migrate just a subset of the volumes that are managed by the vFiler unit it is migrating.
- ◆ The destination cluster node must be able to accommodate the size of the vFiler unit with all its associated volumes.
- ◆ You can create up to 32 vFiler units on a storage system.
- ◆ The names of the vFiler units and volumes being moved from the source to the destination must be unique on the destination.

Although you can rename a volume, it is best not to do so if you have NFS clients, because the renaming of the volume is not transparent to NFS clients. When the storage system uses NFS to export a file system, the volume name is part of the exported path name. NFS clients try to mount using the old path name; to access the data after the vFiler unit has been moved, clients will need to remount using the new path name.

What happens during a SnapMover vFiler unit migrate operation

You use the `vfiler migrate -m nocopy` command to carry out SnapMover vFiler unit migration. This command does the following:

- ◆ Ensures that no vFiler unit with the same name exists on the destination node
- ◆ Ensures that the SnapMover license is installed on both the source and destination cluster nodes
- ◆ Ensures that the licenses and Data ONTAP versions are the same on both the source and destination cluster nodes
- ◆ Saves the IP configuration and binding information that you supplied when you created the vFiler unit
- ◆ Saves the quota information from the source vFiler unit's `/etc/quotas` file

- ◆ Stops the source vFiler unit
- ◆ Destroys the source vFiler unit
- ◆ Rewrites the disk ownership information so that the ownership of the vFiler unit volumes passes from the source cluster node to the destination cluster node
- ◆ Re-creates the vFiler unit on the destination cluster node

Enabling SnapMover vFiler unit migration

To enable SnapMover vFiler unit migration between clustered nodes running MultiStore, complete the following steps.

Step	Action	
1	On each cluster node, confirm that MultiStore is licensed. At the command prompt, enter the following command and make sure that Multistore is listed as a licensed feature: license	
2	Temporarily disable the cluster by entering the following command: cf disable	
3	On each cluster node, confirm that the software-based disk ownership is enabled by entering the following command in the command-line interface (CLI) of one of the cluster nodes: disk show	
	If...	Then...
	The system displays disks in the cluster to which disk ownership is assigned	Software-based disk ownership is enabled. Go to Step 10 .
4	Reboot each cluster node. During the boot process, press Ctrl-C to display the boot menu options.	
5	Enter the choice for booting in maintenance mode.	

Step	Action
6	On each cluster node, in maintenance mode, enter the following command: disk upgrade-ownership This command writes software-based disk ownership information to enable SnapMover.
7	Enter the following command on each cluster node to confirm the new software-based disk ownership scheme: disk show The system displays all the disks on the cluster to which ownership has been assigned. Disks assigned to either cluster node will be visible.
8	To halt each cluster node to exit maintenance mode, enter the following command: halt
9	Reboot each cluster node in normal mode.
10	On each cluster node, install the license for SnapMover: license add <i>snapmover_license</i>
11	To re-enable the cluster operation, enter the following command: cf enable

Migrating a vFiler unit from one cluster node to another using SnapMover

Once SnapMover vFiler unit migration is enabled on both cluster nodes, you can carry out a SnapMover vFiler unit migration with the `vfiler migrate -m nocopy` command. To carry out a SnapMover vFiler unit migration, complete the following steps.

Step	Action
1	On the destination cluster node, enter the following command: <pre>vfiler migrate -m nocopy vfilername@source_cl_partner</pre> <i>vfilername</i> is the name of the vFiler unit that you are migrating. <i>source_cl_partner</i> is the cluster node from which you are moving the vFiler unit. Alternatively, enter the following command: <pre>vfiler migrate [-m nocopy [-f]] [-l user:password] [-e ifname:IP address:netmask, ...] vfilername@source_cl_partner</pre> <i>user</i> is the administrative login ID. <i>password</i> is the password for the login ID. <i>ifname</i>, <i>IPaddress</i>, <i>netmask</i> is the interface name, IP address, and netmask, respectively, for the destination vFiler unit. Note_____ If you use the alternate command, skip Step 2.
2	Answer the prompts, including the following information: ◆ A valid administrative login ID and password ◆ The IP address and binding information for the destination vFiler unit Result: The vFiler unit is migrated from the source cluster node to the destination cluster node.
3	Verify that the vFiler unit was moved by entering the following command on the destination cluster node: <pre>vfiler status -r vfilername</pre>

Storage expansion on clusters using software-based disk ownership

If you decide to expand the cluster storage system by adding a disk shelf, you can use the `disk assign` command to divide primary ownership of the disks on that shelf between both cluster nodes regardless of the physical A loop and B loop connections of that shelf.

Because software-based disk ownership is configured on clusters on which vFiler unit migration is enabled, the task of expanding storage on these clusters involves the software assignment of the new disks to one cluster node or the other.

To assign disks that are currently labeled “not owned,” complete the following steps.

Step	Action
1	In the CLI of the cluster node to which you want to assign disks, use the <code>disk show -n</code> command to view all disks that do not have assigned owners.
2	On the cluster node to which you want to assign disks, use the following command to assign the disks that are labeled “not owned”: <pre>disk assign {<i>disk1</i> [<i>disk2</i>] [...] all} [-p {0 1}]</pre> <i>disk1</i> [<i>disk2</i>] [...] are the names of the disks to which you want to assign ownership. <code>all</code> is the option to assign all of the unowned disks to the current storage system head. <code>-p {0 1}</code> is the option to specify a pool assignment (either 0 or 1) for the specified disk if SyncMirror® volume replication is set up on the current cluster node. Example: The following command assigns six disks in the cluster to the storage system fh1: <pre>fh1> disk assign 0b.43 0b.41 0b.39 0b.37 0b.35 0b.33</pre> Result: The specified disks are assigned as disks to the specified storage system.
3	Use the <code>disk show -v</code> command to verify the disk assignments that you have just made and survey the disks that remain to be assigned.

After you have assigned disks to one of the two cluster nodes, you can assign those disks to volumes on the storage system that owns them or leave them as spare disks on the storage system that owns them.

**Disabling
SnapMover vFiler
unit migration and
software-based disk
ownership**

To disable SnapMover vFiler unit migration between clustered nodes running MultiStore, and return to A loop B loop ownership assignment, you might first need to reconfigure your software-based disk ownership assignments to be consistent with A loop B loop ownership assignment.

To disable SnapMover vFiler unit migration and software-based disk ownership, complete the following steps.

Step	Action
1	On one of the cluster nodes, enter the following command to ascertain whether or not the current software-based disk ownership assignments align with the cluster's A loop B loop topology: aggr status -r For backward compatibility, you can also enter the following command: vol status -r Results: In the output of this command, all disks assigned to the current cluster node are listed as volume disks or as spare disks; all disks assigned to the cluster's partner are listed as partner disks. All disks on the same shelf must be assigned to the same cluster node and all disk shelf assignments must be consistent with the A loop B loop topology of the cluster.

Step	Action	
2	If	Then..
	The disk ownership deviates from the A loop B loop topology because you have migrated a vFiler unit with the vfiler migrate -m nocopy command...	Change it back with the same command. See “ Migrating a vFiler unit from one cluster node to another using SnapMover ” on page 144. Then go to Step 3.
	The disk ownership deviates from the A loop B loop topology because you have split ownership of a single expansion disk shelf between the two cluster nodes...	Use the disk assign command to assign all disks on the shelf to the cluster node attached to the shelf’s A loop connection. See the section on modifying disk assignments in the <i>Data ONTAP Storage Management Guide</i> . Then go to Step 3.
	The disk ownership deviates from the A loop B loop topology because you have reassigned a spare disk with the disk assign command...	Change it back with the disk assign command. See the section on modifying disk assignments in the <i>Data ONTAP Storage Management Guide</i> . Then go to Step 3.
	The disk ownership reflects the cluster’s A loop B loop topology...	Go to Step 3.
3	Disable clustering by entering the following command. cf disable	
4	On each cluster node, uninstall the license for SnapMover. license delete snapmover	
5	Reboot each cluster node, and when prompted, press Ctrl-C to display the boot menu options.	

Step	Action
6	On each cluster node, enter the choice for booting in maintenance mode.
7	In maintenance mode on each cluster node, enter the following command: disk remove_ownership This command disables software-based disk ownership.
8	Halt each cluster node to exit maintenance mode: halt
9	Reboot each cluster node in normal mode. The system reverts to ownership based on A loop and B loop connections.
10	To re-enable the cluster operation, enter the following command: cf enable

About this chapter This chapter discusses how to perform virus scanning for vFiler units that run the CIFS protocol.

Topics in this chapter This chapter discusses the following topics:

- ◆ “[Understanding virus scanning for a vFiler unit](#)” on page 150
- ◆ “[Configuring virus scanning for a vFiler unit](#)” on page 152

Understanding virus scanning for a vFiler unit

Who can configure virus scanning

You (the hosting storage system administrator) can specify how virus scanning works on files owned by the hosting storage system and files owned by vFiler units. vFiler unit administrators, however, cannot configure virus scanning for vFiler units.

Storage Systems with which virus scanners can be registered

Virus scanners can be registered with the hosting storage system or any vFiler unit. You can determine whether files on a vFiler unit are scanned by virus scanners registered with the hosting storage system or the vFiler unit.

Note

A virus scanner local to a vFiler unit always takes precedence over the virus scanner registered with the hosting storage system. If a virus scanner is registered with a vFiler unit and that scanner is functional, the files on the vFiler unit will be scanned by this scanner. If this scanner becomes unavailable, and the `vscan options use_host_scanners` command is set to On and a scanner is registered with the hosting storage system, the files on the vFiler unit will be scanned by the hosting storage system's virus scanner until the scanner local to the vFiler unit becomes available.

Virus scanning for vFiler0

Because vFiler0 is the hosting storage system, files on vFiler0 can be scanned only by virus scanners registered with the hosting storage system.

Requirements for virus scanning for a vFiler unit other than vFiler0

These requirements must be met before you can scan files on a vFiler unit other than vFiler0:

- ◆ Virus scanning must be enabled for each vFiler unit.
- ◆ A virus scanner must be available for the vFiler unit because one of the following requirements is met:
 - ❖ A virus scanner must be registered with the vFiler unit.
 - ❖ A virus scanner must be registered with the hosting storage system and the vFiler unit must be allowed to use it.

**Effect of virus
scanner availability
on CIFS access**

Even if virus scanning is enabled and the mandatory_scan option for the vscan command is set to On, CIFS clients of the vFiler unit are not allowed to open any files on the vFiler unit if no virus scanner is available.

Configuring virus scanning for a vFiler unit

Configuring virus scanning

To specify the virus scanner for scanning files on a vFiler unit, complete the following steps.

Step	Action
1	Enter the following command to specify the virus scanner for scanning files on a vFiler unit: <pre>vfiler run vfilertemplate vscan options use_host_scanners { on off }</pre> Set the <code>use_host_scanners</code> option to On to allow the vFiler unit to use the virus scanner registered with the hosting storage system. Doing so enables the hosting storage system and its vFiler units to share the virus scanner. However, the vFiler unit uses the virus scanner registered with the hosting storage system only when the virus scanner registered with the vFiler unit is unavailable. Set the <code>use_host_scanners</code> option to Off if you do not want to allow the vFiler unit to use the virus scanner registered with the hosting storage system. Note_____ The <code>use_host_scanners</code> option is applicable only to a vFiler unit you created. You cannot set it on vFiler0 or a storage system. _____
2	Enter the following command to enable or disable virus scanning: <pre>vfiler run vfilertemplate vscan { on off }</pre>

The challenge

The various networking and vFiler unit capabilities available on the NetApp storage system can be used together to create a scalable, secure, and shared network without using a large number of physical network adapters. However, understanding how this network can be created can be challenging. This section uses an example to describe how to use vifs, VLANs, vFiler units, and IPspaces together to create such a network.

The need

Application Service Provider (ASP) A wants to build a highly available network architecture that can provide 32 separate, isolated customer networks using four dual port network adapters (e1a, e1b, e2a, e2b, e3a, e3c, e4a, e4b) on a storage system, StorageSystem1.

The answer

To achieve the above network configuration without additional physical infrastructure, the following need to be created:

- ◆ 1 single and 2 multimode vifs
- ◆ 32 VLANs
- ◆ 32 vFiler units, each binding to an IP address (from the sequence 10.10.1.1, 10.10.2.1, and so on), associated with a qtree in vol0 (/vol/vol0/cust1, vol/vol0/cust2, and so on)
- ◆ 32 IPspaces

To implement the above network configuration, perform the following steps.

Note

You must add the vif and VLAN commands to the /etc/rc file in order to make them persistent across reboots.

Step	Action
1	Enter the following commands to create two multimode vifs consisting of four ports each: <pre>vif create multi active1 e1a e2a e3a e4a vif create multi passive1 e1b e2b e3b e4b</pre> Note_____ Make sure you also create multimode trunks on the switches to support the vifs.
2	Enter the following command to create a second-level vif consisting of the vifs you created in the previous step: <pre>vif create single failoverpair active1 passive1</pre>
3	Enter the following commands to create VLAN interfaces on the second-level vif called failoverpair: <pre>vlan create failoverpair customer1vlan vlan add failoverpair customer2vlan vlan add failoverpair customer3vlan</pre> Repeat the <code>vlan add</code> command until you have 32 VLAN interfaces.
4	Enter the following commands to create IPspaces—one per customer—and to assign them to the VLAN interfaces you created previously: <pre>ipspace create customer1ipspace customer1vlan ipspace create customer2ipspace customer2vlan</pre> Repeat the <code>ipspace create</code> command until you have created 32 IPspaces.
5	Enter the following commands to create vFiler units: <pre>vfiler create cust1vfiler -s customer1ipspace -i 10.10.1.1 /vol/vol0/cust1 vfiler create cust2vfiler -s customer2ipspace -i 10.10.2.1 /vol/vol0/cust2</pre> Repeat the <code>vfiler create</code> command until you have created 32 vFiler units.

Index

Symbols

- /etc directory
 - configuration files in 16
 - effects of vFiler unit creation on 13
- /etc/dgateways file 56
- /etc/exports file
 - changing path names in after renaming volume 49
 - exporting all file systems in 82
- /etc/messages file 42
- /etc/quotas file
 - format of 91
 - language for encoding 14
- /etc/snapmirror.conf file 57
- /etc/usermap.cfg file, language for encoding 14

A

- activating a disaster-recovery vFiler unit 110, 116, 119
- adding and removing vFiler unit resources, effects of 25
- adding new disks 99
- addresses relative to IPspace 64
- administering resources from the hosting storage system 24
- aggr add command 99
- aggr create command 100
- aggr status command 146
- aggregates, for a vFiler unit 49
- allowing or disallowing
 - CIFS or NFS, effects of 38
 - iSCSI, effects of 38
 - protocols, steps for 39
 - rsh, effects of 39
- assigning interfaces to IPspaces 73
- assigning ownership to disks. *See* software-based disk ownership
- AutoSupport messages for a vFiler unit 42

B

- backing up a vFiler unit 51
- backup vFiler unit 97, 110
- broadcast packets in an IPspace 65

C

- checklists
 - network (for migration to destination storage system) 108
 - storage (for migration to destination storage system) 107

CIFS

- local user accounts 83
- path names for shares 81
- prefdc command (domain controller) 116
- preparing for 83
- statistics 59
- support for a vFiler unit 38
- virus scanning 151
- cifs setup command 22, 23
- cifs stat command 59
- clients, effect of vFiler unit move on 135
- clusters
 - considerations when used with IPspaces 67
 - naming IPspaces for 67
- commands for a vFiler unit
 - entering from hosting storage system 43
 - entering from vFiler unit context 43
 - entering through rsh 46
 - executable from hosting storage system 43
 - ipspace command 71
 - prerequisites for entering through rsh 45
 - setup 82
 - vfiler add command 27
 - vfiler allow command 39
 - vfiler command, purpose of 21
 - vfiler context command 44
 - vfiler create command 19
 - vfiler destroy command 36
 - vfiler disallow command 40
 - vfiler dr activate command 116

- vfiler dr configure command 113
- vfiler dr delete command 118
- vfiler dr resync command 120
- vfiler dr status command 113
- vfiler migrate cancel command 137
- vfiler migrate complete command 137
- vfiler migrate start command 138
- vfiler migrate status command 136
- vfiler move command 28
- vfiler remove command 27
- vfiler run command 43
- vfiler setup command 22
- vfiler start command 34
- vfiler status command 41, 42, 99
- vfiler stop command 33
- ways to execute 43
- consolidating multiple servers 2
- context of vFiler unit, switching to 44
- copying vFiler unit data 135
- creating a vFiler unit
 - in nondefault IPspaces 76
 - options with default settings 16
 - prerequisites for 13
 - required status of network interface 17
 - results of 16
- creating disaster recovery vFiler unit 110, 112
- creating volumes 100

D

- daemon
 - disabling the routed 11
 - effects of disabling the routed 56
 - enabling the routed 11
- data migration, using a vFiler unit for 3, 98
- Data ONTAP custom MIBs 58
- Data ONTAP tasks 7
- decreasing the limit for vFiler units per storage system 31
- default IPspace, interfaces in 64
- default limits for vFiler units per storage system 29
- default vFiler unit, definition of 4
- deleting disaster recovery vFiler unit 118
- destination storage system, preparing for migration 99

- destroying a vFiler unit 35
- destroying IPspaces 75
- disabling MultiStore license 11
- disabling SnapMover vFiler unit migration 146
- disallowing or allowing
 - CIFS or NFS, effects of 38
 - iSCSI, effects of 38
 - protocols, steps for 39
 - rsh, effects of 39
- disaster recovery
 - about 98
 - quotas 101
 - vFiler unit 110
- disk assign command 145
- disk ownership. *See* software-based disk ownership
- disk upgrade-ownership command 143
- disks for a vFiler unit, moving 14
- displaying
 - quota reports 93
 - quota status 92, 101
- distinct IP address space 62
- DNS servers
 - and vFiler unit 3
 - migration disaster recovery 105
- domain controller name, changing 116
- domains, multiple 3

E

- enabling MultiStore license, effects of 11
- entering commands from the vFiler unit 43
- enterprises
 - using a vFiler unit for 3
 - using IPspaces 69
- establishing a vFiler unit, major steps for 10
- expanding storage system 145

F

- FCP LUNs 52
- FilerView
 - configuring resources with 18
 - using to manage hosting storage system resources 7
 - using to manage vFiler unit resources 8
- flexible volume. *See* FlexVol volume

- FlexVol volume
 - for a vFiler unit 13, 49
- FTP support for a vFiler unit 38

H

- hosting storage system
 - access to data on a vFiler unit 5
 - administering vFiler unit from 24, 44
 - administration tasks 7
 - CIFS commands on 83
 - definition of 4
 - format of quota reports on 93
 - managing a vFiler unit from 7
 - managing vFiler unit storage from 24
 - quotas not copied to new vFiler unit 101
- HTTP support for a vFiler unit 38

I

- incoming packets for an IPspace 65
- increasing the limit for vFiler units per storage system 29
- interfaces, assigning to IPspaces 73
- IP addresses
 - and migration, disaster recovery 102
 - assigned to a vFiler unit 17
 - configuring for vFiler unit creation 17
 - in different IPspaces 76
 - planning for when creating a vFiler unit 13
 - removing from a vFiler unit 25
 - removing from an interface 73
- IP alias, removing 17
- IPSec, configuring vFiler unit for 56
- ipspace command 70
- IPspaces
 - and migration, disaster recovery 103
 - assigning interfaces to 73
 - case studies of 69
 - considerations when used with clusters 67
 - creating 71
 - creating a vFiler unit in nondefault 76
 - destroying 75
 - destroying vFiler units in different 35
 - guidelines for using 62
 - incoming and outgoing packets in 65

- interfaces in 64
- listing 72
- managing using the ipspace command 70
- names for in a cluster 67
- restricting broadcast packets in 65
- routing tables for 56, 62, 65
- typical applications of 62
- VLAN tagging and 66
- iSCSI and a vFiler unit 54
- iSCSI LUNs 52
- iSCSI service on a vFiler unit 54
- iSCSI support for a vFiler unit 38

L

- language, guidelines for 14
- license for MultiStore, enabling or disabling 11
- licenses required for vFiler unit migration 140
- limit on the number of vFiler units per storage system 29
- listing IPspaces 72
- local user accounts 83
- LUNs on a vFiler unit 52

M

- managing a vFiler unit 8
- managing storage system
 - backups and data recovery 7
 - Snapshot and SnapMirror 7
 - volumes, disks, and RAID groups 7
- mandatory_scan option 151
- maximum number of vFiler units 4
- messages, AutoSupport for a vFiler unit 42
- MIBs
 - Data ONTAP custom 58
 - standard 58
- migrate
 - by copying data 135
 - data using a vFiler unit 3
 - storage system data 3
 - ways to 133
- moving a vFiler
 - how to 133
- moving a vFiler unit
 - quotas 101

- moving resources, about 25
- multiple security domains 3
- multiple server consolidation 2

N

- naming a vFiler unit, guidelines for 14
- naming storage resources in a vFiler unit 14
- network checks for migration to destination storage system 101, 108
- network interfaces
 - configuring down 18
 - for a vFiler unit 5
 - removing an IP alias from 17
- network resources
 - adding (to a vFiler unit) 27
 - base IP address 17
 - changing (for a vFiler unit) 25
 - considerations for moving between vFiler units 26
 - IP alias 17
 - planning for when creating a vFiler 13
 - removing (from a vFiler unit) 27
 - requirements for moving and removing 25
 - steps for moving between vFiler units 28
 - types of 2
- NFS
 - path names for exporting 81
 - starting the protocol 82
 - statistics 59
 - support for a vFiler unit 38
- nfsstat command 59
- NIS servers
 - and migration, disaster recovery 104
 - and vFiler unit 3
- no-copy transfer 140
- no-vfiler-ips? variable, setting 129

O

- oplock settings for qtrees on a vFiler unit, changing 50
- options
 - after vFiler unit creation 16
 - settable on a vFiler unit 46
 - specifying path names for 47

- options snapmirror command 127
- options snapmirror.access command 127
- outgoing packets for an IPspace 65

P

- packets
 - broadcast in an IPspace 65
 - incoming for an IPspace 65
- partitioning storage system resources 3
- path names, for NFS exports and CIFS shares 81
- performance, monitoring 59
- primary storage unit, definition of 13
- protocols
 - allowing or disallowing 38
 - supported for a vFiler unit 38
- pseudo-root, definition of 80

Q

- qtree command output (how it differs for a vFiler unit and hosting storage system) 50
- qtrees
 - assigning to a vFiler unit 14
 - creating on a vFiler unit 49
 - creating on vFiler0 49
 - oplock settings for 50
 - security styles for 50
 - who can create on a vFiler unit 49
- quota report command 101
- quotas
 - displaying status of 92
 - effects of destroying a vFiler unit on 35
 - guideline for 15
 - information in standard MIB 58
 - messages from exceeding thresholds 95
 - prerequisite for turning on and off 87
 - reports on 93
 - resizing 90
 - seeing where enforced from 101
 - that are copied to new vFiler unit 101
 - tree 93
 - turning on and off 88
 - types supported for a vFiler unit 86
 - user and group 91
 - vFiler unit names in report 93, 94

where user and group quotas take effect 91
who specifies 86

R

rebooting storage system, effects on a vFiler unit 48
removing IP addresses from an interface 73
removing IP aliases from an interface 17
removing resources 25
renaming a vFiler unit 32
replacement vFiler unit, creating after disaster 132
resizing quotas 90
resources
 administering from the hosting storage system 24
 assigning 18
 changing 25
 guidelines for assigning 13
 network 2
 storage 2
restricting storage system traffic 3
resynchronizing, vFiler units 119
routed daemon
 disabling 11
 effects of disabling 56
 enabling 11
routing table
 for a vFiler unit in default IPspace 56
 for the storage system 56
rsh
 access to vFiler unit from clients 45
 allowing or disallowing on a vFiler unit 39
 enabling 45
 entering commands for a vFiler unit through 43
 support for a vFiler unit 38
rsh.access option 45
rsh.enable option 45
running state of a vFiler unit 16

S

sample vFiler unit 5
scanners, virus 150
secure routing and IPspace 64
server consolidation 2
Server Manager 83

service providers, using a vFiler unit for data hosting 3
setting up a vFiler unit
 introduction to 22, 82
 steps for 23
setup command for a vFiler
 syntax for 23
 unitpurpose of 22
setup command for a vFiler unit
 purpose of 82
SnapMirror
 on hosting storage system 57
 options snapmirror.access command 127
 using to reactivate vFiler unit 126
snapmirror break command 128
snapmirror commands
 considerations for 57
 snapmirror quiesce command 127
 snapmirror resync command 126
 snapmirror update command 127
 where to enter 57
snapmirror update command 130
snapmirror.access option 127
snapmirror.allow file 127
SnapMover
 about 133, 140
 disabling 146
 enabling 142
 vFiler unit migration 97, 144
SNMP
 on the hosting storage system 58
 using with a vFiler unit 58
software-based disk ownership
 about 140
 assigning ownership 145
 changing, in order to migrate a vFiler unit 133
 disk upgrade-ownership command 143
 enabling 142
 expanding storage in clusters 145
SSPs and IPspaces 69
standard MIB 58
starting a vFiler unit
 meaning of 33
 step for 34
state of a vFiler unit

- after vFiler unit creation 16
- displaying 41
- effect of storage system reboot on 48
- stopping a vFiler unit 33
- storage area network (SAN) guidelines 15
- storage checks for migration to destination storage system 99, 107
- storage resources
 - adding (to a vFiler unit) 27
 - administering from the hosting storage system 24
 - assigning qtrees 14
 - assigning volumes 13
 - changing (for a vFiler unit) 25
 - considerations for moving between vFiler units 26
 - moving between a vFiler unit 28
 - naming 14
 - planning for when creating a vFiler unit 13
 - primary unit 13
 - removing (from a vFiler unit) 27
 - requirements for moving and removing 25
 - types of 2
- storage system data migration 3
- storage system partitioning 2
- storage system reboot, effects on a vFiler unit 48
- storage system resources, partitioning 3
- subnet, moving vFiler unit to a different 138
- syslogd messages 42
- sysstat command 59

T

- TCP statistics in standard MIB 58
- thresholds, quota 95
- timezone 22
- traditional volume for a vFiler unit 13, 49
- traditional volume for vFiler unit migration 141
- traffic, restricting on storage system 3
- trusted host
 - and migration, disaster recovery 106
 - changing after migration 139
 - changing on disaster-recovery vFiler 114

U

- UDP statistics in standard MIB 58
- uptime command 59
- User Manager 83
- UUID for vFiler unit 41

V

- vFiler unit
 - administering resources from hosting storage system 24
 - aggregates, assigning 49
 - backup for disaster recovery 97
 - changing resources for 25
 - CIFS support 38
 - commands. *See* commands for a vFiler unit
 - copying data from a 135
 - decreasing the limit of 31
 - default 4
 - default limit 29
 - disaster recovery
 - about 98
 - activating a 110, 116, 119
 - checking storage space 107
 - creating a 110, 112, 114
 - deleting a 118
 - networking information for 108
 - preparing for 99, 101
 - displaying status 41
 - FTP support 38
 - group in Data ONTAP custom MIB 58
 - HTTP support 38
 - increasing the limit of 29
 - iSCSI support 38
 - limit per hosting storage system 4, 29
 - LUNs and 52
 - maximum number of 4, 29
 - migrating a 140
 - migration
 - copying data for 134
 - disabling 146
 - method for 133
 - SnapMover, enabling 142
 - traditional volumes 141
 - using the vfiler migrate command for 141

- moving
 - and migrating 98
 - by copying data 135
 - to different subnet 138
 - to different Windows domain 138
- NFS support 38
- no-copy transfer 140
- preparing backup for disaster recovery 110
- protocols supported 5
- reactivating by resynchronizing 119, 120
- reactivating using vfiler dr commands 129
- reactivating via SnapMirror commands 126
- re-creating a destroyed 37
- rename 32
- replacing on original storage system 130
- resynchronization (resync)
 - handling failures 124
 - how it works 120
- RSH support 38
- server consolidation, using for 2
- setup 22, 23, 82
- SnapMover 97
- states, effect of storage system reboot on 48
- states, stopped or running 33
- traditional and FlexVol volumes 13, 49
- using vifs with 153
- using VLAN tagging with 153
- vFiler0 4
- vFiler0

- definition of 4
- included in vFiler unit limit 29
- vfilertemplate, defined 21
- vifs, using with vFiler units 153
- virus scanning
 - configuring 150
 - registering scanners for 150
 - requirements for 150
- VLAN tagging
 - used with IPspaces 66
 - using with vFiler units 153
- volumes in a vFiler unit
 - assigning 13
 - effects of renaming 49
 - taking offline 49

W

- Windows domain
 - and migration, disaster recovery 106
 - changing 116
- WINS server
 - and migration, disaster recovery 106
 - changing 116
 - changing after migration 139
- worksheet for migration, disaster recovery
 - networking 108
 - storage 107

